

## Base Language - Feature #2138

### finish SEARCH-TARGET support

04/26/2013 02:26 PM - Greg Shah

<b>Status:</b>	Closed	<b>Start date:</b>	08/09/2013
<b>Priority:</b>	Normal	<b>Due date:</b>	08/16/2013
<b>Assignee:</b>	Constantin Asofiei	<b>% Done:</b>	100%
<b>Category:</b>		<b>Estimated time:</b>	40.00 hours
<b>Target version:</b>	Runtime Support for Server Features	<b>version:</b>	
<b>billable:</b>	No		
<b>vendor_id:</b>	GCD		
<b>Description</b>			
<b>Related issues:</b>			
Related to Base Language - Feature #1921: implement super-procedures		<b>Closed</b>	<b>11/09/2012</b>
Related to Base Language - Bug #1971: emit worker versions for functions to f...		<b>Closed</b>	<b>08/07/2013 08/08/2013</b>
Related to Base Language - Feature #1606: implement persistent procedures/sup...		<b>Closed</b>	<b>11/09/2012</b>

#### History

##### #1 - 04/26/2013 02:27 PM - Greg Shah

The work on this task is defined in [#1921](#) note 25:

A resume on what else needs to be done regarding super-procedure chains:

- make the SEARCH-TARGET and SEARCH-SELF emit as integer constants (was not a good idea to emit them as string literals, as this is not the way these are defined in 4GL) - 2 hours
- ProcedureManager.java - the TODO is related to SEARCH-TARGET mode, if the super-procedures list gets changed while in super calls for this mode, no more super calls are possible. - this is hard to estimate, as I couldn't figure out how 4GL keeps the procedure chain, so that it is invalidated when changed... I want to reserve at least 25 hours for this.
- once [#1971](#) is completed, do a thorough check of how the super procedures get invoked, in both SEARCH-TARGET and SEARCH-SELF modes - 10 hours

##### #2 - 05/01/2013 02:31 PM - Greg Shah

- Due date changed from 08/26/2013 to 09/10/2013
- Start date changed from 08/20/2013 to 09/04/2013

##### #3 - 05/02/2013 12:27 PM - Greg Shah

- Due date changed from 09/10/2013 to 08/16/2013
- Start date changed from 09/04/2013 to 08/09/2013

##### #4 - 09/18/2013 08:49 AM - Greg Shah

- Parent task deleted (#1606)

##### #5 - 11/18/2013 03:01 AM - Constantin Asofiei

- Status changed from New to WIP

**#6 - 12/09/2013 06:53 AM - Constantin Asofiei**

- File *ca\_upd20131209a.zip* added

This fixes the support for the SEARCH-SELF and SEARCH-TARGET modes; regression testing is running. The changes for the [#1968](#) (map-to clause) and [#1971](#) (java function calls) are merged in this update and will be released together.

**#7 - 12/09/2013 01:31 PM - Constantin Asofiei**

1209a.zip has passed regression testing (runtime and conversion).

**#8 - 12/09/2013 02:45 PM - Greg Shah**

Code Review 1209a

I'm OK with this update except for:

1. In ProcedureManager, there is this code:

```
if (mode == null || oldPp.searchMode == null || oldPp.searchMode.equals(mode))
{
    newPp = oldPp == null ? new ProcedureData(h, mode) : oldPp;
```

The previous code checked for oldPp == null in the if(). This code looks like it might generate an NPE in that same case unless oldPp is only ever null when mode is also null. It seems a bit dangerous and is potentially confusing to the reader.

2. In the above referenced code I would prefer parens inserted like this: newPp = (oldPp == null) ? new ProcedureData(h, mode) : oldPp;

**#9 - 12/10/2013 08:34 AM - Constantin Asofiei**

- File *ca\_upd20131210a.zip* added

I've rewritten the code to make it more clear... the old logic was not that OK, although it did work. Local tests pass, I'll re-run regression again, just to make sure.

