

Supplementary Material

Software Requirements:

Python Version 2.7
Biopython Version 1.60
Python Modules: httplib, urllib2, ete2 (An Environment for Tree Exploration (ETE), Optional)
ARB Version 5.1

1 **Goal:** Importing alignment, phylogenetic tree and meta-data from external resources into ARB.

2 **Motivation:** Exploratory Analysis and Primer design in ARB.

3 **Dataset:** Major Facilitator Superfamily (MFS) Protein Family (440 sequences)

4 Available from the NCBI Conserved Domain Database (CDD)

5 <http://www.ncbi.nlm.nih.gov/Structure/cdd/cddsrv.cgi?uid=212314>

6 MFS: "...a large and diverse group of secondary transporters that includes uniporters,
7 symporters, and antiporters. MFS proteins facilitate the transport across cytoplasmic or
8 internal membranes of a variety of substrates including ions, sugar phosphates, drugs,
9 neurotransmitters, nucleosides, amino acids, and peptides."

10

Procedure:

11 1. Download the MFS dataset from CDD. Two files: 1 for alignment and 1 for tree.

12 <http://www.ncbi.nlm.nih.gov/Structure/cdd/cddsrv.cgi?hsfl=1&uid=cd06174&seqhrch>

13

14 *Note: Set the Type Selection to 'top listed sequences' and the Aligned Rows to All 440
15 Rows before downloading the alignment file via the Structure View button.
16

The screenshot shows the NCBI CDD interface for the cd06174 sequence cluster. The top left panel displays dataset metadata: Aligned: 440 rows, ThresholdBitScore: 36.9081, ThresholdSettingGI: 74575184, Created: 9-Jul-2008, Updated: 23-Oct-2008. It also includes links to download Cn3D models and scroll to sequence alignment. The main central area is a hierarchical tree visualization of the sequence cluster, with nodes colored by superfamily. A callout box points to the tree with the text "Select 'top listed sequences'". To the right, a "Sub-family Hierarchy" panel offers an interactive display with a CDTREE. At the bottom, a "Sequence Alignment" panel shows a sequence alignment for Feature 1 (1PW4_A) with a color-coded sequence logo. A callout box points to the alignment panel with the text "Select 'top listed sequences'". On the left, a "Structure" panel has its "Aligned Rows" dropdown set to "All 440 rows". A callout box points to this dropdown with the text "Set to 440". Another callout box points to the "Program" dropdown in the "Structure" panel with the text "Download Cn3D".

17

Supplementary Material

- 18 2. Download/Install CDTree tool for dataset visualization and extraction of tree.
19 <http://www.ncbi.nlm.nih.gov/Structure/cdtree/cdtree.shtml>

The screenshot shows the NCBI CDTree homepage. At the top, there's a search bar for Entrez databases and links for Structure Home, 3D Macromolecular Structures, Conserved Domains, PubChem, BioSystems, and Help?. On the left, a sidebar includes links for CD-Search, Find CDs, CDD home, CDD help, NCBI Handbook, Structure, MMBDB, Cn3D, VAST, Research, and CDD FTP site, with a note that it was revised on 08/01/2011. The main content area features a large phylogenetic tree with colored nodes representing different protein domains. To the right of the tree, there's a summary of CDTree's features, including its integration with Cn3D, PSI-BLAST interface, and various analysis tools. A separate section highlights new features added in CDTree 3.1.

- 20 3. Open MFS dataset project (*cddsrv.cn4*). Export MFS phylogenetic tree from CDTree:
21 a. Right click>Select Sequence Tree
22

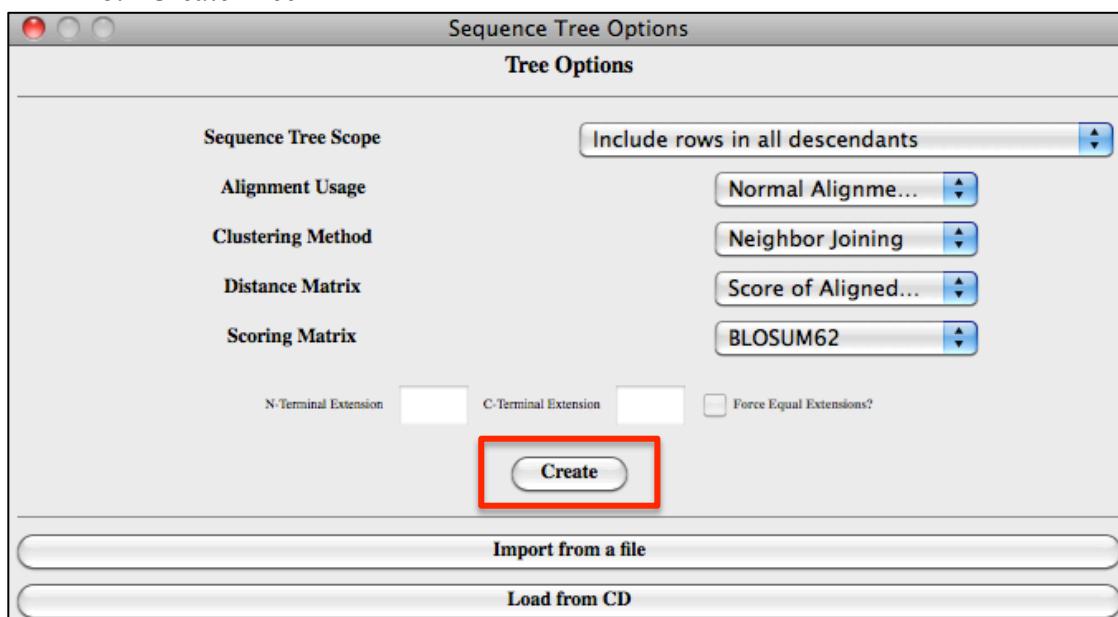
The screenshot shows the Cn3D interface with a sequence tree viewer. The tree has nodes labeled with domain names like "cd06174", "440/440", and "MFS". A context menu is open over one of the tree nodes, listing options such as "CD Info Viewer", "Annotations", "Annotations Matrix", "Sequence List", "Sequence Tree" (which is highlighted in blue), "Cross Hits", "Taxonomy", and "CDART". A status bar at the bottom indicates that the "MFS" dataset contains 440 sequences and 440 rows.

