projekt kivitendo - Fehler #204

Automatisches Auslagern beim Rechnung schreiben funktioniert mit Commit 8c1d5d nicht mehr

20.08.2016 12:28 - Jan Büren

Status:	Gelöst	**Beginn:**	
Priorität:	Normal	**Abgabedatum:**	
Zugewiesen an:		**% erledigt:**	0%
Kategorie:		**Geschätzter Aufwand:**	0.00 Stunde
Zielversion:		**Aufgewendete Zeit:**	0.00 Stunde

Beschreibung

Von Ronny aus der Devel-Liste:

Wenn das Feature "automatisches Auslagern" bei der Mandantenkonfiguration gesetzt ist, funktioniert das Rechnungsschreiben nicht mehr - sobald man auf "Buchen"
geht "Drucken und Buchen" etc. geht.
Das ganze funktioniert seit dem Commit 8c1d5d nicht mehr.
Es wäre es schön, wenn diese Problem schnell behoben werden könnte, da
wir dieses Feature intensiv nutzen.

Historie

#1 - 20.08.2016 12:59 - Jan Büren

Hier nochmal, die ursprüngliche Anforderung:

http://trac.kivitendo.de/ticket/2491

#2 - 30.08.2016 15:41 - Martin Helmling

Habe ich gerade versucht zu reproduzieren:
neue Verkaufrechnung angelegt, eine Position angelegt mit Anzahl 10.
Dann gebucht, und unter Lagerbewegungen werden 10 Stk des Artikel herausgenommen/verschickt.

Was funktioniert den nicht !!

#3 - 05.09.2016 16:50 - Bernd Bleßmann

Jan hat das gefixt in 573d7fd17b43a09a21e1acb2423b0160523819a8 und f5b13074ae7356ed59a529ace1ccd76c5b4e49c9.

Entweder hat er vergessen, das Ticket hier zu schließen oder hat sich darauf verlassen, dass das "fix für #204" in seiner commit message das für ihn erledigt.

#4 - 05.09.2016 18:22 - Martin Helmling

Ja nun muss das Ganze wieder an single-dbh angepasst werden
oder ist $provided_dbh bereits das single-dbh ?

Werde es mal anschauen

#5 - 05.09.2016 22:12 - Bernd Bleßmann

Martin Helmling schrieb:

> Ja nun muss das Ganze wieder an single-dbh angepasst werden
> oder ist $provided_dbh bereits das single-dbh ?

Nun, da jetzt alles single-dbh sein sollte, sollte es auch $provided_dbh sein. Und wenn nicht, ist es auf jeden Fall der Handler der aufrufenden Routine (wenn die nicht irgendwas ungewöhnlich komisches übergibt. Hier ist es aber SL::DB::Inventory->new->db->dbh).
Der commit hat sich wahrscheinlich durch die Umstellung auf single-dbh erledigt, da get_standard_dbh ja jetzt den Rose-Handler liefert.

Genau da lag ja vorher das Problem. Beim Auslagern aus der Rechnung wird post_invoice nämlich mit dem Rose-Handler (in einer Transaktion) aufgerufen, der auch das Auslagern übernimmt, um das alles in einer Transaktion zu haben. Durch Deine Änderung wurde aber für dieselbe DB-Zeile, um mtime zu lesen, ein neuer/anderer Handler (standard_dbh) verwendet. Und der wartete auf das Ende der Transaktion (und wartete, und wartete).

#6 - 10.10.2016 15:02 - Jan Büren

- Status wurde von Neu zu Gelöst geändert

Hi, ja das ist in der Tat jetzt gefixt. Ich hatte das hier nur nochmal dokumentiert.
Ist jetzt geschlossen