User manual Appendix:

Matching name records with WFO using OpenRefine

Version: 29 Nov. 2023

Walter Berendsohn

Contents

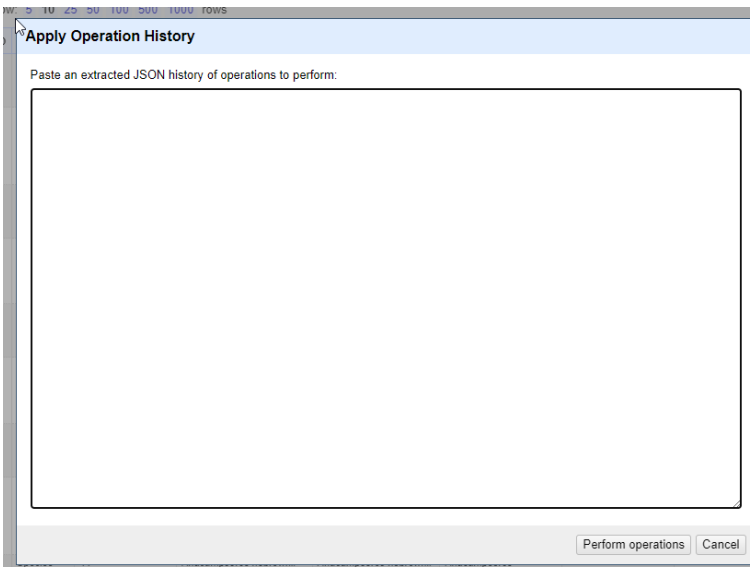
1 Data preparation 1
2 Processing with OpenRefine 1
3 Appendix: .json scripts 5
CDML-CreateStandardisedTable.json (no author corrections) 5
CDML-CreateStandardisedTable4EuroMed.json (caveat: autonyms) 5
CDML-ReconcileWithWFO.json 5
Complete script to reconcile Caryophyllales names 6

1 Data preparation

- Export selected subtree in CDM-light format (Option w/o HTML)
- (Optional: Make a copy of the table scientificName;
  Open that copy of the table in a spreadsheet program (Excel, but better: LibreOffice Calc); make sure that the delimiter is set to comma only, and the character set to UTF-8;
  To simplify further processing delete the following columns: LSID and all from BasionymExAuthorTeam_Fk onwards except AuthorTeamString and FullNameWithRef and save the file in text (csv) format;
  Delete rows with pseudotaxa and other unwanted data records

2 Processing with OpenRefine

- Load the table into OpenRefine creating a new project
- The following steps can be executed by pasting the .json code in the appendix into OpenRefine (left panel, “Undo/Redo”, click “Apply…”, paste code and click “Perform operations”)



- The json code will
  - Standardise author abbreviations that do not follow the TDWG Standard (e.g. Tropicos style with space between initials and last name, Euro+Med with spaces everywhere)[1]
  - Create the to-be matched name string from nameTitleCache replacing authorshipCache with the standardAuthor[2] (this should also correctly treat autonyms)

[1] OpenRefine create column standardAuthorTeamString based on authorTeamString: value.replace(". ",".").replace(".&",". &").replace(".ex",". ex")
[2] value.replace(cells["AuthorTeamString"].value,cells["StandardAuthorTeamString"].value)

- The resulting table can be exported and uploaded to the WFO Plant List name matching tool (https://list.worldfloraonline.org/matching.php)

Instead of exporting and uploading a file, the WFO reconciliation service can be used to directly integrate the matching results with the data in OpenRefine.

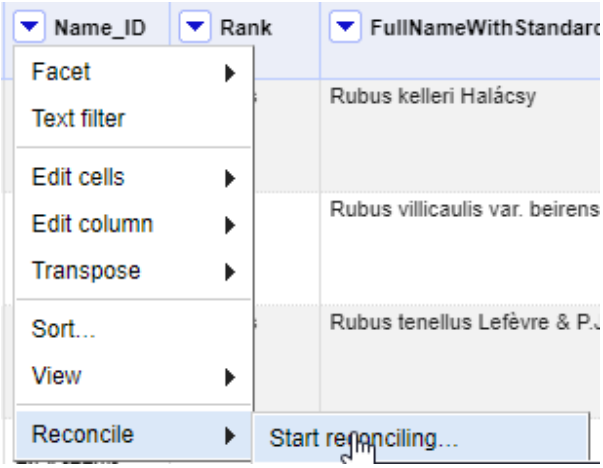
For this, use the .json script CDML-Create column StandardAuthorTeamString at index 17 based on column AuthorTeamString using expression grel:value.replace(". ",".").replace(".&",". &").replace(".ex",". ex")