23
24
25

Supplementary Material

26

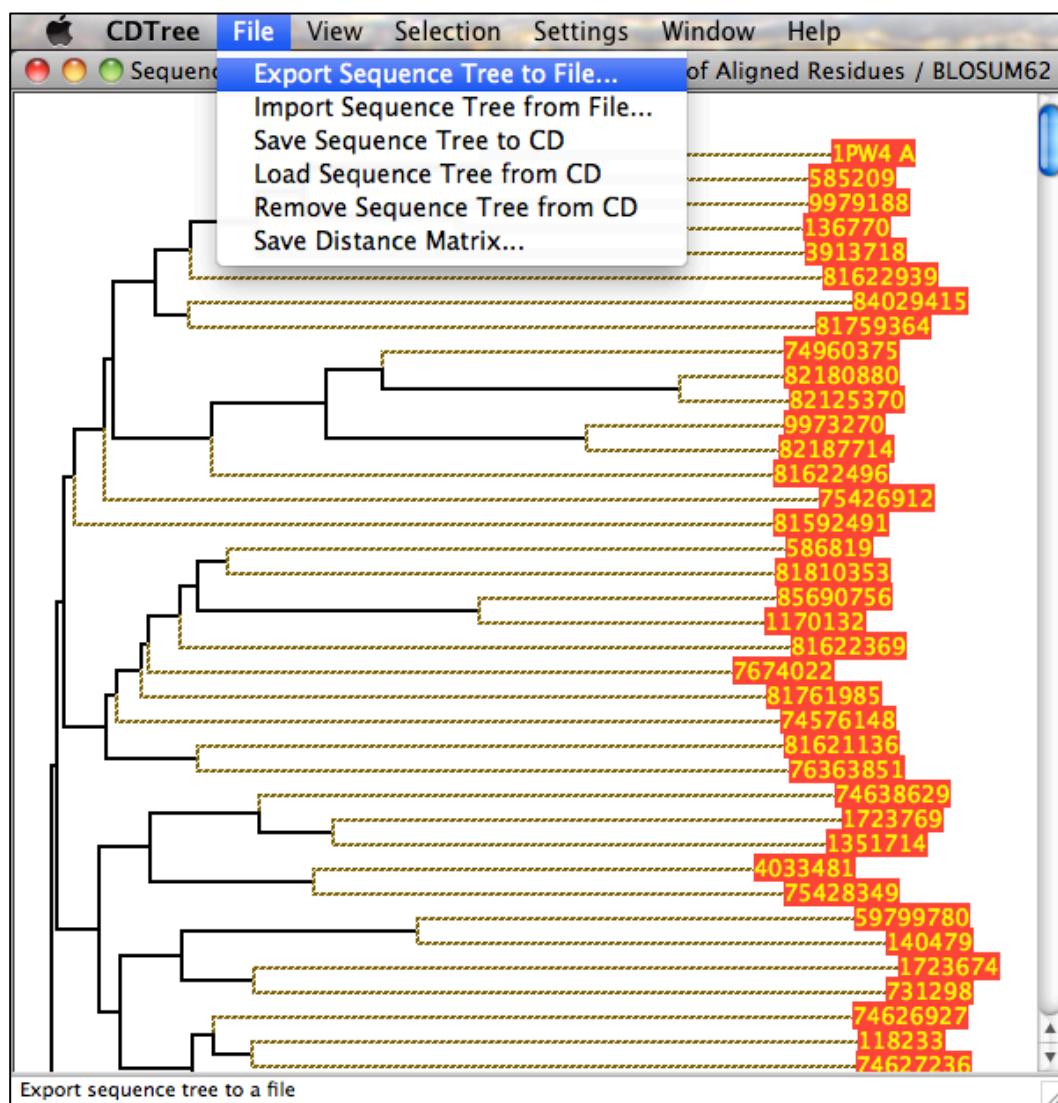
b. Create Tree



27

28

c. Export Sequence Tree: Save as *MFS.tree*



Supplementary Material

29
30
31

4. Download/Install/Run Cn3D tool for exporting MFS alignment.
- <http://www.ncbi.nlm.nih.gov/Structure/CN3D/cn3d.shtml>

The screenshot shows the NCBI Structure Home page with the Cn3D viewer open. The viewer interface includes a toolbar at the top with options like File, View, Select, Style, Window, CDD, Help, and a menu for CDD Descriptive Items. Below the toolbar is a main window displaying a 3D molecular structure with purple and yellow annotations. A tooltip for 'WD40' is shown, providing details about the domain: 'WD40; WD40 domain, found in a cover a wide variety of functions modules in signal transduction, protein assembly, typically contains a short N-terminal and the C-terminal 40 residues, hence the name a conserved core; serves as a structural tetrad'. To the right of the viewer, there are sections for 'Highlights' (New Features in Cn3D 4.3, New Features in Cn3D 4.1, New Features in Cn3D 4.0), and a 'Conservation with CDTree' section.

32
33

5. Open MFS dataset project (*cddsrv.cn3*). Export alignment: Save as *MFS_Align.fasta*

The screenshot shows the Cn3D application window with the 'View' menu expanded. The 'Export...' option is highlighted, with a submenu showing 'Fasta', 'A2M FASTA', 'Text', 'HTML', and 'PSSM'. The main workspace shows a sequence alignment viewer with a large amount of sequence data and alignment results. The status bar at the bottom indicates the date and time as Fri 4:03 PM.

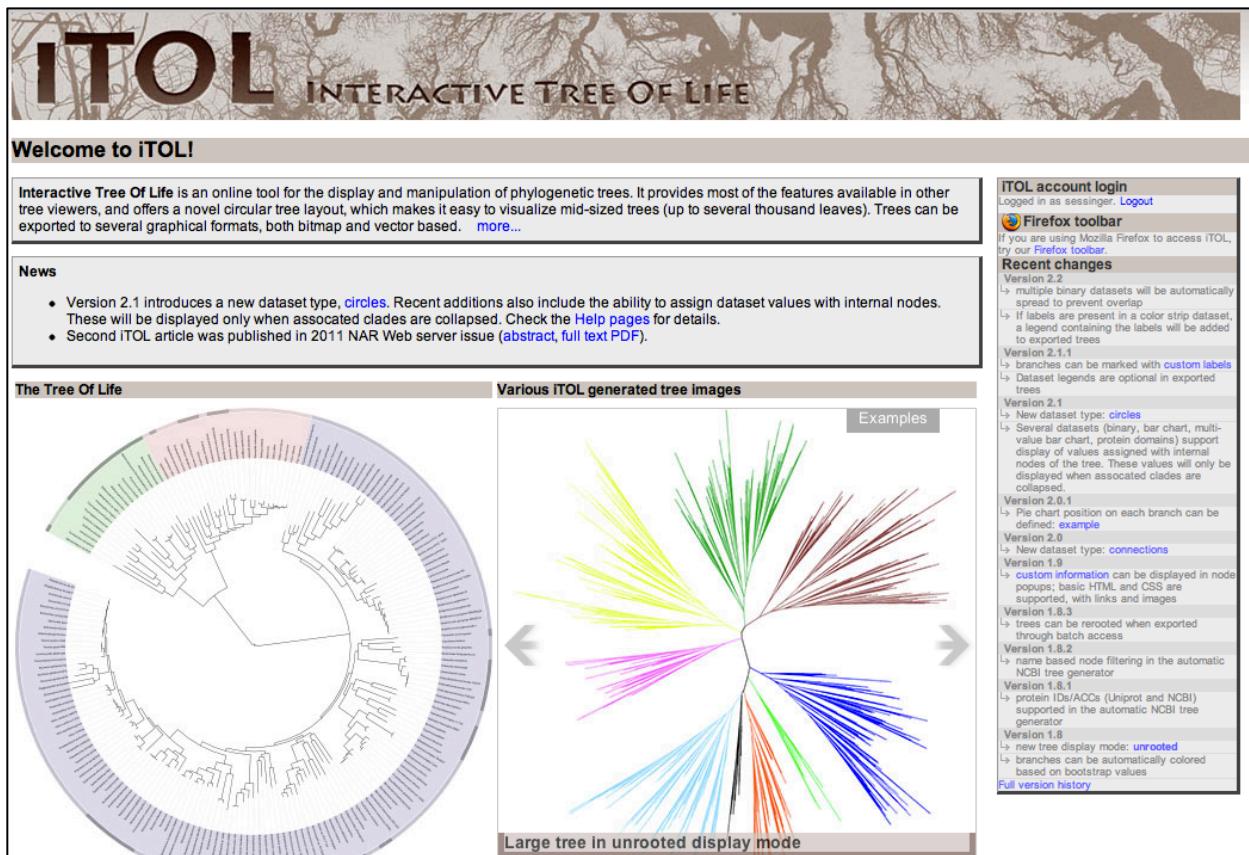
34
35

Supplementary Material

36 6. Assemble the Database -- Collect Meta-Data and Sequences
37
38 From a terminal, change the directory to the ***GetDatabase*** folder. Run the script
39 ***buildDatabase.sh*** (e.g. `./buildDatabase.sh`) to obtain the GenBank files for each sequence
40 in the dataset. Note that this script may only be run between the hours of 9pm-5am EST
41 as per NCBI requirements. Once all of the GenBank files have been downloaded the
42 script will proceed to construct the custom database. The completed database
43 (***MFS_metaData.txt***) may be found at ***GetDatabase/MFS/Output/*** along with the names
44 of the acquired meta-data fields (***MFS_Field_labels.txt***). This second file is required for
45 building the ARB import filter described in the following section. Note that there are
46 additional files in the Output directory that are not used as part of this tutorial.
47
48 The ***buildDatabase.sh*** script contains a python script (***NextractGB.py***) for constructing
49 the custom database from the downloaded GenBank files. The ***NextractGB.py*** is also
50 capable of reconstructing the entire NCBI taxonomy and using this information to
51 populate taxonomy field labels within the custom database. To enable this feature you
52 must uncomment lines 89-100 in the ***NextractGB.py*** file. Note that this feature requires at
53 least 6GB of RAM (disabled by default). The supporting files and functions for this
54 feature is contained in the ***GetDatabase/NCBI_Taxonomy/*** directory.
55
56 ***buildDatabase.sh*** dependencies:
57 Python version 2.7
58 Biopython version 1.60
59 Python modules httplib, urllib2
60
61 ***NCBI_Taxonomy*** dependencies (additional):
62 taxid2lineageModule.py module (script in NCBI_Taxonomy)
63 ete2 python module (An Environment for Tree Exploration (ETE))
64 <http://pypi.python.org/pypi/ete2/>
65 ete2 dependencies:
66 Python Modules:
67 numpy, scipy, pyqt4, mysq, lxml
68

Supplementary Material

- 69 7. From the directory containing ***MFS_Field_labels.txt***, run the ***buildFilter.py*** (e.g. python
70 ***buildFilter.py***) script to generate the ARB import filter for the custom database. Use the
71 ***MFS_Field_labels.txt*** as the input file and name the output, ***MFS_import_filter.ift***. Place
72 it in the ARB directory: ***arb/lib/import/*** (Typically, this directory is located at
73 ***/opt/local/share/arb/lib/import/*** in OSX).
- 74
- 75 8. We need to rename the headers in the alignment file and the leaves on the tree to the
76 respective sequence's unique ID. This will ensure that we can link the meta-data,
77 sequences, alignment and tree all together in ARB.
- 78
- 79 a. For the alignment:
80 i. Run the ***getAccession.py*** script from the directory containing
81 ***MFS_metaData.txt***. This will produce an alignment file
82 (***MFS_UID.fasta***) that replaces the sequence headers with the respective
83 unique ID.
- 84 b. For the tree labels:
85 i. We will use the Interactive Tree of Life (iTOL) to revise our tree labels.
86 <http://itol.embl.de/>



