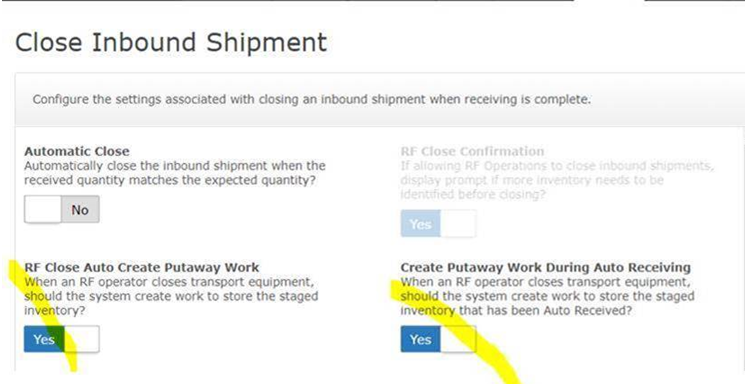
# Configure workflow for create PUTDIR work for ASN pallet when close trailer

## Summary:

This workflow is aimed to address the issue:

When ASN pallets are auto received, currently system only move these pallets to receive stage lane without creating putaway work, which is different with non-auto received pallet for which we do directed putaway and when deposit to receive stage, system is able to create PUTDIR work based on next move configuration, also system is able to deallocate allocated storage location since the movement zone on stage lane has clr\_dstloc\_flg enabled. We also tried to use policies:



With above 2 policies turned on, we are able to get putawy work created, but not able to clear the destination location allocated which is still a gap, this will cause user not able to putaway a pallet to a better location since the better location may occupied by another pallet which is still in stage lane.

To fill this gap, we determined to use a trailer close workflow to create putaway work without involving above 2 policies with assumptions:

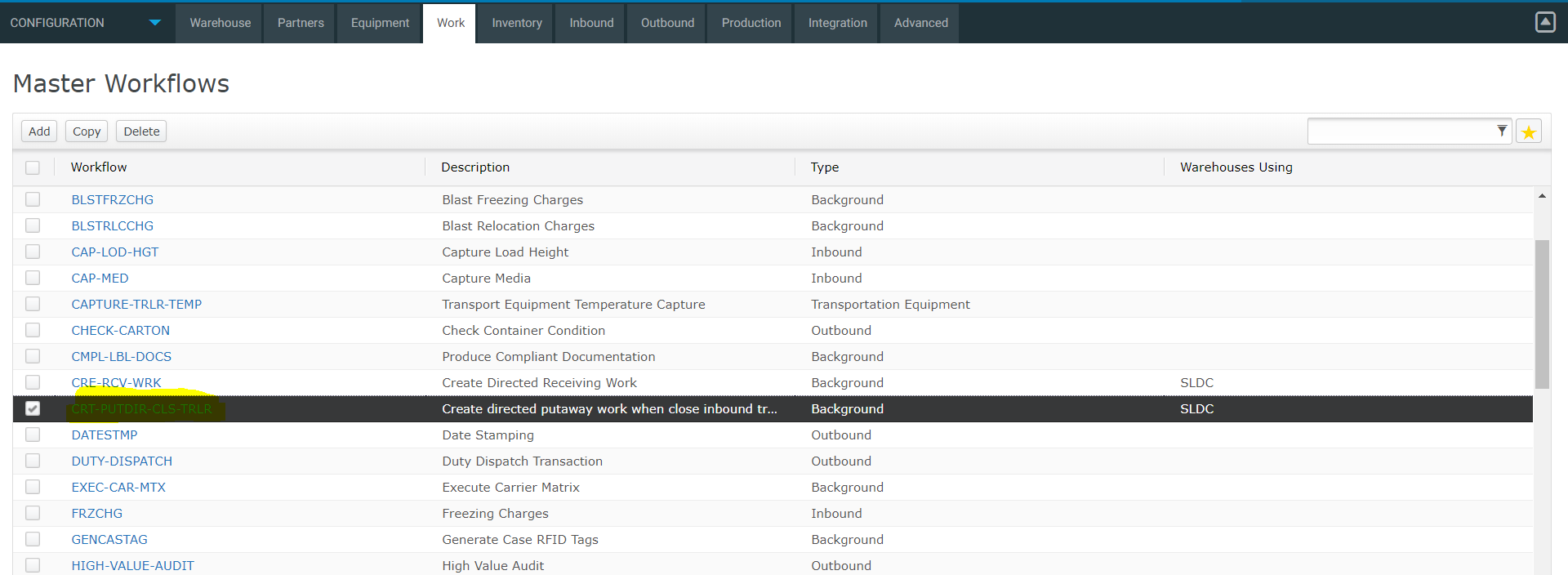
1. **The workflow only works for ASN pallets since non-ASN pallet worked well.**
2. **The workflow only creates PUTDIR work for pallet which sit in receive stage lane.**
3. **To reduce the performance impact when close trailer, the workflow will create a deferred execution other than create work on the spot, in maximum, it needs to wait the job DEFERRED-EXECUTION to run to create the work, currently this job is configured to run for every 1 minute.**

