

Mulgara - Bug #132

GraphAnswer improperly built

07/10/2008 12:54 AM - Paula Gearon

Status:	New	Start date:	
Priority:	High	Due date:	
Assignee:	Paula Gearon	% Done:	0%
Category:	Mulgara	Estimated time:	0.00 hour
Target version:	SPARQL Query Engine		
Resolution:			
Description			
<p>This class was built as a result type for CONSTRUCT queries, as per #112. As DESCRIBE queries also return graphs, the initial implementation is also used as the result type for these queries as well.</p> <p>Unfortunately, when I first built CONSTRUCT I was half thinking of how they are used in TQL INSERT/SELECT queries, where a multiple of 3 selection items are automatically inserted into the correct column. This translated into the GraphAnswer object accepting only 3 columns of results, and renaming these to "subject", "predicate" and "object". This is incorrect, as it prevents templates from having more than 3 columns (they must be a multiple of 3), prevents sorting by selected variables, and disallows different variable names. These are all incorrect.</p> <p>CONSTRUCT queries are converted to SELECT queries with a multiple of 3 columns. The appropriate Answer will still have only 3 columns, which may be renamed to "subject", "predicate", and "object", but it must wrap this wider Answer, and not a 3 column Answer as is currently presumed.</p> <p>The implementation will keep an internal counter for iterating over groups of 3 columns in the wrapped Answer with each call to next(). This continues until the end of a row. At the end of a row, the counter resets to zero, and the wrapped Answer has next() called on it. If this <i>wrapping</i> Answer is applied late enough, then sorting will be handled automatically.</p> <p>DESCRIBE queries need not change from their current implementation, but if this happens then sorting will not be applied. Given that DESCRIBE has no requirements on returned data, this may be acceptable. However, there is an alternative.</p> <p>Like CONSTRUCT queries, DESCRIBE can have their own Answer type that wraps a given Answer. In this case, we can presume DESCRIBE queries to return multiple columns, with the first 2 given the labels of "predicate" and "object". All other columns will be mapped to "subject". The implementation will need to scan each of these columns to find that <i>one</i> column that is bound for that row, and return that as the subject. Again, this will provide sorting, if the wrapper is applied late enough.</p> <p>Both types of Answer will appear to have only 3 Variables: "subject", "predicate" and "object". Many of the filtering characteristics will also be the same, in that Subjects cannot be a Literal, and Predicate cannot be a Literal nor a Blank Node. For this reason, GraphAnswer should be changed to an abstract class implementing this functionality, with the implementations of next() and getObject() performed in the extending classes.</p>			