Supplementary Material

87

ii. Login and upload the tree file from step 3c: **MFS.tree**

Project 1
No description.
Tree Description Datasets Saved views Inserted Features
There are no trees associated with this project
Upload a new tree to this project
Add a new project...
Citations: Letunic and Bork (2006) Bioinformatics 23(1):127-8 and Letunic and Bork (2011) Nucleic Acids Res doi: 10.1093/nar/gkr201

ITOL INTERACTIVE TREE OF LIFE
Upload a tree into the project 'Project 1' on workspace 'MFS'
Use this form to upload your own phylogenetic tree. It should be in plain text, in one of supported formats (Newick, Nexus or PhyloXML). Please check our help pages
You can either paste your tree into the box, or upload a file using the file selector below. Don't forget to select the correct tree format.
Paste or type the tree:
Paste your tree here...
Upload a file which contains your tree:
Choose File No file chosen
Tree format: Newick
Make sure the file is plain text, and contains only trees in the selected format.
Optional information
Tree name: MFS If you don't specify a name, a numeric ID will be used instead
Tree description: can also be set/changed later in your account page
Advanced options (show)
Upload (if you're uploading extra data with your tree, fill the dataset section below before clicking "Upload")

ITOL INTERACTIVE TREE OF LIFE
Multiple trees successfully uploaded
Tree upload summary:
The following table shows the uploaded trees and their corresponding projects. Use the links next to each tree to interactively assign color ranges (opens in a new window). You can also automatically assign taxonomy to multiple trees by ticking the 'Assign' checkbox and clicking the 'Automatically assign taxonomy' button below the table.

Name	Project	Automatic taxonomy assignment	Other options
MFS	Project 1	<input type="checkbox"/> Assign	Define color ranges
14411815766877813468824281	Project 1	<input type="checkbox"/> Assign	Define color ranges

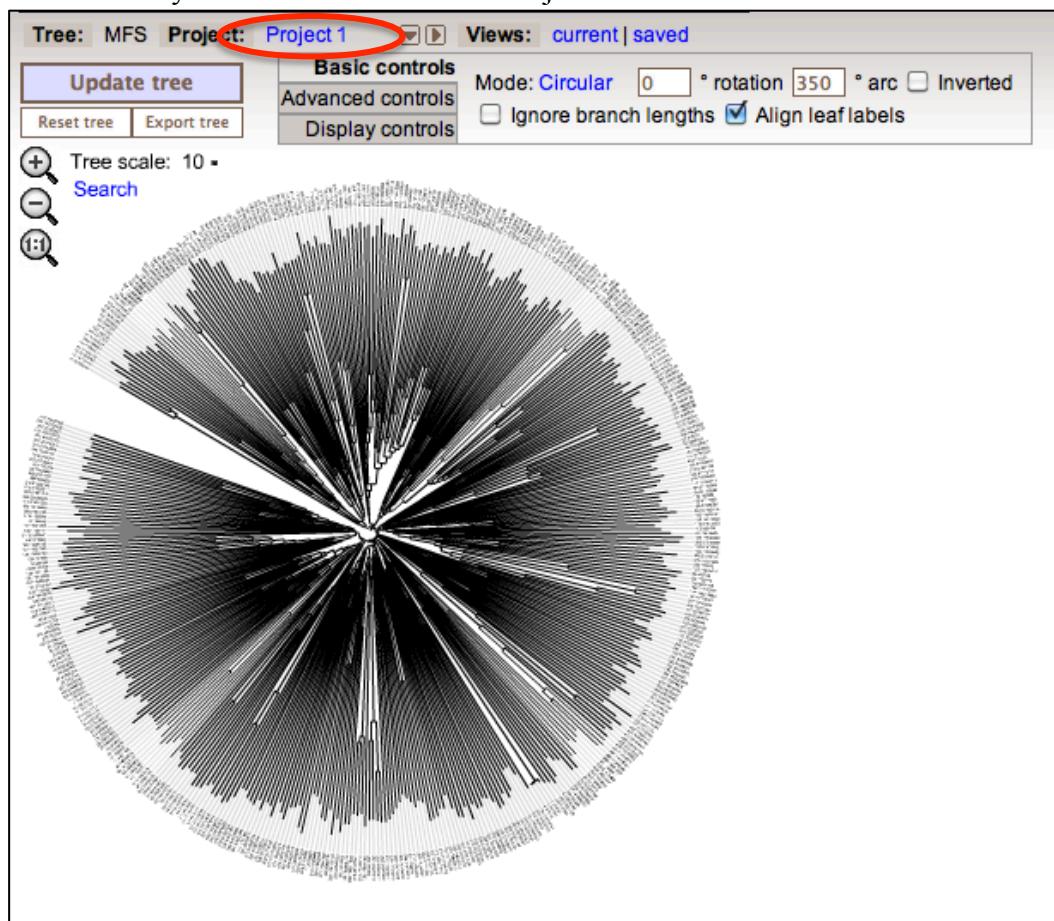
Select all
Automatically assign taxonomy Back to your personal page

88

Supplementary Material

89

Verify tree then return back to Project 1.



90

91

92 iii. Rename the leaves using the file *TreeLabels_Mapped_New.txt*

93 *This file was created using the *rename_tree_leaves.py* script (see Appendix A).

The screenshot shows the 'Project 1' interface in MFS. It displays two trees: 'MFS' (uploaded tree) and '14411815766877813468824281' (uploaded tree). A context menu is open over the second tree, with the 'Tree options' button highlighted with a red oval. The menu options include: Rename tree, Edit tree, Root mid-point, Duplicate tree, Re-label leaves (highlighted with a red oval), Assign taxonomy, Edit colors, and Add a dataset. A tooltip for 'Re-label leaves' states: 'Allows you to edit the leaf labels. You can also upload a file and re-label the whole tree at once.' At the bottom, citation information is provided: 'Citations: Letunic and Bork (2006) Bioinformatics 23(1):127-8 and Letunic and Bork (2011) Nucleic Acids Res doi: 10.1093/nar/gkr234'.

94

95

96

97

98

Supplementary Material

99

Use the file *TreeLabels_Mapped_New.txt* here.

Editing leaf labels for tree 'MFS'

You can either use the form below to directly edit the labels, or upload a file with label definitions. The file should be in plain text and tab separated, with one label definition per line. First field should contain the leaf ID and the second one the new label.

Note: The labels are independent from actual leaf IDs in the tree. If you add new datasets to this tree, the dataset file must use the original leaf IDs and not the labels defined here.

Option 1: Upload label definitions

Label definition file: Choose File No file chosen

Option 2: Edit labels directly

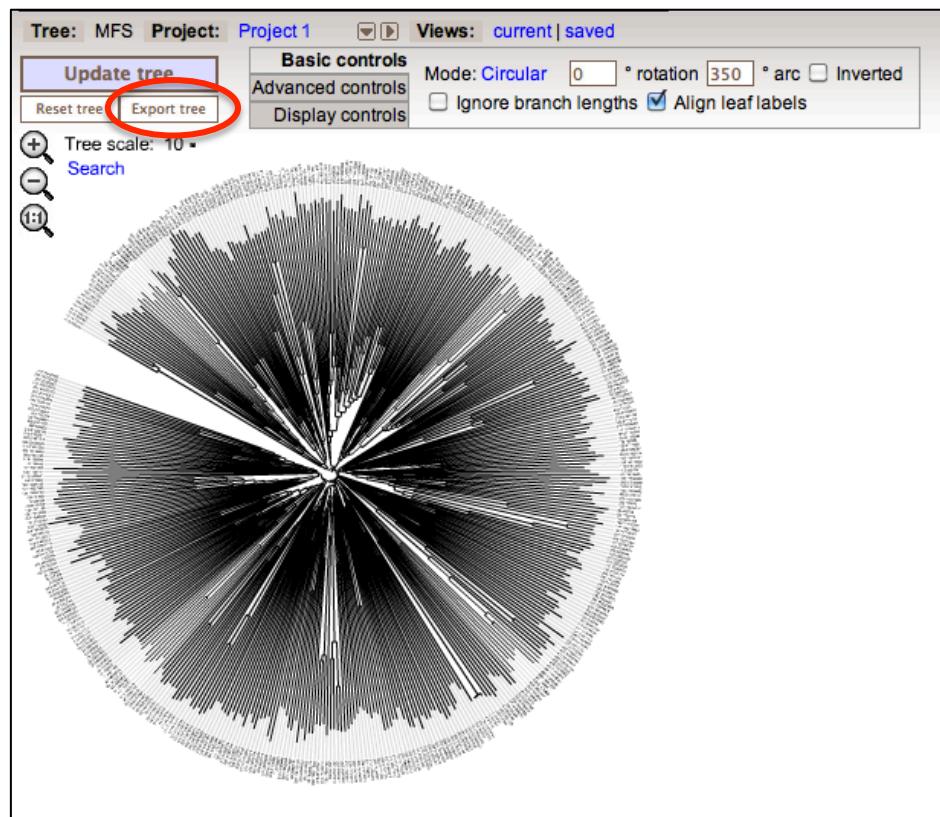
Edit the individual labels and click 'Submit changes' when finished. Use the letter shortcuts to quickly find the individual entries.

