

## Database - Bug #4564

### unqualified field references can get the wrong buffer name if a FORM or DEF FRAME is present

03/03/2020 10:08 AM - Roger Borrello

<b>Status:</b>	New	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>		<b>% Done:</b>	0%
<b>Category:</b>		<b>Estimated time:</b>	0.00 hour
<b>Target version:</b>		<b>case_num:</b>	
<b>billable:</b>	No	<b>version:</b>	
<b>vendor_id:</b>	GCD		
<b>Description</b>			

#### History

#1 - 03/03/2020 10:16 AM - Roger Borrello

#### Description

Extra scoping causes issues with wrong buffer being used for unqualified field names. Same results within FRAME or FORM.

#### Testcase

uast/shared-buffer/shared-buffer-form2.p

```
def new shared buffer address-buf for address.  
form address.street with frame x.  
find address-buf where street = "garbage" no-error.  
addr-id = 1.  
message avail address-buf.
```

uast/shared-buffer/shared-buffer-frame2.p

```
def new shared buffer address-buf for address.  
def frame x address.street.  
find address-buf where street = "garbage" no-error.  
addr-id = 1.  
message avail address-buf.
```

#3 - 03/03/2020 10:44 AM - Greg Shah

- Subject changed from Need tighter scope binding for shared buffer within FIND to unqualified field references can get the wrong buffer name if a FORM or DEF FRAME is present

This has nothing to do with the FIND. The issue is in the assignment.

./shared-buffer/shared-buffer-form2.p

Elapsed job time: 00:00:02.784

ERROR:

com.goldencode.p2j.pattern.TreeWalkException: ERROR! Active Rule:

RULE REPORT

Rule Type : WALK
Source AST: [ addr-id ] BLOCK/ASSIGNMENT/ASSIGN/FIELD\_INT/ @5:1 {249108103224}
Copy AST : [ addr-id ] BLOCK/ASSIGNMENT/ASSIGN/FIELD\_INT/ @5:1 {249108103224}
Condition : throwException(sprintf(spec, bufname, reftype), this)
Loop : false
--- END RULE REPORT ---

at com.goldencode.p2j.pattern.PatternEngine.run(PatternEngine.java:1070)
at com.goldencode.p2j.convert.TransformDriver.processTrees(TransformDriver.java:542)
at com.goldencode.p2j.convert.ConversionDriver.back(ConversionDriver.java:562)
at com.goldencode.p2j.convert.TransformDriver.executeJob(TransformDriver.java:876)
at com.goldencode.p2j.convert.ConversionDriver.main(ConversionDriver.java:983)

Caused by: com.goldencode.p2j.pattern.CommonAstSupport\$UserGeneratedException: Cannot find buffer named p2j\_test.address\_p2j\_test.address for ref type 21! [FIELD\_INT id <249108103224> 5:1]

at com.goldencode.p2j.pattern.CommonAstSupport\$Library.throwException(CommonAstSupport.java:2817)
at com.goldencode.expr.CE8622.execute(Unknown Source)
at com.goldencode.expr.Expression.execute(Expression.java:373)
at com.goldencode.p2j.pattern.Rule.apply(Rule.java:497)
at com.goldencode.p2j.pattern.Rule.executeActions(Rule.java:745)
at com.goldencode.p2j.pattern.Rule.coreProcessing(Rule.java:712)
at com.goldencode.p2j.pattern.Rule.apply(Rule.java:534)
at com.goldencode.p2j.pattern.Rule.executeActions(Rule.java:745)
at com.goldencode.p2j.pattern.Rule.coreProcessing(Rule.java:712)
at com.goldencode.p2j.pattern.Rule.apply(Rule.java:534)
at com.goldencode.p2j.pattern.Rule.executeActions(Rule.java:745)
at com.goldencode.p2j.pattern.Rule.coreProcessing(Rule.java:712)
at com.goldencode.p2j.pattern.Rule.apply(Rule.java:534)
at com.goldencode.p2j.pattern.Rule.executeActions(Rule.java:745)
at com.goldencode.p2j.pattern.Rule.coreProcessing(Rule.java:712)
at com.goldencode.p2j.pattern.Rule.apply(Rule.java:534)
at com.goldencode.p2j.pattern.RuleContainer.apply(RuleContainer.java:585)
at com.goldencode.p2j.pattern.RuleSet.apply(RuleSet.java:98)
at com.goldencode.p2j.pattern.AstWalker.walk(AstWalker.java:262)
at com.goldencode.p2j.pattern.AstWalker.walk(AstWalker.java:210)
at com.goldencode.p2j.pattern.PatternEngine.apply(PatternEngine.java:1633)
at com.goldencode.p2j.pattern.PatternEngine.processAst(PatternEngine.java:1531)
at com.goldencode.p2j.pattern.PatternEngine.processAst(PatternEngine.java:1479)
at com.goldencode.p2j.pattern.PatternEngine.run(PatternEngine.java:1034)
... 4 more

4GL compile listing which shows that only address-buf is a valid buffer name:

...d-buffer-frame2.p 03/02/2020 07:08:41 PROGRESS (R) Page 1

{ } Line Blk
---
1 def new shared buffer address-buf for address.
2 def frame x address.street.
3 //form address.street with frame x.
4 find address-buf where street = "garbage" no-error.
5 addr-id = 1.
6 message avail address-buf.

...d-buffer-frame2.p 03/02/2020 07:08:41 PROGRESS (R) Page 2

Table with 4 columns: File Name, Line Blk, Type, Tran, Blk. Label

```
...d-buffer-frame2.p    0 Procedure    Yes
  Buffers: p2j_test.address-buf
  Frames:  x
```

4207a revision 11422 resolves this issue in record phrases, but there remains a general problem as seen here in an assignment.

**#4 - 09/09/2021 02:30 PM - Roger Borrello**

We are still patching some procedures to fully-qualify their field-references. They are willing to incorporate the updates into their code-base, which would alleviate us from the extra step, which gets challenging then they are actively updating code.

There have been many changes in the persist area... might this task have been addressed?

**#5 - 09/09/2021 02:56 PM - Greg Shah**

No. `com.goldencode.p2j.persist.*` is a runtime package. It has nothing to do with conversion.