

## Database - Feature #1586

Feature # 1585 (New): add conversion and runtime support for word indexes

### add quick and dirty support for word indexes using LIKE operator

10/11/2012 11:33 PM - Eric Faulhaber

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Ovidiu Maxiniuc	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	16.00 hours
<b>Target version:</b>	Conversion Support for Server Features	<b>vendor_id:</b>	GCD
<b>billable:</b>	No		
<b>Description</b>			
<b>Related issues:</b>			
Precedes Database - Feature #1587: implement full support for word indexes			<b>Closed</b>

#### History

##### #1 - 10/22/2012 02:10 PM - Eric Faulhaber

- Start date deleted (10/15/2012)

##### #2 - 10/31/2012 04:10 PM - Greg Shah

- Target version set to Milestone 7

##### #3 - 01/31/2013 11:00 AM - Eric Faulhaber

- Assignee set to Ovidiu Maxiniuc

- Target version changed from Milestone 7 to Milestone 4

This issue isn't really about word indexes per se, but it is a quick and dirty precursor to true word index support, which we eventually will implement using something like Lucene or Hibernate Search.

This is about supporting the 4GL CONTAINS operator using an SQL LIKE operator. We already do something similar for the 4GL BEGINS operator. Please look at that code as a guide. Note that we don't expect this to be efficient for very large text records, which is why we have the follow up issue [#1587](#).

##### #4 - 02/05/2013 01:17 PM - Ovidiu Maxiniuc

- Status changed from New to WIP

I learned that this operator is a little strange, only works on whole words (btw, the field on which it is applied must be word-indexed) but on the other hand, is flexible enough to allow rather complicated search using parenthesis, "and" and "or" operators which connect more words prefixes. I am thinking of transforming CONTAINS string parameter into some regexp. I am not sure if sql LIKE is powerful enough, but I will do some dirty work there.

##### #5 - 02/05/2013 05:27 PM - Eric Faulhaber

For the current project, the use of CONTAINS is very consistent. The AST generally looks like:

```
contains [KW_CONTAINS]
  myField [FIELD_CHAR]
```

```
myVar [VAR_CHAR]
```

which corresponds to:

```
... myField CONTAINS myVar ...
```

What this means is that myVar will drop out as a query substitution parameter in all cases, and it will need to be processed/converted to an SQL LIKE phrase at runtime, not conversion time. We do something similar for the BEGINS and MATCHES operators, using `character.convertToSQLBegins(myVar)` and `character.convertToSQLLike(myVar)`, respectively. Let's add a runtime stub for `character.convertToSQLContains(character)` and wrap the right operand (e.g., myVar) of KW\_CONTAINS with a call to this new method.

This is a temporary solution until we have full word index support, so I don't want to put any significant effort into this solution that will not translate to the permanent solution. For now, since we don't have to parse any of the CONTAINS-specific syntax at conversion time, don't spend any time implementing `character.convertToSQLContains(character)`. We will need to implement this for Milestone 7, however.

#### **#6 - 02/12/2013 01:11 PM - Ovidiu Maxiniuc**

- File `om_upd20130212c.zip` added

- File `om_upd20130212d.zip` added

Implemented temporary solution.

Some more details were added to the javadoc of `character.convertToSQLContains()` method.

#### **#7 - 02/15/2013 10:07 AM - Eric Faulhaber**

- % Done changed from 0 to 50

- Status changed from WIP to Test

We are regression testing the update as is, which contains conversion support and some of the runtime implementation. Note that we also will need a server-side implementation of the contains function (i.e., PL/Java).

#### **#8 - 02/15/2013 12:31 PM - Eric Faulhaber**

- File `om_upd20130214b.zip` added

Sorry, I misspoke (mistyped?) in my last update. I made some minor changes to the 20130212c drop to merge with newer versions of some of the files and to fix a javadoc error. The update being tested is attached.

#### **#9 - 02/16/2013 09:05 AM - Eric Faulhaber**

- % Done changed from 50 to 100

- Status changed from Test to Closed

Update was regression tested and checked into bzd revision 10173.

**#10 - 11/16/2016 11:08 AM - Greg Shah**

- Target version changed from Milestone 4 to Conversion Support for Server Features

**Files**

---

om_upd20130212c.zip	105 KB	02/12/2013	Ovidiu Maxiniuc
om_upd20130212d.zip	1.76 KB	02/12/2013	Ovidiu Maxiniuc
om_upd20130214b.zip	106 KB	02/15/2013	Eric Faulhaber