Leaf ID	Label
0	top
0_1PW4A	0_1PW4A
1	top
100_167008994	100_167008994
101_2829802	101_2829802
102_74957183	102_74957183
103_74947906	103_74947906
104_2498056	104_2498056
105_48428688	105_48428688
106_82184762	106_82184762
107_187663979	107_187663979
108_74938574	108_74938574
109_13634050	109_13634050
10_74627205	10_74627205

100

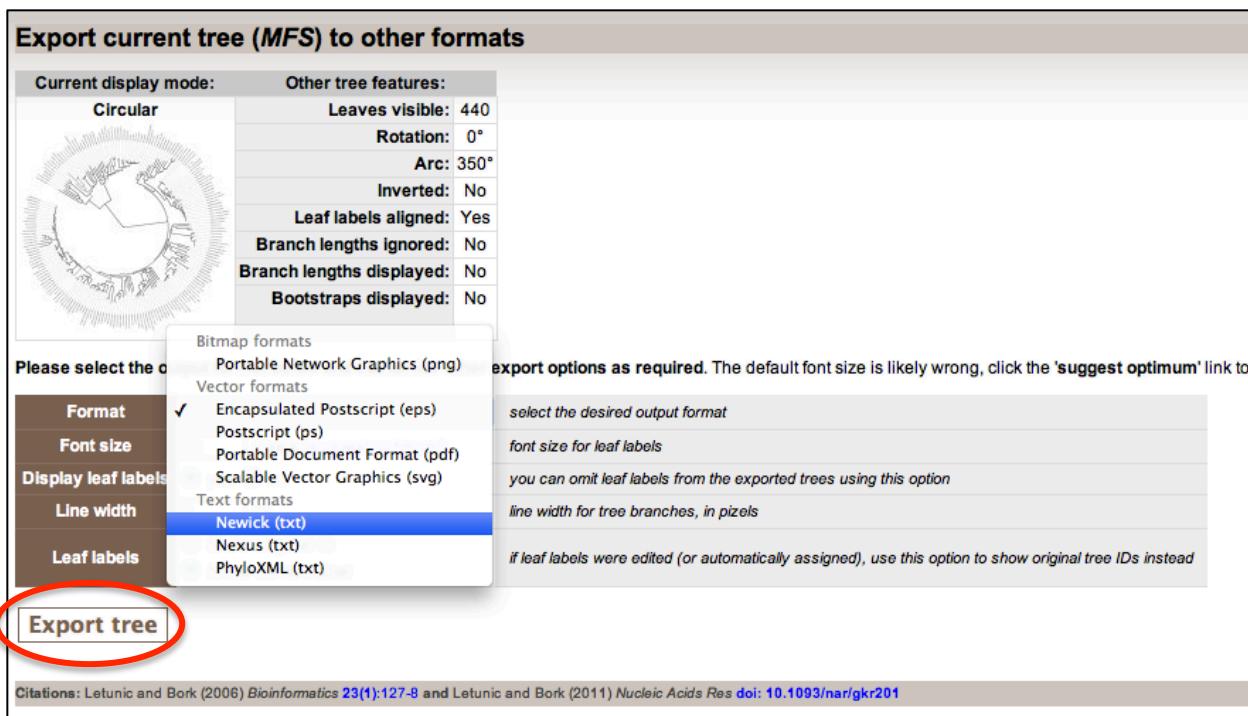
101

iv. Return to the tree view to export the tree with the newly assigned labels.



Supplementary Material

102 Export the tree in Newick format and save as *MFS_UID.tree*.



103

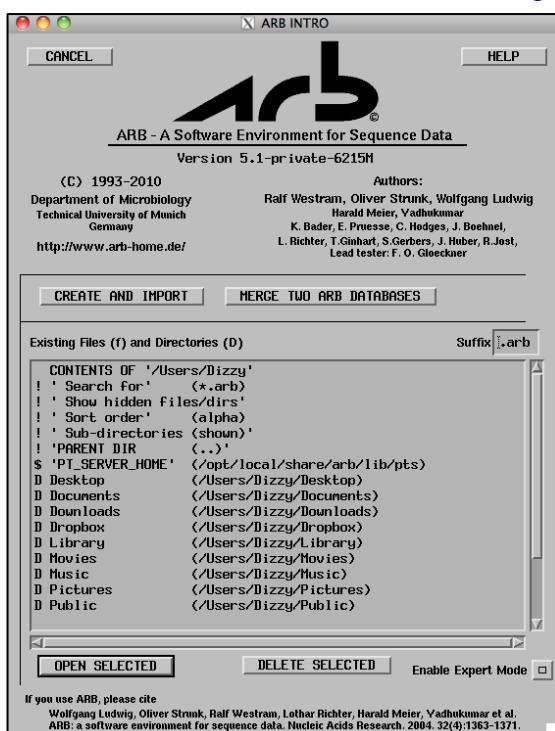
104

105 9. At this point we have four files and are ready to import them into ARB:

- Tree file with unique IDs: *MFS_UID.tree*
- Alignment file with unique IDs: *MFS_UID.fasta*
- Database file containing sequences with meta-data: *MFS_metaData.txt*
- ARB import filter: *MFS_import_filter.ift*

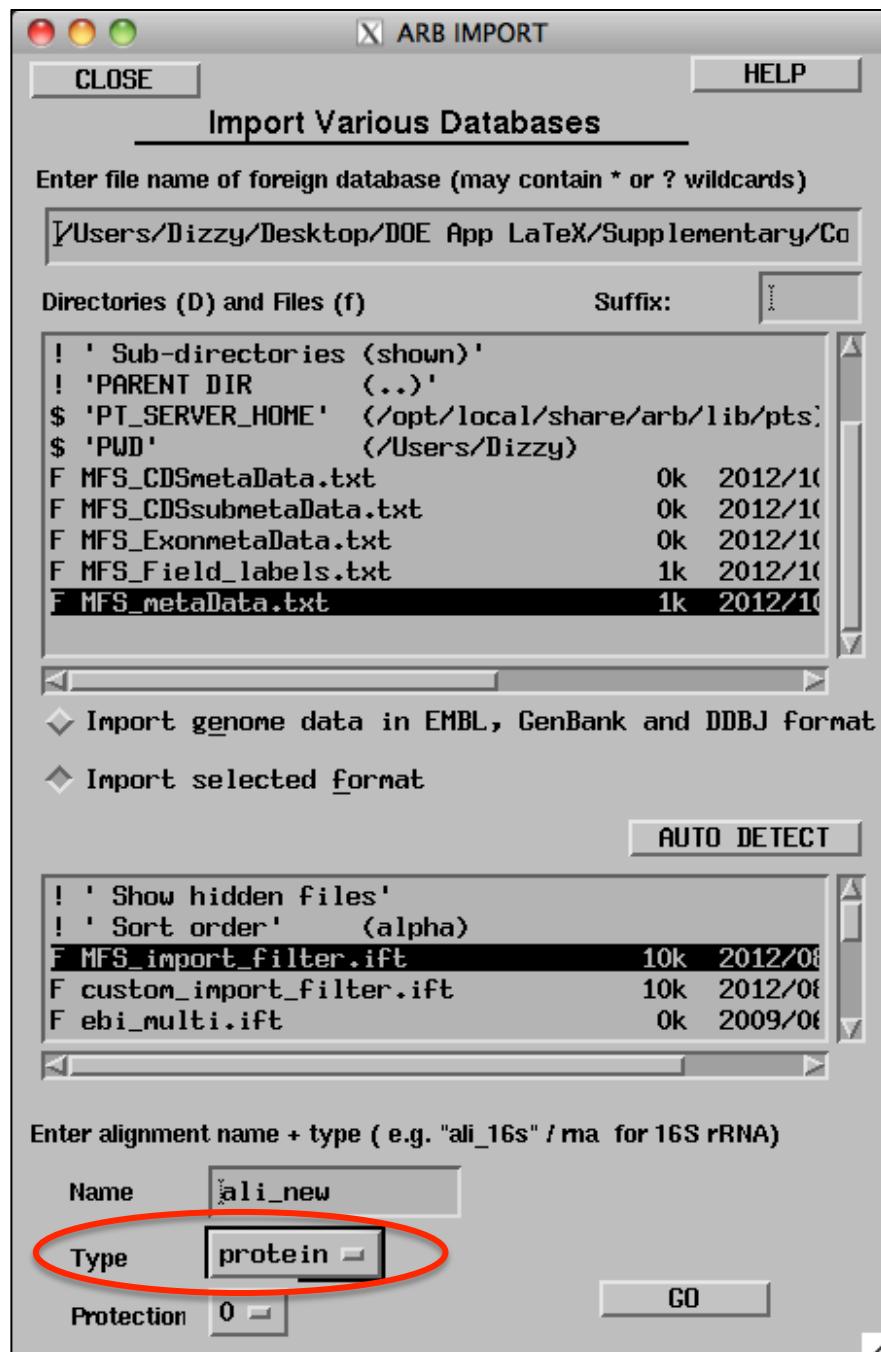
110

111 10. Start ARB from the command line. ARB is available at: <http://www.arb-home.de/>.



