Database - Feature #2541

replace the JDBC metadata calls at server startup with DMO annotations

03/30/2015 12:30 PM - Ovidiu Maxiniuc

Status:	New	**Start date:**	
Priority:	Normal	**Due date:**	
Assignee:		**% Done:**	0%
Category:		**Estimated time:**	0.00 hour
Target version:			
billable:	No	**version:**	
vendor_id:	GCD		

Description

History

#1 - 03/30/2015 01:02 PM - Ovidiu Maxiniuc

Currently, P2J will use the JDBC metadata calls at server startup to gather index information. The information on each index is used to trace back whether the index unique and for each component, the order of each component and in the case of character components, whether it is indexed as a case-sensitive or in ignore-case mode.

The Persistence.queryIndexData() and retrieveIndexData() will use a dedicated database connection and build an in-memory structure of the database metadata that will be later used for constructing further queries. A special case is preserving the case-sensitivity of char fields in dirty databases that will duplicate the structure of the persistent database with a different dialect (H2). If the database was altered, the memory-database will have a different structure than the original database, which can lead to failure execute queries.

This solution is both:

- **slow** because one table is queried at a time;
- **error-prone** because information between the two sources could differ (metadata from a database altered externally).

We can use of DMO annotations to gather the same information, in a faster, safer way. The iteration of DMO should be much faster because it's an in-process operation and the information retrieved are the same P2J expects to find as they are generated at conversion time. To further improve server startup time, a lazy initialization/creation of the 'dirty' tables could be implemented.

We already have the annotations in place, each DMO impl is annotated with enough data to recreate the schema for the dirty database (fields & indexes). The next step is to rework Persistence.queryIndexData() and retrieveIndexData() to use java annotation API for DMOs instead of JDBC metadata API.

#2 - 03/28/2016 11:54 AM - Eric Faulhaber

- Target version set to 24

Would require conversion and runtime changes.

#3 - 04/18/2016 05:17 AM - Paul E

I like this idea.

#4 - 11/16/2016 01:16 PM - Greg Shah

- Target version deleted (24)