

## Database - Bug #5956

### FWD does not reset AVAILABLE flag after UDF error.

01/08/2022 04:12 AM - Igor Skornyakov

<b>Status:</b>	New	<b>Start date:</b>	01/08/2022
<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>vendor_id:</b>	GCD		
<b>Description</b>			

### History

#### #2 - 01/08/2022 04:27 AM - Igor Skornyakov

- File available.p added

Consider the following code (attached):

```
define temp-table udfctest
  field id as integer
  field fdate as date
  field fmt as character
  field fstr-result as character
  index pk as primary id
.

OUTPUT TO available.txt.

create udfctest. id = 1. fdate = date(07, 19, 2021). fmt = '999999'. fstr-result = "071921".
create udfctest. id = 2. fdate = date(07, 05, 2021). fmt = '999999'. fstr-result = ?.

message "After create:" AVAILABLE udfctest.

run test(?, ?) NO-ERROR.

run test(date(07, 05, 2021), '999999') NO-ERROR.
run test(date(07, 19, 2021), '999999') NO-ERROR.
run test(date(07, 05, 2021), '999999') NO-ERROR.

message 'done'.

procedure test:
  def input param val as date.
  def input param fmt as char.
  def var err-flag as logical init true.
  def var nmsg as integer no-undo.
  message "test(" val ", " fmt ")".
  inner:
  do on stop undo, leave inner:
    FIND FIRST udfctest
    WHERE udfctest.fdate = val
      AND udfctest.fmt = fmt
      AND string(fdate, fmt) = udfctest.fstr-result
    NO-ERROR
  .
  err-flag = false.
end.
message "After find - ERROR:" ERROR-STATUS:ERROR
"NUM-MESSAGES:" ERROR-STATUS:NUM-MESSAGES
"AVAILABLE:" AVAILABLE udfctest
"err-flag:" err-flag.
```

```

IF ERROR-STATUS:NUM-MESSAGES > 0 THEN DO:
  do nmsg = 1 to ERROR-STATUS:NUM-MESSAGES:
    message "****" ERROR-STATUS:GET-NUMBER(nmsg) ':' ERROR-STATUS:GET-MESSAGE(nmsg) .
  end.
END.
end.
OUTPUT CLOSE.

```

The STRING(date(07, 05, 2021), '99999') causes \*\* Date format 99999 is incomplete. (154) error.

The output with 4GL is:

```

After create: yes
test( ?, ? )
After find - ERROR: yes NUM-MESSAGES: 1 AVAILABLE: no err-flag: no
*** 565 : ** FIND FIRST/LAST failed for table udfctest. (565)
test( 07/05/21 , 99999 )
** Date format 99999 is incomplete. (154)
After find - ERROR: no NUM-MESSAGES: 1 AVAILABLE: no err-flag: yes
*** 154 : ** Date format 99999 is incomplete. (154)
test( 07/19/21 , 999999 )
After find - ERROR: no NUM-MESSAGES: 0 AVAILABLE: yes err-flag: no
test( 07/05/21 , 99999 )
** Date format 99999 is incomplete. (154)
After find - ERROR: no NUM-MESSAGES: 1 AVAILABLE: no err-flag: yes
*** 154 : ** Date format 99999 is incomplete. (154)
done

```

With FWD it is:

```

After create: yes
test( ?, ? )
After find - ERROR: yes NUM-MESSAGES: 1 AVAILABLE: no err-flag: no
*** 565 : ** FIND FIRST/LAST failed for table udfctest. (565)
test( 07/05/21 , 99999 )
** Date format 99999 is incomplete. (154)
After find - ERROR: no NUM-MESSAGES: 2 AVAILABLE: no err-flag: yes
*** 154 : ** Date format 99999 is incomplete. (154)
*** 154 : ** Date format 99999 is incomplete. (154)
test( 07/19/21 , 999999 )
After find - ERROR: no NUM-MESSAGES: 0 AVAILABLE: yes err-flag: no
test( 07/05/21 , 99999 )
** Date format 99999 is incomplete. (154)
After find - ERROR: no NUM-MESSAGES: 2 AVAILABLE: yes err-flag: yes
*** 154 : ** Date format 99999 is incomplete. (154)
*** 154 : ** Date format 99999 is incomplete. (154)
done

```

We see the following.

1. After the temp-table population both with 4GL and FWD AVAILABLE udfctest == yes.
2. After run test(?, ?) both with 4GL and FWD AVAILABLE udfctest == no
3. After first test(date(07, 05, 2021), '99999') both with 4GL and FWD AVAILABLE udfctest == no.
4. After test(date(07, 19, 2021), '999999') both with 4GL and FWD AVAILABLE udfctest == yes.
5. After second test(date(07, 05, 2021), '99999') both with 4GL AVAILABLE udfctest == no **but with FWD it is yes.**

The duplicate \*\* Date format 99999 is incomplete. (154) message with FWD is a separate issue (see [#5930](#)).

**Files**

---

available.p

1.42 KB

01/08/2022

Igor Skornyakov