Supplementary Material

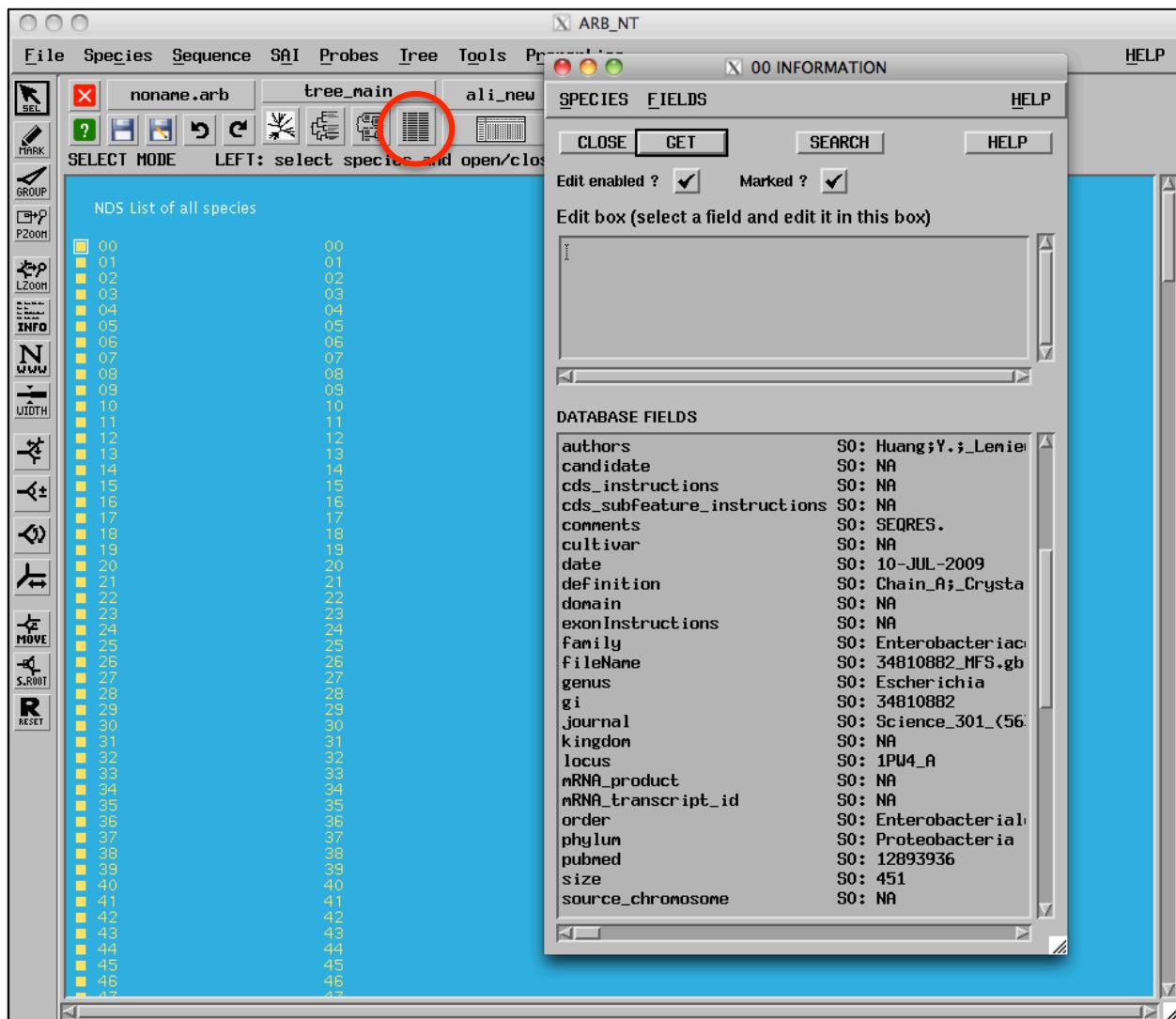
112 11. **Create and Import** the ARB database. Navigate to the directory containing the sequences
113 and metadata .../MFS/Output/. Choose the **MFS_metaData.txt** file and use the
114 **MFS_import_filter.ift** created in step 7. Change Type to Protein.
115 Note: If prompted, select 'Format (all)'. Also, if prompted choose 'Use found names'.
116



117
118
119
120
121

Supplementary Material

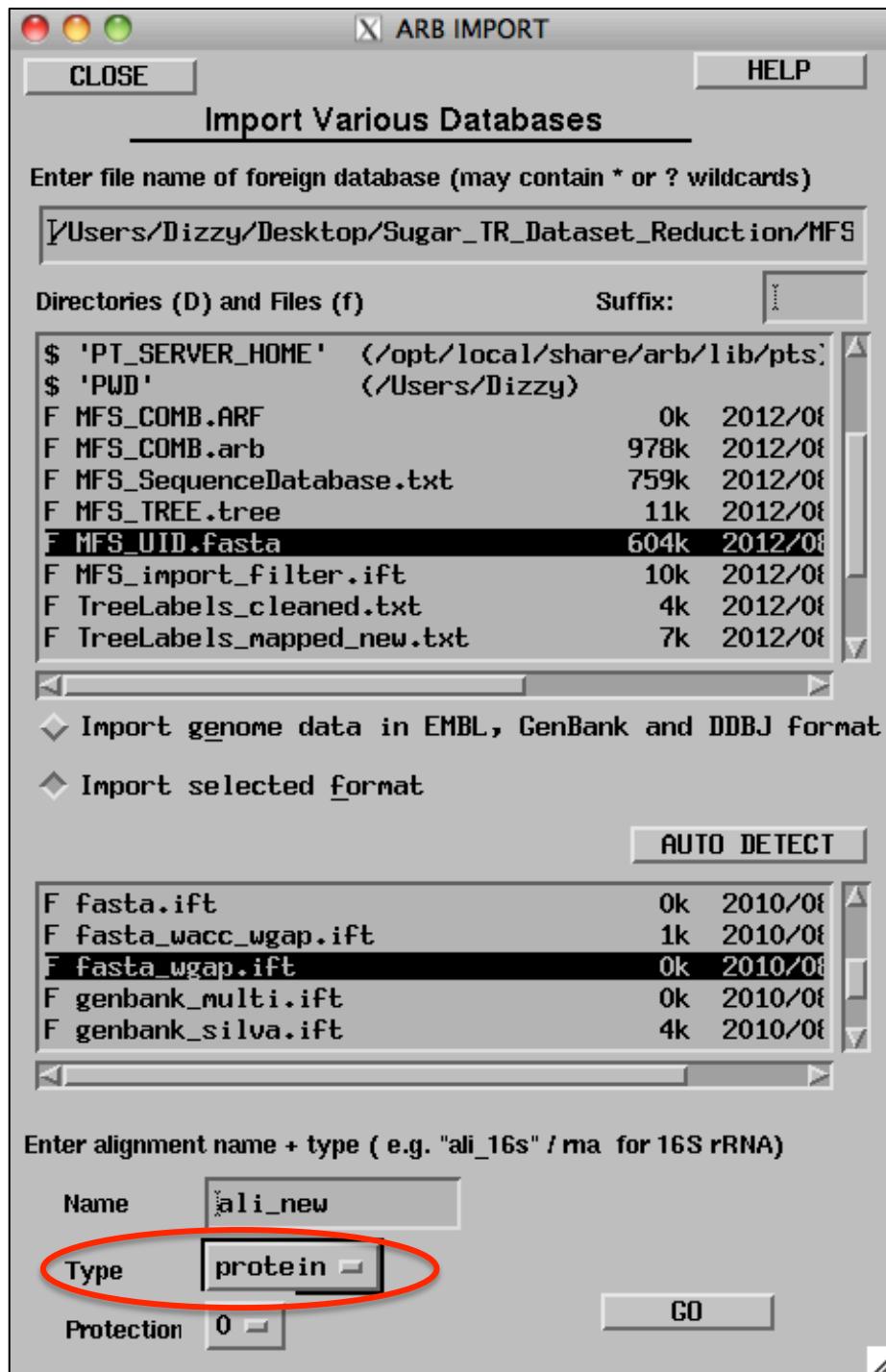
- 122 12. Verify the sequences/meta-data has been imported. Save the database as **DB_Meta.arb**.
123 Close ARB.
124 *Note: You may need to select the list-view button to observe the list of all species.*



125
126
127
128
129
130
131
132

Supplementary Material

- 133 13. Open ARB. Import the alignment file. Use the *fasta_wgap.ift* filter this time. Change
134 Type to Protein.
135 Note: If prompted, select 'Format (all)'. Also, if prompted choose 'Use found names'.