Create column FullNameWithStandardAuthors1 at index 5 based on column FullNameWithAuthors using expression grel:value.replace(cells["AuthorTeamString"].value,cells["StandardAuthorTeamString"].value)

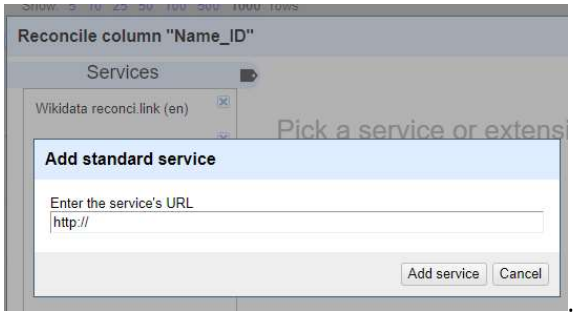
Create column FullNameWithStandardAuthors at index 6 based on column FullNameWithStandardAuthors1 using expression grel:value.replace("subg.","subgen.")


```
[{"op": "core/column-addition","engineConfig": {"facets": [],"mode": "row-based"},"baseColumnName": "AuthorTeamString","expression": "grel:value.replace(\". \",\".\").replace(\".&\",\". &\").replace(\".ex\",\". ex\")","onError": "set-to-blank","newColumnName": "StandardAuthorTeamString","columnInsertIndex": 17,"description": "Create column StandardAuthorTeamString at index 17 based on column AuthorTeamString using expression grel:value.replace(\". \",\".\").replace(\".&\",\". &\").replace(\".ex\",\". ex\")"},{"op": "core/column-addition","engineConfig": {"facets": [],"mode": "row-based"},"baseColumnName": "FullNameWithAuthors","expression": "grel:value.replace(cells[\"AuthorTeamString\"].value,cells[\"StandardAuthorTeamString\"].value)","onError": "set-to-blank","newColumnName": "FullNameWithStandardAuthors1","columnInsertIndex": 5,"description": "Create column FullNameWithStandardAuthors1 at index 5 based on column FullNameWithAuthors using expression grel:value.replace(cells[\"AuthorTeamString\"].value,cells[\"StandardAuthorTeamString\"].value)"},{"op": "core/column-addition","engineConfig": {"facets": [],"mode": "row-based"},"baseColumnName": "FullNameWithStandardAuthors1","expression": "grel:value.replace(\"subg.\",\"subgen.\")","onError": "set-to-blank","newColumnName": "FullNameWithStandardAuthors","columnInsertIndex": 6,"description": "Create column FullNameWithStandardAuthors at index 6 based on column FullNameWithStandardAuthors1 using expression grel:value.replace(\"subg.\",\"subgen.\")"}]
```


CDML-ReconcileWithWFO.json in the appendix after the last step in the previous script – but before doing that, you have to add the respective service to OpenRefine. For that, use the menu at any column:

