

## Mulgara - Bug #15

### Product of Sum form has very poor performance.

09/06/2006 04:30 AM - Andrae Muys -

<b>Status:</b>	In Progress	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Andrae Muys -	<b>% Done:</b>	0%
<b>Category:</b>	Mulgara	<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>			
<b>Resolution:</b>			

#### Description

[[OrderedAppend]] currently assumes that all it's operands share a common ordering. This is currently guaranteed by [[TuplesOperations]].append(), but the use of calls to project() on all operands not matching the first's variable list. This has significant performance ramifications as the desired ordering is unknown until any parent-join is optimised.

<br/>

<br/>

Without the projections currently done by [[TuplesOperations]].append(), we need new logic in [[OrderedAppend]] to handle mapping variables to operands columns. This is not a problem unless there the disjunction is subject to a join containing a left-bound prefix including mismatched variables.

<br/>

<br/>

Consider the following tuple expressions:

<br/>

<br/>

$A[\$z] \wedge B(C[\$z \ \$y] \vee D[\$y \ \$z])$

<br/>

<br/>

Note:

<br/>

&nbsp;&nbsp;&nbsp;A[] will provide \$z prefix to B[].

<br/>

&nbsp;&nbsp;&nbsp;B[] currently passes this on blindly to C[] and D[].

<br/>

&nbsp;&nbsp;&nbsp;This will bind \$z for \$y in D[], leading to an incorrect result.

<br/>

&nbsp;&nbsp;&nbsp;Currently the non-union compatible

<br/>

&nbsp;&nbsp;&nbsp;

Fixing this will require either reordering D[], or filtering it; deciding between them is a performance optimisation issue.

<br/>

<br/>

$A[\$y \ \$z] \wedge B(C[\$y] \vee D[\$z])$

<br/>

<br/>

Note:

<br/>

&nbsp;&nbsp;&nbsp;Here there is a full prefix provided by A[] to B[].

<br/>

&nbsp;&nbsp;&nbsp;The prefix needs to be decomposed for C[] and D[]

<br/>

&nbsp;&nbsp;&nbsp;

This is a case of non-union compatible disjunction, and will probably result in UNBOUND's.

<br/>

```
<br/>
Cases where this is probably a problem:
<br/>

<br/>
non-symmetric sum-of-products (ie. { $a <foo> $b ^ $b <bar> $c in <m1> } v { $b
<bar> $c ^ $a <foo> $b in <m2> } )
<br/>
 
Note: this is one of the key areas of concern. Differences between models can cause join-optimisa
tion to generate a different ordering; This interferes with attempts to improve SOP performance wh
ich is required to support efficient views.
<br/>

<br/>
symmetric disjunctions: (ie. $s <foo> $o v $o <bar> $s). - uncommon.
<br/>

<br/>
non-union compatible disjunction (ie. $s <name> $name v $s <email> $email)
<br/>
 
Note: this query is better phrased as a subquery anyway.
```

## History

---

### #1 - 05/15/2007 04:47 AM - Andrae Muys -

With the NUC-disj fixes the workaround for this bug has been implemented - the performance considerations remain. With the workaround merged to trunk this bug is downgraded from major to minor.

### #2 - 05/19/2008 08:24 AM - Andrae Muys -

- Status changed from New to In Progress

Original Topic: [OrderedAppend](#) needs to check variable orderings for arguments before passing prefix. [TuplesOperations.append\(\)](#) needs to defer variable mapping to [OrderedAppend](#).