

## EDIT - bug #9668

### Secundum reference is switched when swap synonym and accepted taxon

06/15/2021 09:36 AM - Katja Luther

<b>Status:</b>	In Progress	<b>Start date:</b>	
<b>Priority:</b>	Highest	<b>Due date:</b>	
<b>Assignee:</b>	Katja Luther	<b>% Done:</b>	0%
<b>Category:</b>	cdmlib	<b>Estimated time:</b>	0:00 hour
<b>Target version:</b>	Release 5.45	<b>Found in Version:</b>	
<b>Severity:</b>	normal		

#### Description

mail NadjaK:

ich bin gerade auf eine Fehlerquelle gestoßen:

Ich habe *Salmonopuntia schickendantzii* als neues Synonym hinzugefügt, und das Secundum ist Köhler et al. Dieser Name soll auch der neue akzeptierte werden.

The screenshot shows a web interface for a taxonomic database. On the left, a list of synonyms for *Brasiliopuntia schickendantzii* (F.A.C.Weber) R.Puente & Majure in Succ. Pl. Res. 8: 25. 2014 is displayed. The selected synonym is *Salmonopuntia schickendantzii* (F.A.C.Weber) Font & M.Köhler in Phytotaxa 505(3): 267. 2021. On the right, a details panel for this synonym is shown, including a title cache, an appended phrase (Secundum) with the text: "Köhler, M., Font, F., Puente-Martinez, R. & Majure, L.C. 2021: 'That's Opuntia, that was!', again: a new combination for an old and enigmatic Opuntia s.l. (Cactaceae). – Phytotaxa 505(3): 262–274", and a section for links with checkboxes for 'Exclude Authorship', 'Synonym is doubtful', and 'Synonym is published' (checked).

Aber, wenn ich das mit dem vorher akzeptierten Namen tausche, werden auch die Secundum-Referenzen getauscht, und zwar ohne Warnung.

*Salmonopuntia schickendantzii* bekommt dann das Secundum Kiesling et al. – also das des vorher akzeptierten Taxons, während *Brasiliopuntia schickendantzii* als Synonym das Secundum Köhler et al. bekommt. Und beides ist natürlich falsch.

Ich habe das mehrmals ausprobiert, mit demselben Ergebnis. kann mich nicht erinnern, dass das früher auch schon so war, ist das vielleicht ein Bug? Das ist natürlich eine Riesen-Fehlerquelle. Schaut Euch das mal an.

**Salmonopuntia schickendantzii (F.A.C.Weber) Font & M.Köhler in Phytotaxa...**

- Salmonopuntia schickendantzii (F.A.C.Weber) Font & M.Köhler in Phytotaxa 505(3): 267. 2021
  - Opuntia schickendantzii F.A.C.Weber in Bois, Diet. Hort.: 898. 1898
  - Cylindropuntia schickendantzii (F.A.C.Weber) Backeb., Kaktus-ABC: 122. 1936 ["1935"]
  - Austrocylindropuntia schickendantzii (F.A.C.Weber) Backeb. in Cact. Succ. J. (Los Angeles) 23: 14. 1951
  - Brasiliopuntia schickendantzii (F.A.C.Weber) R.Puente & Majure in Succ. Pl. Res. 8: 25. 2014
  - Mortolopuntia schickendantzii (F.A.C.Weber) Guiggi in Cactology 5 (Suppl. I): 1. 2015
- Opuntia cochabambensis Cárdenas in Revista Agric. (Cochabamba) 9: 20. 1955
- Platyopuntia conjungens F.Ritter, Kakteen Südamerika 2: 494. 1980
  - Opuntia conjungens (F.Ritter) P.J.Braun & Esteves in Succulenta (Netherlands) 74(3): 133. 1995

**Details**

**Taxon:** Salmonopuntia schickendantzii (F.A.C.Weber) Font & M.Köhler sec. Kiesling & al. 2014

**Title Cache:** Salmonopuntia schickendantzii (F.A.C.Weber) Font & M.Köhler sec. Kiesling & al. 2014

**Appended Phrase Secundum:** Kiesling, R., Oakley, L., Metzging, D., Muruaga, N., Albesiano, A.S., Korotkova, N., Huaylla Limachi, L. & Quispe, N. 2014: Cactaceae, pp. 443-471. – In: Jørgensen, P.M., Nee, M.H. & Beck, S.G., Catálogo de las plantas vasculares de Bolivia

