

## Base Language - Bug #4682

### modify FILE-INFO processing to honor searches for 4GL programs

06/16/2020 04:34 PM - Greg Shah

<b>Status:</b>	Closed	<b>Start date:</b>	
<b>Priority:</b>	Normal	<b>Due date:</b>	
<b>Assignee:</b>	Constantin Asofiei	<b>% Done:</b>	100%
<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>			
<b>Related issues:</b>			
Related to Runtime Infrastructure - Feature #3565: provide better support for...		<b>WIP</b>	
Related to Base Language - Feature #4381: implement LIBRARY() and MEMBER() bu...		<b>Closed</b>	

### History

#### #1 - 06/16/2020 04:34 PM - Greg Shah

- Related to Feature #3565: provide better support for file system access across platforms added

#### #3 - 06/16/2020 04:43 PM - Greg Shah

There is a common pattern in 4GL code that is used for "security". The idea is that the code will use some form of file system inspection to see if r-code exists for a given program. If that r-code does not exist then the execution of the program is bypassed. See [#3565-6](#) and #4208-711.

Its most common form would try to set FILE-INFO:FILENAME to the value of a .r file OR it may use SEARCH() to look for a .r file. When a file cannot be found AND if the name being searched for ends in .p, .w or .r, to use SourceNameMapper.lookupLegacyName() to detect if there is a valid converted program that is associated with this name. If so, we would fake out the results to appear as if there was a real file system resource by that name in existence. For the cases seen so far, this would allow the subsequent RUN statement to execute which is all we want.

This solution is not perfect. It won't work for program names that don't follow common conventions. It also won't work if a "deeper" inspection of the file is done (e.g. more details like the size of the file read from FILE-INFO). That is OK. For now we just want to handle the common case.

#### #4 - 06/17/2020 04:04 PM - Roger Borrello

The code in adm2/containr.p does not specifically perform a LIBRARY call, prior to the code snippet that we are attempting to get functioning.

That snippet is below:

```
def var lib-info      as character no-undo init "dates.r".
def var cDotRFile    as character no-undo.

ASSIGN
  FILE-INFO:FILE-NAME = lib-info
  cDotRFile           = FILE-INFO:FULL-PATHNAME
  /* If .r in proc lib, get the member name (ie, filename). */
  cMember = MEMBER(cDotRFile).

message "cDotRFile = " cDotRFile.
message "cMember = " cMember.
```

I have run this in the 4gl, with dates.r in the PROPATH. The output is:

```
cDotRFile= z:\testcases\uast\dates.r  
cMember= ?
```

I am not sure which procedure library the MEMBER would be referencing. [#4381](#) was directed at handling LIBRARY and MEMBER, but may not be implemented correctly, if there is supposed to be a default library searched.

**#5 - 06/17/2020 04:05 PM - Roger Borrello**

- Related to Feature #4381: implement LIBRARY() and MEMBER() built-in functions added

**#6 - 06/17/2020 04:12 PM - Greg Shah**

For a .r file that is part of the application, there is no procedure library involved. For a procedure that is a member of the ADM stuff, maybe that is found in one of the 4GL procedure libraries. Why do we care? Does it have anything to do with this "fall back" behavior?

**#7 - 06/17/2020 05:34 PM - Roger Borrello**

Greg Shah wrote:

For a .r file that is part of the application, there is no procedure library involved. For a procedure that is a member of the ADM stuff, maybe that is found in one of the 4GL procedure libraries. Why do we care? Does it have anything to do with this "fall back" behavior?

Even after implementing the "fall back" behavior, when we are passed a .r and we find a .p, and return it in the full-pathname attribute, the use of MEMBER subsequently will not yield the proper results, as the string that MEMBER parses is the library<<member>> format. It will return unknown in that case. This was confirmed in the 4GL. If SEARCH is performed, it will return the correct formatted library<<member>> if the .r is found in a procedure library in the PROPATH, which isn't being done in containr.p.

Yes, this is only related to the one use-case. I just want to look ahead, to validate it is the correct solution to pursue at this time.

**#8 - 06/17/2020 05:50 PM - Roger Borrello**

More info related to the 4GL's behavior. It indicates we can proceed.