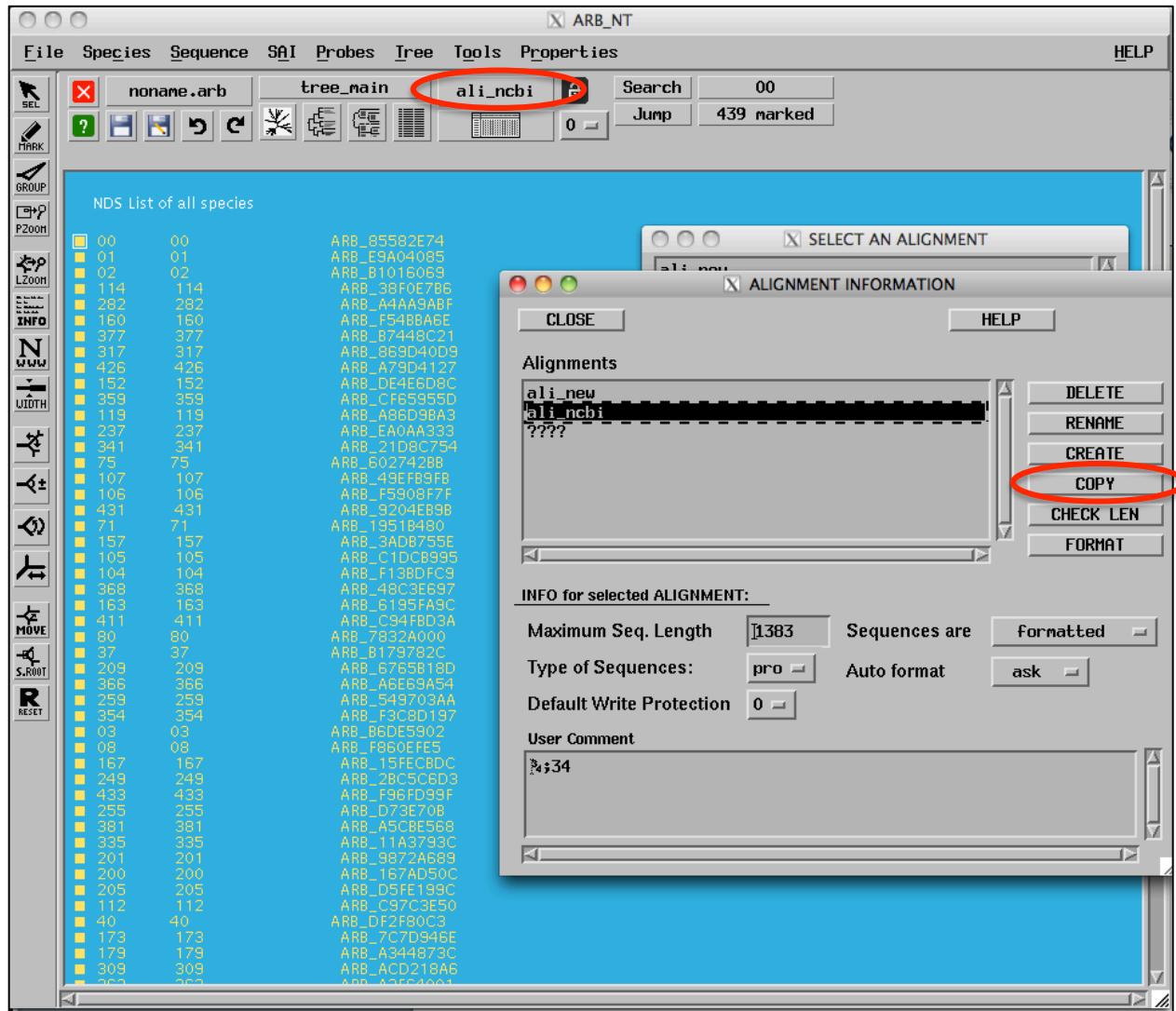
136
137

138

Supplementary Material

139

14. Create a new alignment field by copying *ali_new*. Name the new field *ali_ncbi*.



140

141 15. Save the database as *DB_Aln.arb*.

142

143

144

145

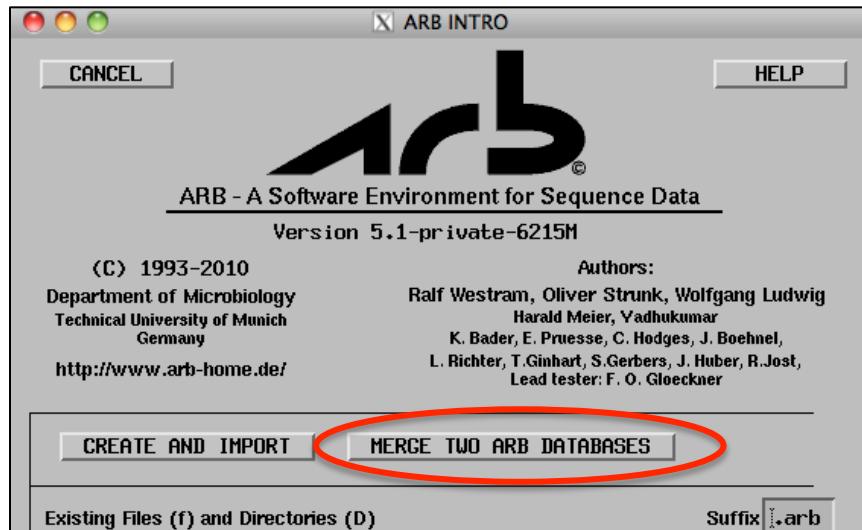
146

147

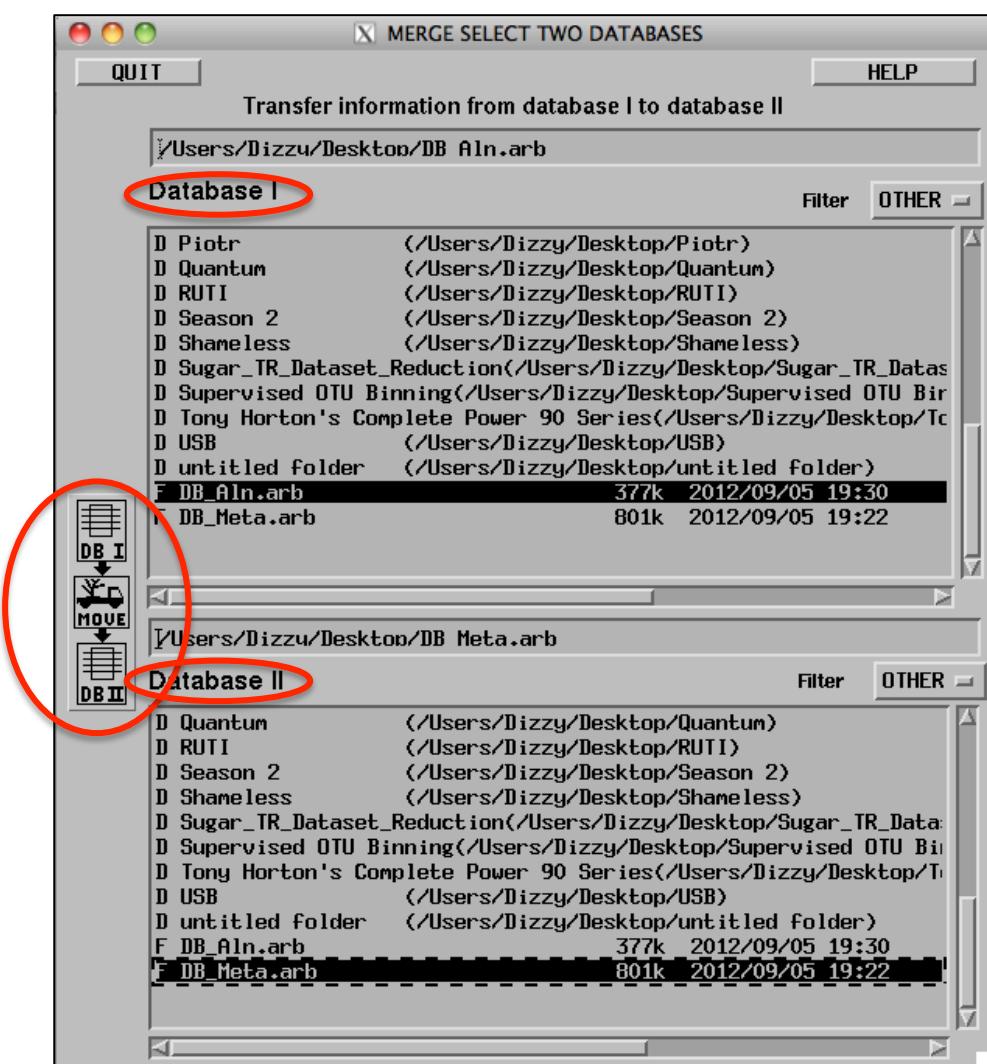
148

Supplementary Material

- 149 16. Merge both ARB databases so that all meta-data, sequences and alignments are linked.
150 a. Open ARB and select the merge database feature.
151 i. Set database I to **DB_Aln.arb**
152 ii. Set database II to **DB_Meta.arb**