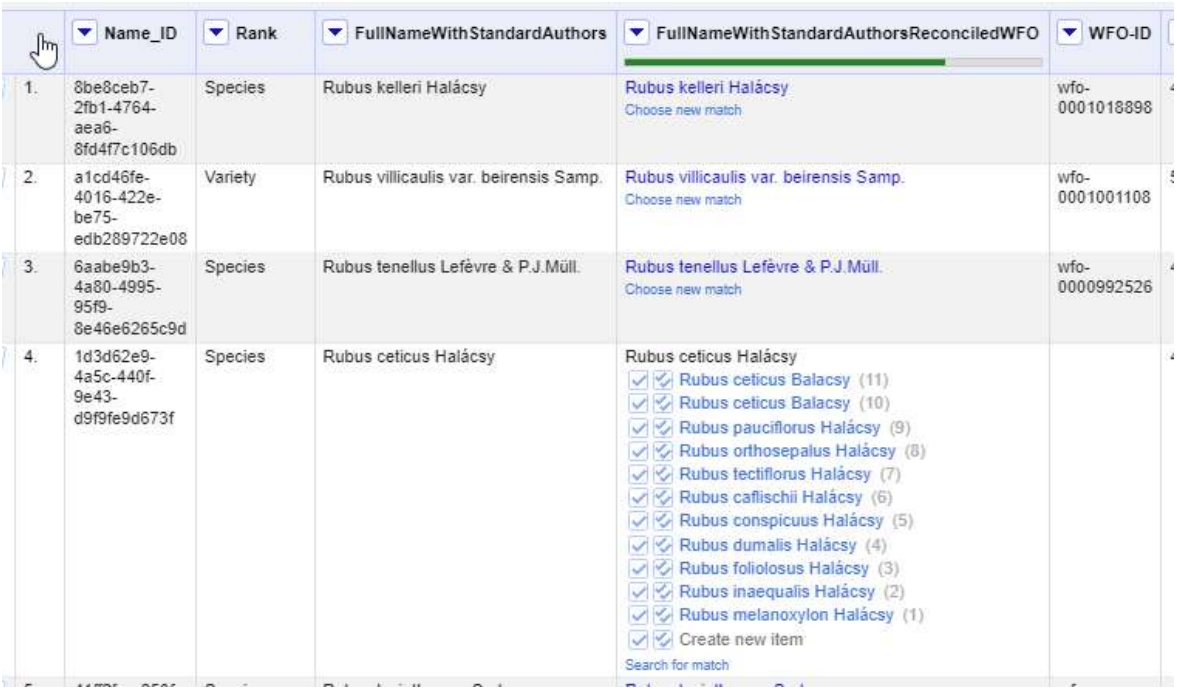


and add the WFO service (URL: https://list.worldfloraonline.org/reconcile):



After having clicked Add service you can Cancel the process – your OpenRefine installation will remember the services.

The advantage is that both, exact matches and candidates are seen together:



	Name_ID	Rank	FullNameWithStandardAuthors	FullNameWithStandardAuthorsReconciledWFO	WFO-ID
1.	8be8ceb7-2fb1-4764-aea6-8fd4f7c106db	Species	Rubus kelleri Halácsy	Rubus kelleri Halácsy Choose new match	wfo-0001018898
2.	a1cd46fe-4016-422e-be75-edb289722e08	Variety	Rubus villicaulis var. beirensis Samp.	Rubus villicaulis var. beirensis Samp. Choose new match	wfo-0001001108
3.	6aabe9b3-4a80-4995-95f9-8e46e6265c9d	Species	Rubus tenellus Lefèvre & P.J.Müll.	Rubus tenellus Lefèvre & P.J.Müll. Choose new match	wfo-0000992526
4.	1d3d62e9-4a5c-440f-9e43-d9f9fe9d673f	Species	Rubus ceticus Halácsy	Rubus ceticus Halácsy Rubus ceticus Balacsy (11) Rubus ceticus Balacsy (10) Rubus pauciflorus Halácsy (9) Rubus orthosepalus Halácsy (8) Rubus tectiflorus Halácsy (7) Rubus caflischii Halácsy (6) Rubus conspicuus Halácsy (5) Rubus dumalis Halácsy (4) Rubus foliolosus Halácsy (3) Rubus inaequalis Halácsy (2) Rubus melanoxylon Halácsy (1) Create new item Search for match	

Unfortunately, if one of the options is choosen as a match, the corresponding wfo-ID is not shown directly but you can see it in the step that was added to the list on the left. The candidate name itself links to its detail from WFO.