**Detail:**

**Links:**

Exclude Authorship   
 Taxon is doubtful   
 Taxon is published

**Name:** Salmonopuntia schickendantzii (F.A.C.Weber) Font & M.Köhler

## Associated revisions

### Revision 63004250 - 06/15/2021 09:40 AM - Katja Luther

ref #9668: hotfix for secundum handling when swap syn and accepted

### Revision 5b77dba4 - 06/15/2021 10:08 AM - Katja Luther

ref #9668: hotfix for secundum handling when swap syn and accepted

### Revision c9e4e6f8 - 06/15/2021 02:12 PM - Katja Luther

ref #9668: hotfix for secundum handling when swap syn and accepted - continue

### Revision e13687ce - 07/14/2021 03:44 PM - Andreas Müller

ref #9668 fix CCE due to unordered result.getUpdatedObjects() and NPE due to missing secSources and some cleanup and javadoc

### Revision 9b5e79ee - 08/11/2021 02:48 PM - Katja Luther

ref #9340, #9734, #9668: further improvements for configurable sec handling during taxon(base) operations

### Revision f13ee832 - 08/11/2021 03:08 PM - Katja Luther

ref #9340, #9734, #9668: further improvements for configurable sec handling during taxon(base) operations

### Revision 296fa5af - 08/12/2021 11:36 AM - Katja Luther

ref #9340, #9734, #9668: result should be of type DeleteResult

## History

### #1 - 06/15/2021 09:36 AM - Katja Luther

- Target version changed from Unassigned CDM tickets to Release 5.25

### #2 - 06/15/2021 10:14 AM - Katja Luther

As a workaround the secundum reference is switched in editor for a quick hotfix.

The complete implementation needs to be done in the service layer method.

### #3 - 06/15/2021 10:49 AM - Katja Luther

- Description updated

### #4 - 06/15/2021 11:01 AM - Katja Luther

- Assignee changed from Katja Luther to Andreas Müller

The hotfix is implemented, please review with nightly.