153

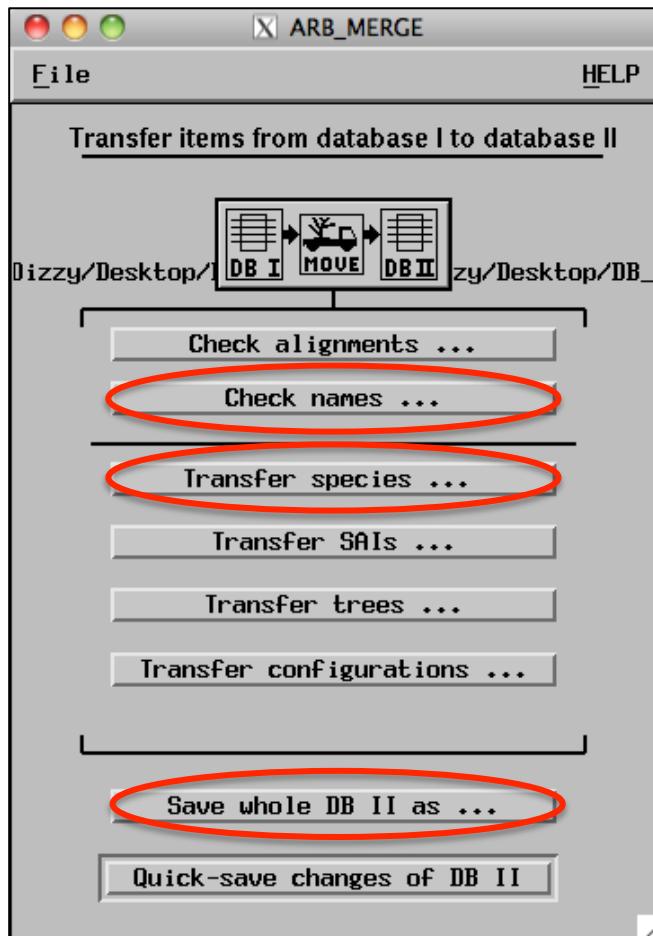


Supplementary Material

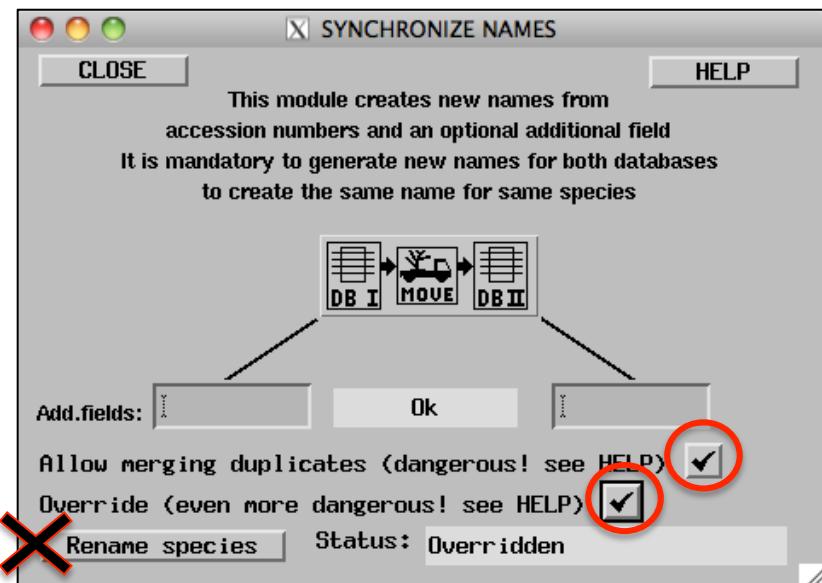
154
155
156
157

- b. Check names...
 - i. Check Allow Merging
 - ii. Check Override

Note: Do NOT click the Rename Species button.

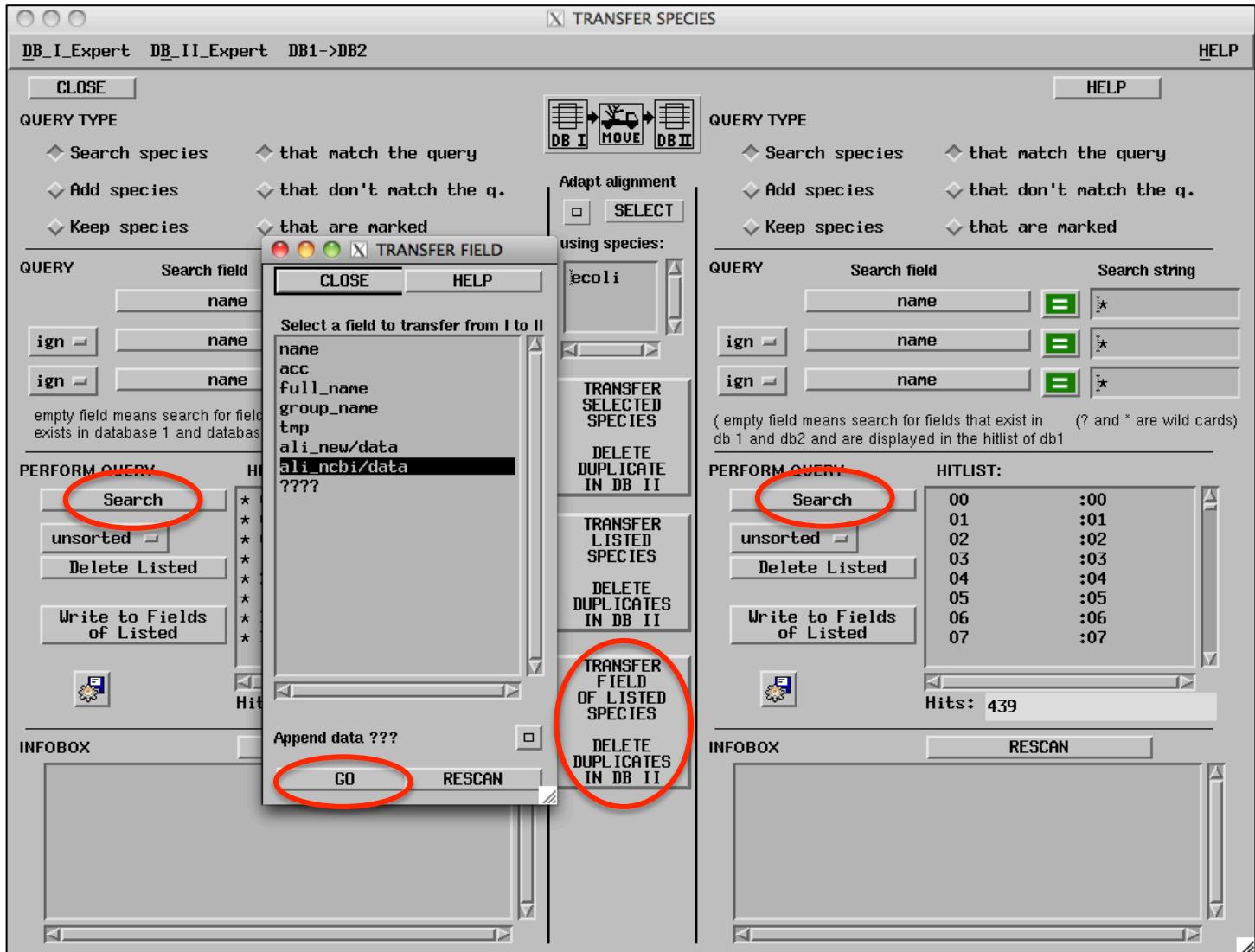


158



Supplementary Material

- 159 c. Transfer Species...
- 160 i. Mark All Species in both DBI and DBII (via Search)
- 161 ii. Click Transfer field of listed species...
- 162 iii. Select name of alignment to transfer: *aln_ncbi/data*
- 163 iv. Click Go.



- 164
- 165 d. Close Transfer Species. Save the merged database as *MFS_COMB.arb*.