The export from OpenRefine in csv format does not include the candidate list. For those records that could actually be resolved in the list, a comment can be added in the wfo-ID; this is included in the

output when exported from OpenRefine:

Rubus latifrons (Progel) Hayek	Rubus latifrons Hayek Choose new match	resolved, same publication, basionym author missing in WFO	4
Rubus cordifolius Weihe & Nees	Rubus cordifolius Weihe & Nees Choose new match	wfo-0001018647	4
Rubus renifrons Sabr.	Rubus renifrons Sabr.	wfo-	4

3 Appendix: .json scripts

CDML-CreateStandardisedTable.json (no author corrections)

Create column FullNameWithStandardAuthors at index 5 based on column FullNameWithAuthors using expression grel:value.replace("subg.","subgen.")


```
[{"op": "core/column-addition","engineConfig": {"facets": [{"type": "list","name":
"WFO-ID","expression": "isBlank(value)","columnName": "WFO-ID","invert":
false,"omitBlank": false,"omitError": false,"selection": [{"v": {"v": true,"l":
"true"}}],"selectBlank": false,"selectError": false}],"mode": "row-
based"},"baseColumnName": "FullNameWithAuthors","expression":
"grel:value.replace(\"subg.\",\"subgen.\")","onError": "set-to-blank","newColumnName":
"FullNameWithStandardAuthors","columnInsertIndex": 5,"description": "Create column
FullNameWithStandardAuthors at index 5 based on column FullNameWithAuthors using
expression grel:value.replace(\"subg.\",\"subgen.\")"}]
```


CDML-CreateStandardisedTable4EuroMed.json (caveat: autonyms)

Create column StandardAuthorTeamString at index 17 based on column AuthorTeamString using expression grel:value.replace(". ",".").replace(".&",". &").replace(".ex",". ex")

Create column FullNameWithStandardAuthors1 at index 5 based on column FullNameWithAuthors using expression grel:value.replace(cells["AuthorTeamString"].value,cells["StandardAuthorTeamString"].value)

Create column FullNameWithStandardAuthors at index 6 based on column FullNameWithStandardAuthors1 using expression grel:value.replace("subg.","subgen.")


```
[{"op": "core/column-addition","engineConfig": {"facets": [],"mode": "row-
based"},"baseColumnName": "AuthorTeamString","expression": "grel:value.replace(\".
\",\".\").replace(\".&\",\". &\").replace(\".ex\",\". ex\")","onError": "set-to-
blank","newColumnName": "StandardAuthorTeamString","columnInsertIndex":
17,"description": "Create column StandardAuthorTeamString at index 17 based on column
AuthorTeamString using expression grel:value.replace(\". \",\".\").replace(\".&\",\".
&\").replace(\".ex\",\". ex\")"},{"op": "core/column-addition","engineConfig":
{"facets": [],"mode": "row-based"},"baseColumnName":
"FullNameWithAuthors","expression":
"grel:value.replace(cells[\"AuthorTeamString\"].value,cells[\"StandardAuthorTeamString\
"].value)","onError": "set-to-blank","newColumnName":
"FullNameWithStandardAuthors1","columnInsertIndex": 5,"description": "Create column
FullNameWithStandardAuthors1 at index 5 based on column FullNameWithAuthors using
expression
grel:value.replace(cells[\"AuthorTeamString\"].value,cells[\"StandardAuthorTeamString\"
].value)"},{"op": "core/column-addition","engineConfig": {"facets": [],"mode": "row-
based"},"baseColumnName": "FullNameWithStandardAuthors1","expression":
"grel:value.replace(\"subg.\",\"subgen.\")","onError": "set-to-blank","newColumnName":
"FullNameWithStandardAuthors","columnInsertIndex": 6,"description": "Create column
FullNameWithStandardAuthors at index 6 based on column FullNameWithStandardAuthors1
using expression grel:value.replace(\"subg.\",\"subgen.\")"}]
```