I was able to position a procedure library in the PROPATH, and the full-pathname attribute is returned in the library<<member>> format if I search for a member I know is in the library.

```
cDotRFile= C:\Progress\OpenEdge\gui\adecomm.pl<<adecomm\get-user.r>>
cMember= adecomm\get-user.r
```

#### #9 - 06/17/2020 06:00 PM - Greg Shah

Yes, but in FWD we have no such thing. I was not suggesting that we implement a "fake" procedure library result.

Do we need to modify that customer's code (or Possenet)?

#### #10 - 06/17/2020 06:40 PM - Roger Borrello

Greg Shah wrote:

Yes, but in FWD we have no such thing. I was not suggesting that we implement a "fake" procedure library result.

Do we need to modify that customer's code (or Possenet)?

I agree. It wouldn't be useful. The `containr.p:constructObject` in Possenet allows flexibility to invoke the procedure the customer passes, regardless of how it is presented to ADM2.

- There may already be a running instance on the server (or WebSpeed client, and the logical name is passed in via the `LogicalObjectName` property. This is not the case in our instance.
- It may be a DB-AWARE object (as it is in our instance)
  - `RunDOOptions` are retrieved. This drives the method of looking for the passed in procedure.
    - `DynamicOnly` (not set in our case) would indicate to pull the `DynamicSDOProcedure` property, which was `adm2/dyndata.w` in our instance, and dynamically call it.
    - `StaticClientOnly` (not set in our case) would enable a check for `SESSION:CLIENT-TYPE = 'WEBCLIENT'` and if so, invoke the passed-in procedure.
    - `ClientOnly` (not set in our case) drives a check for a proxy procedure with `"_cl.r"` appended to the base file name to be invoked
    - `SourceSearch` (proposed to be set in customer code) directs the procedure passed in to be invoked directly.

To modify the Possenet, we'd be creating an offshoot that we would need to document. I believe it's best to utilize their *somewhat* documented approach, and drive the customer code to meet that end.

#### #11 - 06/18/2020 06:39 AM - Greg Shah

To modify the Possenet, we'd be creating an offshoot that we would need to document. I believe it's best to utilize their *somewhat* documented approach, and drive the customer code to meet that end.

OK, I'm convinced.

**#12 - 08/24/2020 11:59 AM - Roger Borrello**

The updates I made in #4845-15 were specific to the use of `possenet/src/adm2/containr.p` to enable the search for a source file by default, which moved us on for that customer case. What is the current priority of this more "generous" search in FILE-INFO?

**#13 - 08/24/2020 12:03 PM - Greg Shah**

We have no plans right now. Is it the cause of more problems in real customer apps?

**#14 - 08/24/2020 12:16 PM - Roger Borrello**

- *Priority changed from Normal to Low*

Greg Shah wrote:

We have no plans right now. Is it the cause of more problems in real customer apps?

None that have come up. I'll change the priority accordingly.

**#15 - 08/24/2020 12:17 PM - Greg Shah**

- *Assignee deleted (Roger Borrello)*

- *Priority changed from Low to Normal*

No reason to make it low priority. It is just a feature that is not needed right now.

**#16 - 05/27/2021 08:07 AM - Constantin Asofiei**

`FileSystemOps.initFileInfo` was changed to find a converted external program or class, too - see 3821c rev 12462

A limitation is that it will find both the `.r` and `.p` versions - FWD doesn't know if there was a `r`code-compiled version of this file or not.

**#17 - 05/27/2021 02:52 PM - Greg Shah**

- *% Done changed from 0 to 100*

The changes are good.

A limitation is that it will find both the `.r` and `.p` versions - FWD doesn't know if there was a `r`code-compiled version of this file or not.

Yes, this is acceptable.

Is there any reason to leave this task open?

**#18 - 05/27/2021 03:06 PM - Constantin Asofiei**

Greg Shah wrote:

Is there any reason to leave this task open?

No.

**#19 - 05/27/2021 04:18 PM - Greg Shah**

- *Status changed from New to Closed*

- *Assignee set to Constantin Asofiei*