## #5 - 06/15/2021 01:09 PM - Andreas Müller

- Status changed from New to Feedback

- Assignee changed from Andreas Müller to Katja Luther

I get

```
last remote method : http://test.e-taxonomy.eu:80/cdmserver/rem_conf_am/remoting/taxon.service
last remote request client time : 2021-06-15T13:01:22.95
last remote request response header time : Tue, 15 Jun 2021 11:01:23 GMT
client error time : 2021-06-15T13:01:22.981
login : admin
editor version : 5.25.0.202106151047
server : test.e-taxonomy.eu (edit-test) / rem_conf_am
schema version : 5.23.0.0.20210422
os : Windows Server 2012 R2 6.3 amd64
java : 1.8.0_131
org.eclipse.swt.SWTException: Failed to execute runnable (java.lang.ClassCastException: eu.etaxonomy.cdm.model.taxon.Taxon cannot be cast to eu.etaxonomy.cdm.model.taxon.Synonym)
    at org.eclipse.swt.SWT.error(SWT.java:4533)
    at org.eclipse.swt.SWT.error(SWT.java:4448)
    at org.eclipse.swt.widgets.Synchronizer.runAsyncMessages(Synchronizer.java:185)
    at org.eclipse.swt.widgets.Display.runAsyncMessages(Display.java:4211)
    at org.eclipse.swt.widgets.Display.readAndDispatch(Display.java:3827)
    at org.eclipse.jface.window.Window.runEventLoop(Window.java:818)
    at org.eclipse.jface.window.Window.open(Window.java:794)
    at org.eclipse.jface.dialogs.MessageDialog.open(MessageDialog.java:396)
    at org.eclipse.jface.dialogs.MessageDialog.open(MessageDialog.java:424)
    at org.eclipse.jface.dialogs.MessageDialog.openWarning(MessageDialog.java:532)
    at eu.etaxonomy.taxeditor.model.MessagingUtils$4.run(MessagingUtils.java:584)
    at org.eclipse.swt.widgets.RunnableLock.run(RunnableLock.java:35)
    at org.eclipse.swt.widgets.Synchronizer.runAsyncMessages(Synchronizer.java:182)
    at org.eclipse.swt.widgets.Display.runAsyncMessages(Display.java:4211)
    at org.eclipse.swt.widgets.Display.readAndDispatch(Display.java:3827)
    at org.eclipse.e4.ui.internal.workbench.swt.PartRenderingEngine$4.run(PartRenderingEngine.java:1121)
    at org.eclipse.core.databinding.observable.Realm.runWithDefault(Realm.java:336)
    at org.eclipse.e4.ui.internal.workbench.swt.PartRenderingEngine.run(PartRenderingEngine.java:1022)
    at org.eclipse.e4.ui.internal.workbench.E4Workbench.createAndRunUI(E4Workbench.java:150)
    at org.eclipse.ui.internal.Workbench$5.run(Workbench.java:693)
    at org.eclipse.core.databinding.observable.Realm.runWithDefault(Realm.java:336)
    at org.eclipse.ui.internal.Workbench.createAndRunWorkbench(Workbench.java:610)
    at org.eclipse.ui.PlatformUI.createAndRunWorkbench(PlatformUI.java:148)
    at eu.etaxonomy.taxeditor.Application.start(Application.java:20)
    at org.eclipse.equinox.internal.app.EclipseAppHandle.run(EclipseAppHandle.java:196)
    at org.eclipse.core.runtime.internal.adaptor.EclipseAppLauncher.runApplication(EclipseAppLauncher.java:134)
)
    at org.eclipse.core.runtime.internal.adaptor.EclipseAppLauncher.start(EclipseAppLauncher.java:104)
    at org.eclipse.core.runtime.adaptor.EclipseStarter.run(EclipseStarter.java:388)
    at org.eclipse.core.runtime.adaptor.EclipseStarter.run(EclipseStarter.java:243)
    at sun.reflect.NativeMethodAccessorImpl.invoke0(Native Method)
    at sun.reflect.NativeMethodAccessorImpl.invoke(NativeMethodAccessorImpl.java:62)
    at sun.reflect.DelegatingMethodAccessorImpl.invoke(DelegatingMethodAccessorImpl.java:43)
    at java.lang.reflect.Method.invoke(Method.java:498)
    at org.eclipse.equinox.launcher.Main.invokeFramework(Main.java:673)
    at org.eclipse.equinox.launcher.Main.basicRun(Main.java:610)
    at org.eclipse.equinox.launcher.Main.run(Main.java:1519)
Caused by: java.lang.ClassCastException: eu.etaxonomy.cdm.model.taxon.Taxon cannot be cast to eu.etaxonomy.cdm.model.taxon.Synonym
    at eu.etaxonomy.taxeditor.editor.name.operation.SwapSynonymAndAcceptedOperation.execute(SwapSynonymAndAcceptedOperation.java:75)
    at eu.etaxonomy.taxeditor.model.AbstractUtility.lambda$3(AbstractUtility.java:149)
    at org.eclipse.ui.internal.UILockListener.doPendingWork(UILockListener.java:162)
    at org.eclipse.ui.internal.UISynchronizer$3.run(UISynchronizer.java:154)
    at org.eclipse.swt.widgets.RunnableLock.run(RunnableLock.java:35)
    at org.eclipse.swt.widgets.Synchronizer.runAsyncMessages(Synchronizer.java:182)
    ... 33 more
```

The reason might be that somewhere in the httpInvoker call the cdmEntity of the update result is also added to the updated objects and therefore it is not defined which of the both elements is returned by iterator().next();

It is also not correct IMO that swapSynonymAndAcceptedTaxon adds only the synonym to the updated objects. The taxon was also updated.

## #6 - 06/15/2021 01:14 PM - Andreas Müller

In general I wonder if it is a good idea to stay with the new implementation of the method. The current problem shows that users expect that data

stays with the name, not with the taxonbase record. So also supplemental data and so needs to be kept.

Shouldn't we simply use the old implementation again by using `newUuidForAcceptedTaxon=true`? The problem that occurred here some time ago is fixed I think.

However, both solutions have advantages and disadvantages, but the current case shows, that the current solution is not simple as one might think and that it is not necessarily what users expect.

