

## EDIT - task #2346

### [Discuss] Version Numbering for CDM Library

04/26/2011 03:50 PM - Andreas Müller

<b>Status:</b>	Rejected	<b>Start date:</b>	
<b>Priority:</b>	Priority14	<b>Due date:</b>	
<b>Assignee:</b>	Andreas Müller	<b>% Done:</b>	0%
<b>Category:</b>	cdmlib	<b>Estimated time:</b>	0:00 hour
<b>Target version:</b>			
<b>Severity:</b>	normal		

#### Description

Halte ich für eine gute Grundlage sobald wir OpenSource Projekt sind. Sollten wir auf jeden Fall dann mit diskutieren.

Derzeit führt es allerdings vermutlich zu einer relative schnellen Erhöhung unserer minor Versionsnummern und auch der major Versionsnummern, da wir doch noch relativ häufig bestehende APIs brechen (was vielleicht ok ist, solange die Anzahl der betroffenen noch so gering ist).

AM

---Ursprüngliche Nachricht---

Von: Hoffmann, Niels

Gesendet: Donnerstag, 21. April 2011 11:07

An: Müller, Andreas; Kohlbecker, Andreas; Koch, Juergen; Luther, Katja

Betreff: AW: Schema update

Hi,

Auch auf die Gefahr hin hier jetzt als eclipse Evangelist zu gelten, möchte ich auf dieses Dokument hinweisen:

[http://wiki.eclipse.org/Version\\_Numbering](http://wiki.eclipse.org/Version_Numbering)

Da kann man sich anschauen wie das die anderen machen.

Ich persönlich würde das Service Segment hochzuzählen wollen. Man kann aber sicher beides argumentieren.

Wir sollten und auch noch einmal Gedanken übe die Schema-Version machen und diese gegebenenfalls mit der cdmlib Version konsolidieren.

Viele Grüße

Niels

---Ursprüngliche Nachricht---

Von: Müller, Andreas

Gesendet: Donnerstag, 21. April 2011 00:59

An: Kohlbecker, Andreas; Hoffmann, Niels; Koch, Juergen; Luther, Katja

Betreff: AW: Schema update

Müsste man diskutieren, da wir das bisher in CdmMetaData.schemaVersion anders handhaben. Dort haben wir ja bis zu 4 Nummern, wobei die beiden ersten Nummern mit der Library-version übereinstimmen. Bei sehr kleinen Schemaänderungen wie der jetzigen ist es ein bisschen fraglich, ob wir unbedingt gleich die cdmlib-Nummer aufwärtszählen müssen. Bisher war das nämlich immer mit Zusatzaufwand verbunden, weil Änderung aller poms, etc. Ist das jetzt irgendwie automatisiert?

AM

----Ursprüngliche Nachricht----

Von: Kohlbecker, Andreas

Gesendet: Mittwoch, 20. April 2011 22:14

An: Müller, Andreas; Hoffmann, Niels; Koch, Juergen; Luther, Katja

Betreff: AW: Schema update

Ich schlage vor bei einem Schema-Update die Minorversionnummer grundsätzlich um +1 zu erhöhen, damit wäre der trunk nun auf version 3.1.0-SNAPSHOT

was haltet Ihr vom diesem Vorschlag?

Andreas

+++++

Von: Müller, Andreas

Gesendet: Dienstag, 19. April 2011 14:05

An: Kohlbecker, Andreas; Hoffmann, Niels; Koch, Juergen; Luther, Katja

Betreff: Schema update

Hallo,

Am 15.4. mussten wir ein dringendes Schemaupdate im Bereich PolytomousKeyNode durchführen. Ich habe jetzt den SchemaUpdater dafür geschrieben, bestehende Datenbanken können damit geupdated werden.

Der Updater fügt der Tabelle PolytomousKeyNode\_AUD das Attribut parent\_id hinzu und löscht die Tabelle PolytomousKeyNode\_PolytomousKeyNode\_AUD.

Ausserdem werden einige schon länger ausstehende Terme hinzugefügt.

Bitte lasst wenn möglich den Updater laufen und ändert bestehende DBs nicht per Hand. Dies ist z.B. mittels des Editors möglich (ggf. Niels fragen).

Und denkt dran, dass zumindest für Produktionsdatenbanken vor dem Update auch der CDM Server geupdated werden muss.

Viele Grüße,

Andreas M.

## History

### #1 - 04/26/2011 04:01 PM - Andreas Müller

- Priority changed from Priority13 to Priority14

### #2 - 04/27/2011 11:31 AM - Andreas Kohlbecker

Halte ich für eine gute Grundlage sobald wir OpenSource? Projekt sind. Sollten wir auf jeden Fall dann mit diskutieren.

Derzeit führt es allerdings vermutlich zu einer relative schnellen Erhöhung unserer minor Versionsnummern und auch der major Versionsnummern, da wir doch noch relativ häufig bestehende APIs brechen (was vielleicht ok ist, solange die Anzahl der betroffenen noch so gering ist).

AM

Ich bin mir nicht sicher ob die Regeln in [http://wiki.eclipse.org/Version\\_Numbering](http://wiki.eclipse.org/Version_Numbering) für die Erhöhung des Service-Segments zum maven-release-plugin passen, das release plugin identifiziert Development-Branches anhand des Suffixes "-SNAPSHOT".

### #3 - 04/27/2011 11:42 AM - Niels Hoffmann

Replying to [a.kohlbecker](#):

Ich bin mir nicht sicher ob die Regeln in [http://wiki.eclipse.org/Version\\_Numbering](http://wiki.eclipse.org/Version_Numbering) für die Erhöhung des Service-Segments zum maven-release-plugin passen, das release plugin identifiziert Development-Branched anhand des Suffixes "-SNAPSHOT".

Das widerspricht sich nicht. In der Version Numbering Policy geht es erstmal nicht um Development Versionen sondern um Release Versionen. Was wir in irgendwelchen Dev Branches (bzw. trunk) machen muss und darf offizielle Release Versionen ja gar nicht beeinflussen IMHO. Wenn man solche Entwicklungen allerdings releasen will muss man sich dann überlegen, ob das ein Service, Minor oder Major upgrade ist.

**#4 - 07/02/2021 08:57 PM - Andreas Müller**

- *Description updated*
- *Status changed from New to Rejected*
- *Target version deleted (cdmlib - Old Next Major Release)*
- *Private changed from Yes to No*

I think I current version numbering works very well for us now even if we do not fully follow the suggestions in [http://wiki.eclipse.org/Version\\_Numbering](http://wiki.eclipse.org/Version_Numbering).

But as cdmlib probably will not become a generally used library in the near future this is not a real issue as long as cdmlib, taxeditor and vaadin work well together.

So I think we can close this ticket. Please reopen a new ticket if you think we need further discussion on this.