166

167

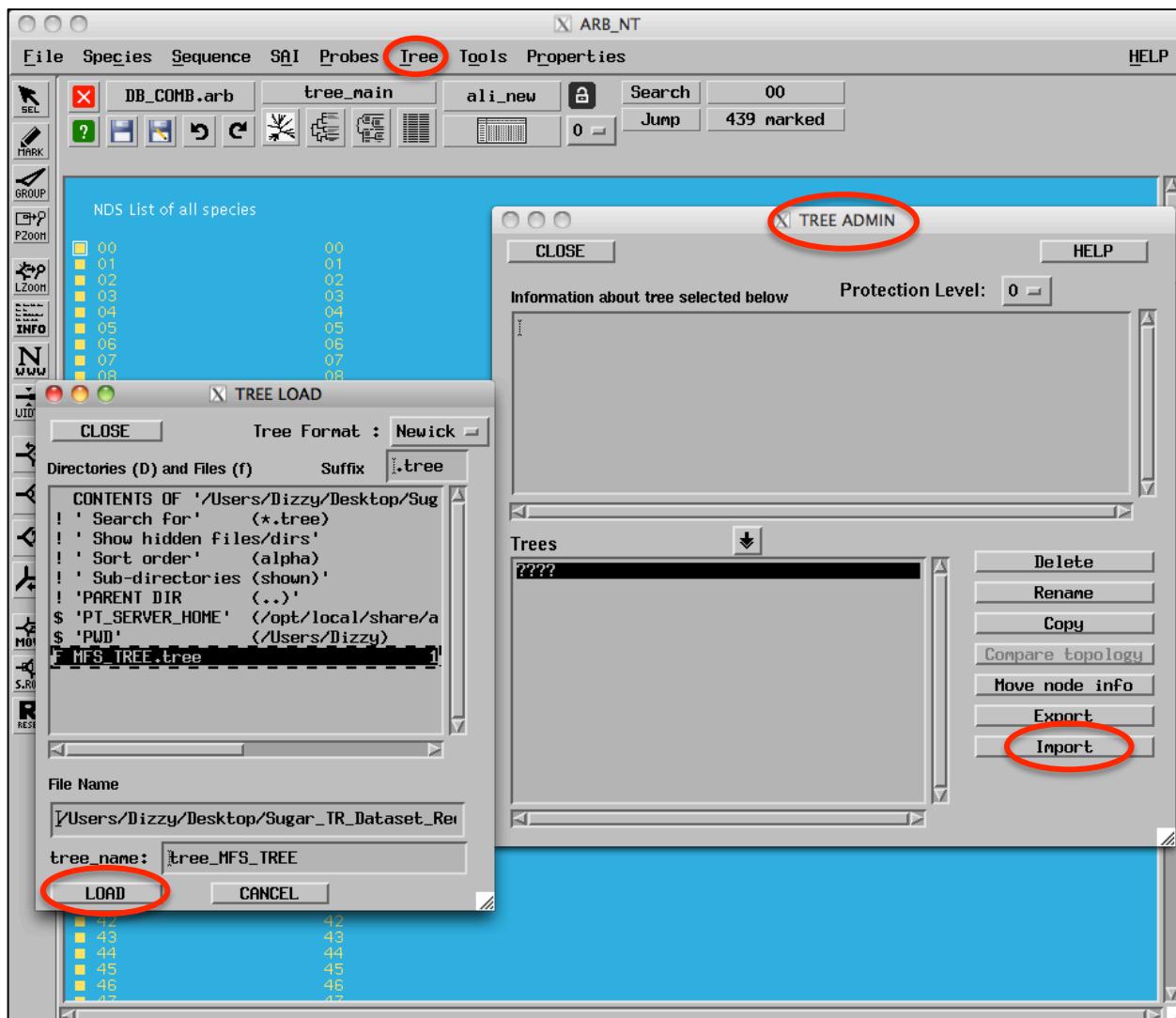
168

169

170

Supplementary Material

- 171 17. Import tree to ARB
172 a. Open ARB and the MFS_COMB database.
173 b. Select Tree/Tree Admin/
174 c. Import MFS_UID.tree from step 8.

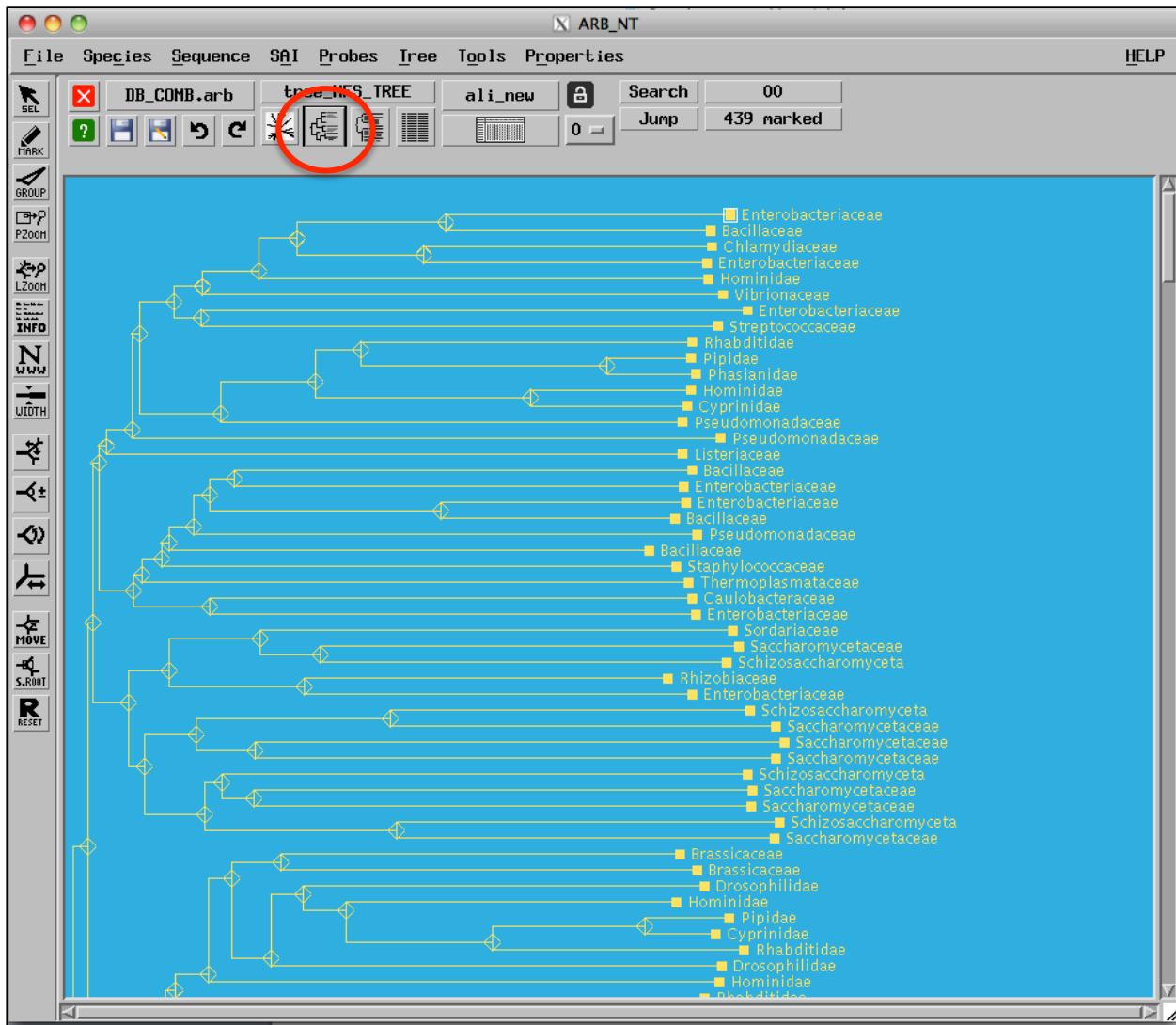


- 176
177
178 18. Save the database again so the tree is stored. The import process is complete!

179
180
181
182
183

Supplementary Material

184 Tree view using meta-label ***Family*** to investigate taxonomic relationship of leaves on tree. The
185 leaf labels may be changed to display any of the meta-data choices via the Node Display Setup
186 (NDS) under the Tree menu. To view different leaf labels on the tree go to: Tree → NDS (Node
187 Display Setup).



188
189
190
191
192

Supplementary Material

- 193 **Appendix A - Renaming Tree Labels**
- 194
- 195 The tree produced via the MFS CDD database contains leaves that are named using an index
196 followed by an underscore followed by the identifier displayed on the tree (e.g. 2_2V8N|A).
197 Note that the index is hidden when viewing the tree in *CDTree*. We need to rename the leaves so
198 that they match the unique identifier in our custom database (i.e. MFS_metaData.txt). This is
199 necessary so that we can link the tree with the alignment and the meta-data once they are all
200 imported into ARB.
- 201
- 202 **Renaming the tree leaves:**
- 203 1. Navigate to the ‘Editing Leaf Labels for Tree MFS’ page on the iTOL website as
204 described in step 8, b, iii.
- 205 2. Highlight, then copy the current leaf labels on this page to a text file. Manually inspect
206 this file to ensure that only leaf names have been copied. Delete anything else that was
207 added to the file. Save this file as *TreeLabels_Orig.txt*.
- 208 3. The MFS CDD dataset has a slight naming inconsistency between the alignment
209 identifiers and the tree leaves. The tree does not contain the ‘|’ character, but for parsing
210 purposes this must be inserted into the *TreeLabels_Orig.txt* file so that it is consistent
211 with the alignment file. There are five instances where ‘|’ must be inserted:
212 0_1PW4|A 1_2GFP|A 2_2V8N|A 3_2CFQ|A 4_1PV7|A
- 213
- 214 4. Place *TreeLabels_Orig.txt*, *MFS_UID.fasta* and *MFS_Align.fasta* in the same directory
215 as *rename_tree_leaves.py*. Run the *rename_tree_leaves.py* script to produce the new tree
216 labels file, *TreeLabels_Mapped_New.txt*. This file is formatted so that each line specifies
217 one leaf name: the original followed by a tab and then the new one (i.e. the unique ID).
- 218 5. Use the *TreeLabels_Mapped_New.txt* as directed in step 8, b, iii.
- 219