

Allgemeines:

Literaturlisten können sowohl in der Datenbank gespeichert werden als auch dynamisch/temporär erzeugt werden.

-> Wie werden temporäre Listen referenziert? Absprache mit DB nötig!

Vorschlag A: Ein bestimmter litListId-Bereich bleibt den temporären Listen vorbehalten. Die DB vergibt diese IDs nie, sondern nur die AL. Solche IDs sind session-spezifisch.

Vorschlag B: Es wird eine eindeutige ID von DB angefordert, die später wieder freigegeben wird.

Vorschlag C: Temporäre LitList's erhalten eine eindeutige ID und werden in der DB gespeichert. DB hält eine Liste der temporären LitList's. Bei Speicherung einer temporären LitList behält die LitList ihre ID und wird nur aus der temporären Liste entfernt.

Gedanken: Änderungen an einer geöffneten LitList werden erst beim Aufruf von "Speichern" in DB gespeichert, vorher sind sie nur im RAM. Temporäre LitList's könnten in der DB leer sein und nur im RAM liegen.

(Sollen LitList-Änderungen nach Session-Timeout als temporäre LitList's in DB gespeichert werden? Zusätzliches Feld: Original-ID.)

Rechte wurden von Multiusern geprüft und liegen bei Anfragen an AL auf jeden Fall vor!

Es ist davon auszugehen, dass die Zeitdifferenz zwischen MU-Rechteabfrage und AL-Anfrage minimal ist. Es zählen die Rechte zur Zeit der Anfrage bei MU, d.h. nach der MU-Rechteabfrage könnten die Rechte geändert worden sein und die alten Rechte sind noch für einen minimalen Zeitraum lang gültig.

Methode 5: Literaturliste umbenennen

Vorgehensweise:

MU an AL: `renameLitList(String litListId, String newName) //besser Key statt Id?`

AL an DB: Umbenennen

Probleme:

-newName bereits vergeben (Meldung von DB?)

-Liste wird gerade bearbeitet

Lösung:

-Exception falls Name existent, evtl. Alternativ-Vorschläge

- "Listener", der alle Bohnen bei Namensänderung updatet

Methode 6: Eintrag aus Literaturliste löschen

Vorgehensweise:

MU an AL: `deleteLitListEntry(String litListId, String entryKey)`

//oder `removeDataset(String litListKey, String dataSetKey)`

AL an DB: Datensatz aus Literaturliste entfernen

Anmerkungen:

Es wird nur die Referenz in der Literaturliste gelöscht, nicht der Datensatz selbst!

Probleme:

-Literaturliste wird gerade bearbeitet

Lösung:

- "Listener", der die Bohnen updatet

Methode 7: Eintrag in Literaturliste einfügen

Vorgehensweise:

MU an AL: `addToLitList (String litListId, String entryKey)`

//oder `addDatasetToLitList(String litListKey, String dataSetKey)`

AL an DB: prüfen ob Datensatz existent (?)

AL an DB: Referenz auf Datensatz in Literaturliste einfügen

Probleme:

-Datensatz existiert nicht/nicht mehr

-Literaturliste wird gerade bearbeitet

Lösung:

-Exception oder Alternativvorschläge existenter Datensätze

-"Listener"

Methode 8: Schließen einer Literaturliste

Vorgehensweise:

MU an AL: closeLitList(String litListId)

//oder closeLitList(String litListKey, boolean saveChanges)

AL: die "Bohne" wird aus dem Speicher entfernt

//AL an JDBC/DB: Literaturliste zur Bearbeitung freigeben

Probleme:

-Es wurden Änderungen an Literaturliste vorgenommen

-Datenbank überlastet

Lösung:

-Exception ODER Änderungen verwerfen

-Exception UND (Liste offen halten ODER Bohne trotzdem verwerfen)

Methode 9: Speichern einer Literaturliste

Vorgehensweise:

MU an AL: saveLitList(String litListId)

//oder saveLitList(String litListId, String newName)

AL: die "Bohne" wird in Datenbank gespeichert

AL an DB: neue Werte speichern

Probleme:

-Änderungen in der Datenbank seit letztem Öffnen der Literaturliste

-neuer Name schon existent

Lösung:

-Änderungen jeweils nur einer Person gleichzeitig erlauben

-Anhängsel an neuen Namen und trotzdem speichern UND Exception

Methode 10: Rechte für Literaturliste ändern

Vorgehensweise:

MU an AL: setLitListRights(String litListId, LitListRights llr);

//vorher: getLitListRights ?

AL an DB: Änderungen speichern

Probleme:

-Liste wird gerade bearbeitet

-Owner versucht sich selbst andere Rechte zu geben

Lösung:

-"Listener", der alle Bohnen bei Namensänderung updatet

-Owner in Zusatzbenutzern ignorieren

Methode 11: Öffnen einer Literaturliste

Vorgehensweise:

openLitList(String litListId);

neue "Bohne" erzeugen

Probleme:

-Literaturliste existiert nicht/nicht mehr

Lösung:

-Exception

Methode 12: Literaturliste löschen

Vorgehensweise:

deleteLitList(String litListId);

Probleme:

-Literaturliste ist gerade geöffnet oder wird bearbeitet

Lösung:

-Exception/Warnung ODER ("Listener" UND Bohne trotzdem löschen)