#### #7 - 06/18/2021 09:04 AM - Katja Luther

Discussion how to handle Secundum Reference:

mail AM:

noch eine bisschen detaillierter Überlegung zu der Secundum Referenz:

Relevant ist das Problem nur, wenn diese zwischen altem akzeptiertem Taxon und altem Synonym abweichend ist. In diesem Fall ist die Bedeutung der sec des akz. Taxons die, dass in dieser Quelle das Konzept beschrieben wird. Die sec. des Synonyms (syn. sec.) besagt, dass eine Quelle diesen Namen als Synonym des o.g. akz. Taxonkonzepts sieht, dieses Synonym aber in der Quelle des akz. Taxons fehlt (sonst könnte diese Quelle selber verwendet werden und es wäre wieder acc. sec. = syn. sec.).

Wenn jetzt das akzeptierte Taxon und das Synonym getauscht werden, ändert sich dadurch das Konzept nicht (oder nur in dem Fall, in dem die Konzeptdefinition einen expliziten akzeptierten Namen miteinbezieht). Es macht also eigentlich Sinn, dass das neue akzeptierte Taxon die sec vom alten akzeptierten Taxon übernimmt. Lediglich sollte der `NameInSource` ggf. angepasst werden, da dieser sich ja ändert. Diesen kann man seit neustem explizit erfassen in der `secundum source`.

Beim Synonym ist das etwas anders: wenn diese besagt, dass ein Name als Synonym zu einem Taxonkonzept gehört, aber im Originalkonzept nicht genannt wurde, sollte i.d.R. das neue Synonym, welches vorher akzeptiertes Taxon war seine sec. (dann als syn. sec.) behalten, da es ja vermutlich (aber nicht zwangsläufig) in der Quelle des Konzepts genannt wurde.

Zusammenfassend: der Default-Fall wäre eigentlich, dass nach der Operation sowohl das neue akzeptierte Taxon als auch das neue Synonym die sec. vom alten akzeptierten Taxon übernehmen und zusätzlich der `NameUsedInSource` des neuen akzeptierten Taxons gesetzt wird.

Zusätzlich sollte für diesen eher seltenen Fall, dass acc. sec. und syn. sec. von einander abweichen, immer mindestens eine Warnung erscheinen, dass die Secundumreferenzen überprüft werden sollten.

Soviel zur Theorie. In der Praxis führte die Tatsache, dass derzeit Sec-Ref des alten akzeptierten Taxons dem neuen akzeptierten Taxon und die des alten akzeptierten Synonyms dem neuen akzeptierten Synonym zugefügt wird zu Irritation bei Nadja, die das genau andersherum erwartet hätte. Dies mag auch daran liegen, dass die Secundum-Referenz in Caryophyllales semantisch teilweise anders verwendet wird als oben beschrieben, wenn ich es richtig verstanden habe.

Da das Problem zumindest derzeit m.E. nur in Caryophyllales und in der Cuba-DB vorkommt, wäre es vor allem von den an diesen DBs Arbeitenden interessant zu wissen, was das erwartete Standardverhalten sein soll, und welche Maßnahmen es geben soll, um von diesem Standard abzuweichen.

Möglich wäre einfach nur eine Warnung, dass ggf. manuelles Nacharbeiten notwendig ist, oder eine Auswahl aus möglichen Handlungsalternativen.

Wie von Katja bereits beschrieben, bräuchten wir für die Supplemental-Data des Taxons/Synonyms (nicht der Namen!) eigentlich etwas Ähnliches. Hier geht es um Annotations, Marker, Sources, Identifier, Extensions, Credits und Rights. Diese werden bislang selten genutzt, aber wenn sie genutzt werden müsste eigentlich in jedem Fall entschieden werden, ob sie beim Taxon/Synonym bleiben, quasi mit dem Namen mitwandern oder ganz gelöscht werden sollen.

Da das nicht ganz trivial ist würde ich erstmal vorschlagen, dass wir es erstmal so handhaben, dass bei Vorliegen von taxonbasierten supplemental data lediglich eine Warnung erscheint, dass diese ggf. manuell verschoben oder gelöscht werden müssen.

Anm.: Bei genauerer Betrachtung müsste eigentlich auch bei acc. sec. = syn. sec. entschieden werden, ob der `NameInSource` (neu) gesetzt werden soll durch die Operation. Diesen gibt es erst seit kurzem, daher war das bislang auch noch nicht Thema. Dies wäre dann für alle DBs relevant, die mit externen Secundum-Referenzen arbeiten (also mit Referenzen, die sich nicht auf die eigene Datenbank beziehen).

mail WB:

vielen Dank für's detaillierte Durchdenken!