CDML-ReconcileWithWFO.json


```
[  {    "op": "core/column-addition",    "engineConfig": {      "facets": [],
"mode": "row-based"    },    "baseColumnName": "FullNameWithStandardAuthors",
"expression": "grel:value",    "onError": "set-to-blank",    "newColumnName":
"FullNameWithStandardAuthorsReconciledWFO",    "columnInsertIndex": 3,
"description": "Create column FullNameWithStandardAuthorsReconciledWFO at index 3 based
```

```
on column FullNameWithStandardAuthors using expression grel:value"  },  {    "op":
"core/recon",    "engineConfig": {      "facets": [],      "mode": "row-based"    },
"columnName": "FullNameWithStandardAuthorsReconciledWFO",    "config": {      "mode":
"standard-service",      "service": "https://list.worldfloraonline.org/reconcile",
"identifierSpace": "https://list.worldfloraonline.org/",      "schemaSpace":
"https://list.worldfloraonline.org/terms/",      "type": {        "id": "TaxonName",
"name": "Taxonomic Name in the ICNABP"      },      "autoMatch": true,
"columnDetails": [],      "limit": 10    },    "description": "Reconcile cells in
column FullNameWithStandardAuthorsReconciledWFO to type TaxonName"  },  {    "op":
"core/column-addition",    "engineConfig": {      "facets": [],      "mode": "row-
based"    },    "baseColumnName": "FullNameWithStandardAuthorsReconciledWFO",
"expression": "cell.recon.match.id",    "onError": "set-to-blank",    "newColumnName":
"WFO-ID",    "columnInsertIndex": 4,    "description": "Create column WFO-ID at index 4
based on column FullNameWithStandardAuthorsReconciledWFO using expression
cell.recon.match.id"  }]
```


Complete script to reconcile Caryophyllales names

Create column FullNameWithStandardAuthors at index 5 based on column FullNameWithAuthors using expression grel:value.replace("subg.","subgen.")

Create column FullNameWithStandardAuthorsReconciledWFO at index 3 based on column FullNameWithStandardAuthors using expression grel:value

Reconcile cells in column FullNameWithStandardAuthorsReconciledWFO to type TaxonName

Create column WFO-ID at index 4 based on column FullNameWithStandardAuthorsReconciledWFO using expression cell.recon.match.id


```
[ { "op": "core/column-addition", "engineConfig": { "facets": [ { "type": "list",
"name": "WFO-ID", "expression": "isBlank(value)", "columnName": "WFO-ID", "invert":
false, "omitBlank": false, "omitError": false, "selection": [ { "v": { "v": true, "l":
"true" } } ], "selectBlank": false, "selectError": false } ], "mode": "row-based" },
"baseColumnName": "FullNameWithAuthors", "expression":
"grel:value.replace(\"subg.\",\"subgen.\")", "onError": "set-to-blank",
"newColumnName": "FullNameWithStandardAuthors", "columnInsertIndex": 5, "description":
"Create column FullNameWithStandardAuthors at index 5 based on column
FullNameWithAuthors using expression grel:value.replace(\"subg.\",\"subgen.\")" }, {
"op": "core/column-addition", "engineConfig": { "facets": [], "mode": "row-based" },
"baseColumnName": "FullNameWithStandardAuthors", "expression": "grel:value", "onError":
"set-to-blank", "newColumnName": "FullNameWithStandardAuthorsReconciledWFO",
"columnInsertIndex": 3, "description": "Create column
FullNameWithStandardAuthorsReconciledWFO at index 3 based on column
FullNameWithStandardAuthors using expression grel:value" }, { "op": "core/recon",
"engineConfig": { "facets": [], "mode": "row-based" }, "columnName":
"FullNameWithStandardAuthorsReconciledWFO", "config": { "mode": "standard-service",
"service": "https://list.worldfloraonline.org/reconcile", "identifierSpace":
"https://list.worldfloraonline.org/", "schemaSpace":
"https://list.worldfloraonline.org/terms/", "type": { "id": "TaxonName", "name":
"Taxonomic Name in the ICNABP" }, "autoMatch": true, "columnDetails": [], "limit": 10
}, "description": "Reconcile cells in column FullNameWithStandardAuthorsReconciledWFO
to type TaxonName" }, { "op": "core/column-addition", "engineConfig": { "facets": [],
"mode": "row-based" }, "baseColumnName": "FullNameWithStandardAuthorsReconciledWFO",
"expression": "cell.recon.match.id", "onError": "set-to-blank", "newColumnName": "WFO-
ID", "columnInsertIndex": 4, "description": "Create column WFO-ID at index 4 based on
column FullNameWithStandardAuthorsReconciledWFO using expression cell.recon.match.id" }
]
```