

Database - Bug #8212

FILL operations do not work correctly with active dataset relations

01/23/2024 01:59 PM - Ovidiu Maxiniuc

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

History

#2 - 01/23/2024 02:08 PM - Ovidiu Maxiniuc

While working on #8178 I noticed some differences in the way FWD populates the database when there are active dataset relations between its buffers.

The testcase is the following:

```
output to \tmp\8212a.out.

define temp-table ttParent
  field pKey as integer
  field fData as character
  index pKey as primary unique pKey.

define temp-table ttChild
  field fKey as integer
  field fData as character.

define dataset sds for ttParent, ttChild
  data-relation fam for ttParent, ttChild relation-fields(ttParent.pKey, ttChild.fKey).

define temp-table srcParent like ttParent.
create srcParent. srcParent.pkey = 1. srcParent.fData = "one".
create srcParent. srcParent.pkey = 2. srcParent.fData = "two".
create srcParent. srcParent.pkey = 3. srcParent.fData = "three".
define data-source dsrcParent for srcParent.
buffer ttParent:attach-data-source(data-source dsrcParent:handle).
data-source dsrcParent:fill-where-string = "WHERE pkey LE 3".
message "P:" data-source dsrcParent:query:prepare-string.

define temp-table srcChild like ttChild.
create srcChild. srcChild.fkey = 1. srcChild.fdata = "uno".
create srcChild. srcChild.fkey = 1. srcChild.fdata = "un".
create srcChild. srcChild.fkey = 3. srcChild.fdata = "tre".
create srcChild. srcChild.fkey = 4. srcChild.fdata = "quattro".
define data-source dsrcChild for srcChild.
buffer ttChild:attach-data-source(data-source dsrcChild:handle).
// data-source dsrcChild:fill-where-string = "WhERE ttChild.fkey=ttparent.pkey".
// data-source dsrcChild:fill-where-string = "WHErE srcChild.fKey=ttParent.pKey".
message "C:" data-source dsrcChild:query:prepare-string.

dataset sds:fill().

for each ttParent:
  message "p:" ttParent.pkey ttParent.fdata.
end.
for each ttChild:
  message "f:" ttChild.fkey ttChild.fdata.
end.
```

Notice the two fill-where-string commented assignments. Enable them individually has different results in output as well as enabling/disabling the data-relation fam.