Wie Du schon sagst ist selbst bei Gleichheit der sec.- und syn.-sec. Referenz der Fall nicht einfach so abzuhandeln. Und ein Swap kann ja auch eine von mehreren Operationen an einem Taxon sein, sodass auch die Übernahme der alten sec.-Ref. mit `NameInSource` daneben liegen kann. Ich plädiere dafür, hier als Default beide zu löschen – dann kommt jedenfalls nichts falsches heraus. Später könnte man das dann eventuell verfeinern, aber derzeit wäre das die sichere Variante.

Hinsichtlich der Supplemental data wäre – wenn vorhanden – ein Hinweis gut, der klarstellt, was da passiert. Ich denke nicht, dass man diese Daten einfach übernehmen kann, kenne aber dazu die Verwendung zu wenig. Die Create/Update Daten werden ja vermutlich ohnehin automatisch geändert.

mail NadjaK:

In der Praxis bei Caryophyllales ist es so, dass wenn ein akzeptiertes Taxon und ein Synonym getauscht werden, auch eine neue Sec.-Referenz hinzukommt, aufgrund derer der Tausch gemacht wird, also zum Beispiel ein neues Treatment. Ein neuer Name, der dann der neue akzeptierte Name werden soll, kann ja überhaupt nur als Synonym eingegeben werden und dann erst getauscht werden. Das bedeutet dann in der Praxis, dass die vorherige sec. Referenz ohnehin aktualisiert werden muss.

Als Default beide zu löschen könnte wahrscheinlich hilfreich sein, dann muss aber auf jeden Fall ein Hinweis erscheinen. Ich fände es aber noch besser, wenn einfach alles beim Namen bleibt, wie es eingegeben war.

mail AM:

ok, aber bitte nur für den Fall, dass syn. sec. <> acc. sec. In den anderen Fällen sollte der Default sein, dass beide die gleiche sec. behalten.

mail AM:

Falls es keinen großen Mehraufwand bedeutet fände ich es schon gut, quasi alle 3 Möglichkeiten anzubieten, also (sec. beide löschen, sec. beim Namen belassen, sec. an der Position akzeptiert/Synonym belassen) mit default auf die erste Lösung. Aber wenn das aufwendig ist, dann besser nur die erste Lösung.

Interessant fand ich deinen Hinweis, Nadja, dass die Operation ja meist nur Teil einer anderen Operation ist, nämlich einen neuen akzeptierten Namen hinzuzufügen. Verstehe ich das richtig, dass es sich also eigentlich um Rekombinationen aufgrund neuer taxonomischer Erkenntnisse handelt?

Eigentlich müssten wir also darüber nachdenken, wie wir diese Operation so anbieten können, dass alle relevanten Schritte in einem stattfinden, also das Anlegen des neuen Namens (wenn notwendig), das Verschieben des Taxons an eine neue Position und das Anpassen der Sec-Referenzen (und weitere Operationen). Im Zuge der TaxonID-Diskussion im Kontext von E+M denken wir da ja schon drüber nach. Später sollten wir das für andere Kontexte wie dem hiesigen machen.

Bezüglich des Handlings der Supplemental Data stimme ich Walters Vorschlag zu.

mail NadjaK:

ja, genau, es sind Neukombinationen aufgrund neue taxonomische Erkenntnisse, oder einer phylogenetischen Studie, und das kommt ja immer wieder vor. In dem Fall, den ich jetzt eben hatte wurde zum Beispiel Brasiliopuntia schickendantzii neu in die Gattung Salmonoopuntia kombiniert – der Name kommt neu hinzu und das Taxon wird in eine andere Gattung verschoben.

Wenn alle Schritte als eine Operation gemacht werden könnten, wäre das natürlich super.

Und dann noch gibt es den Fall, wie jetzt bei der Kakteen-Checkliste: hier haben wir die Neukombinationen, die in dem Paper erscheinen werden, schon mal als Synonyme angelegt, um WFO-IDs vergeben zu können, und erst wenn das Paper erscheint, kann ich swappen und das Taxon verschieben.

mail WB:

ok, aber bitte nur für den Fall, dass syn. sec. <> acc. sec. In den anderen Fällen sollte der Default sein, dass beide die gleiche sec. behalten. Nein, bitte in allen Fällen!