**#10 - 12/10/2013 08:47 AM - Greg Shah**

Code Review 1210a

I am fine with the changes. You can check in and distribute when it passes testing.

**#11 - 12/12/2013 03:23 PM - Constantin Asofiei**

- % Done changed from 0 to 100

- File *ca\_upd20131212a.zip* added

Attached update was merged with 10419 ([#2196](#) work), passed regression testing and was committed to bzt revision 10420.

**#12 - 12/12/2013 04:06 PM - Greg Shah**

- Status changed from WIP to Closed

**#13 - 05/06/2014 04:27 AM - Constantin Asofiei**

- File `ca_upd20140505a.zip` added

After the case found in #2266, the SUPER and RUN SUPER calls were re-investigated and these rules were found, for SEARCH-TARGET mode:

1. for both functions and procedures, if THIS-PROCEDURE is in the super-procedure list for the SESSION or TARGET-PROCEDURE, remove elements (in a LIFO order), using the TARGET-PROCEDURE's list first and SESSION's second. When THIS-PROCEDURE is found, it is removed too, and the removal stops. The next invoked persistent procedure will be the next valid element from the TARGET-PROCEDURE's or SESSION's list (if the TARGET-PROCEDURE's list is empty), in a LIFO order.
2. for procedures, there are some quirks which allow infinite recursion. In these cases, the call-stack is searched and it stops either when:
  - the called persistent procedure is part of the TARGET-PROCEDURE's super-procedure list. This referent is used as THIS-PROCEDURE is used in rule 1 above.
  - a non-super call is reached and the persistent procedure is not the target procedure. In this case, no referent is found, and it is decided no more super-procedures are available.

The testcases are in `testcases/uast/procedures`. To run them, use this command:

```
pro -p run_all.p -nb 20000 -s 10000000 -mmax 65534
```

which ensures that the error messages are not polluted with messages like "not enough stack space, increased automatically", when recursion takes place.

In this tests, infinite recursion is assumed if the same procedure/function is invoked more than 20 times. Also, although the search mode is defined via the `tsm` variable for each test, note that is not necessarily the search mode used for a certain procedure.

Conversion for these tests must be done using the `uast/procedures` folder as the root project folder.

The `ca_upd20150505a.zip` (from #2266) is attached here too.

**#14 - 05/06/2014 08:42 AM - Greg Shah**

Code Review 0505a

1. What is the reasoning behind the elimination of the `pushSuperProcedure()` and `popSuperProcedure()`?
2. Please change the javadoc for `prepareSearches()` as follows:

Remove elements from the end of each queue until the specified referent is found.

**#15 - 05/06/2014 09:10 AM - Constantin Asofiei**

- File *ca\_upd20140506a.zip* added

Greg Shah wrote:

- 1. What is the reasoning behind the elimination of the `pushSuperProcedure()` and `popSuperProcedure()`?

Those were part of the previous (incorrect) implementation of determining the target of a RUN SUPER/SUPER call (for SEARCH-TARGET mode).

- 2. Please change the javadoc for `prepareSearches()` as follows:

Done, see attached.

Just to be safe, runtime testing passed.

**#16 - 05/06/2014 09:13 AM - Greg Shah**

Great! Please check it in and distribute it.

**#17 - 05/06/2014 09:38 AM - Constantin Asofiei**

Greg Shah wrote:

Great! Please check it in and distribute it.

Committed to bzip rev 10518.

**#18 - 11/16/2016 11:42 AM - Greg Shah**

- Target version changed from Milestone 7 to Runtime Support for Server Features

**Files**

ca_upd20131209a.zip	430 KB	12/09/2013	Constantin Asofiei
ca_upd20131210a.zip	430 KB	12/10/2013	Constantin Asofiei
ca_upd20131212a.zip	432 KB	12/12/2013	Constantin Asofiei

ca_upd20140505a.zip	18.2 KB	05/06/2014	Constantin Asofiei
ca_upd20140506a.zip	18.2 KB	05/06/2014	Constantin Asofiei