Die syn.-sec. Referenz sagt aus, dass dieser Name als Synonym des Taxons betrachtet wird. Wenn dies nun zum Taxonnamen erklärt wird, muss eine neue Publikation vorliegen, die die neue Aussage macht.

Die sec.-Ref. sagt aus, dass das Konzept des Taxons hier zu finden ist. In der Praxis ist das aber schwammig mit der Benennung des Taxons verknüpft. Das kann man zwar in Zukunft durch den Name-in-source klarstellen, das wird aber bislang (noch) nicht gemacht. Also sollte es auch hier eine bewusste herausgeberische Entscheidung sein, ob man die sec.-Ref. stehen läßt (und dann ggf. den Name-in-source hinzufügt).

Diese Operation kann Teil verschiedener Operationen sein – ich würde also derzeit dafür plädieren, das nicht mit einem zu verbinden.

mail WB:

das lief parallel. Nadjas Aussage gilt für einen der Fälle (Änderung des Gattungskonzepts). Es gibt aber auch z.B. nomenklatorische Gründe, warum eines der homotypischen Synonyme den korrekten Namen darstellt.

Und dann gibt's noch den Fall, dass ein heterotypisches Synonym zum korrekten Namen wird – das ist im Swap bisher, glaube ich, nicht vorgesehen

...

mail AM:

dieses Vorgehen wäre für einen nicht größeren Teil unserer Portale kritisch, da hier die sec. Referenz gleich dem Portal ist bzw. eine Referenz, die auf einen Subtree zeigt und für alle Taxa und Synonyme des Subtrees gleich ist.

Nur dort, wo explizit mit externen sec.-Referenzen auf einzelnen Taxa gearbeitet wird (derzeit glaube ich nur Caryo\_spp und Cuba und vielleicht Salvador) ist das relevant, für alle anderen ist es eine unerwartete Unterbrechung im Workflow und da sie sich i.d.R. nicht mit sec.-Referenzen beschäftigen auch eher eine nicht so verständliche.

Genau die gleiche Diskussion hatten wir vor einem Jahr bereits bei den Operationen, die ein Synonym zu einem akzeptierten Taxon machen und anders herum (also split und merge).

Damals haben wir das einerseits mit dem genannten Dialog, andererseits mit Präferenzen gelöst, also der Möglichkeit, dass DB spezifisch zu definieren, was passieren soll (immer eine Lösung, immer die andere Lösung oder jedes mal Feedback, ...).

Wenn das für euch also wichtig ist, auch bei syn. sec. = acc. sec. zu warnen und die sec. beide per default zu löschen, dann werden wir auch hier um eine Präferenz nicht herumkommen. Außer alle anderen User stimmen dem Vorgehen zu, was ich mir aber nicht so ganz vorstellen kann, da meine Erfahrung eher ist, dass die User, für die die sec.-Referenz wenig bis gar nicht relevant ist sich auch nicht damit bei jeder Operation beschäftigen wollen.

Oder hast du da andere Erfahrung, Walter?

mail WB:

ich würde dann eigentlich eher die Einstellung „sec.-Ref. immer gleich im Subtree“ in die Präferenzen stellen dann automatisieren und damit die Behandlung der konzeptionell „echten“ sec.-Refs. zum default machen. Aber das ist letztendlich Implementierungssache.

Bei Euro+Med kenne ich diese Behandlung der sec.-Refs. – aber ist das tatsächlich auch bei Cichorieae so? Bei Datenbanken, die im systematischen Teil einer Publikation landen sollen, wird es wohl eher gebraucht.

Aber, wie gesagt, das ändert nichts daran, dass man hier unabsichtliche Fehleinträge – wo relevant – unbedingt vermeiden muss. Die Möglichkeit der Zuschreibung an allen Ecken und Enden ist ja eben ein Alleinstellungsmerkmal der Plattform.

**#8 - 07/07/2021 12:15 PM - Andreas Müller**

- Status changed from Feedback to In Progress

- Target version changed from Release 5.25 to Release 5.45

**Files**

---

picture091-1.png	170 KB	06/15/2021	Katja Luther
picture091-2.png	177 KB	06/15/2021	Katja Luther