## Approach:

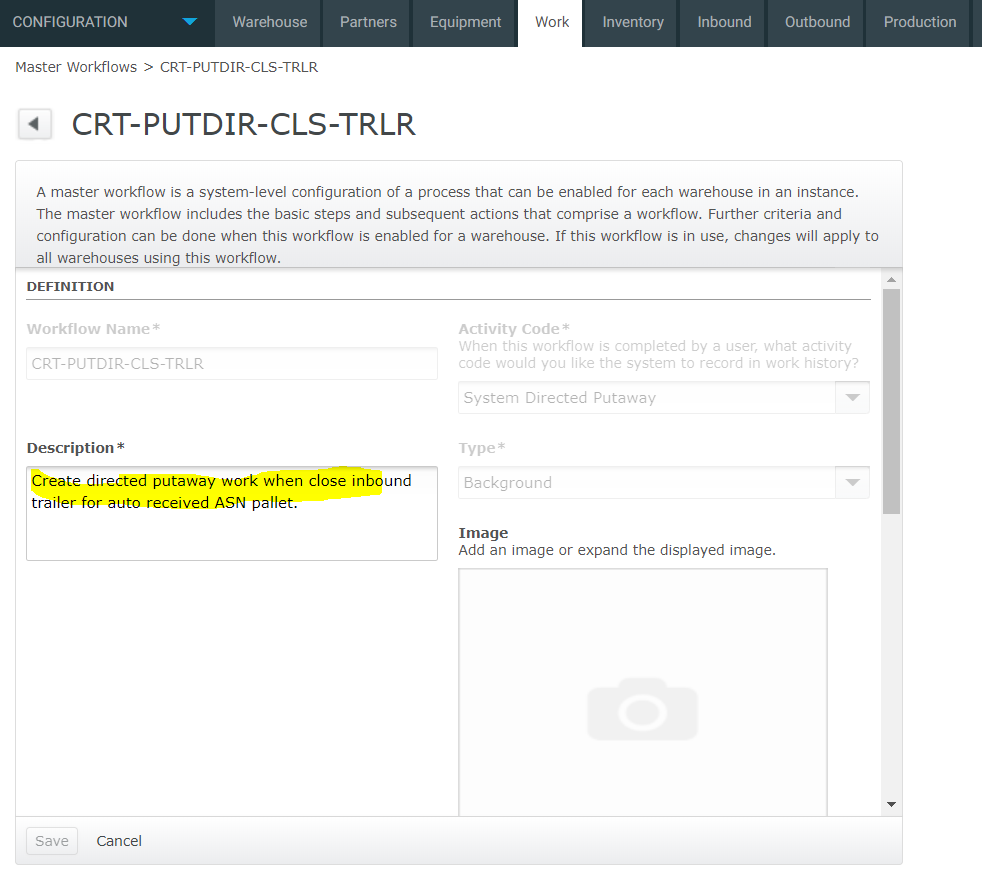
We can use ‘TRLRCLOSED’ background workflow to address the issue.

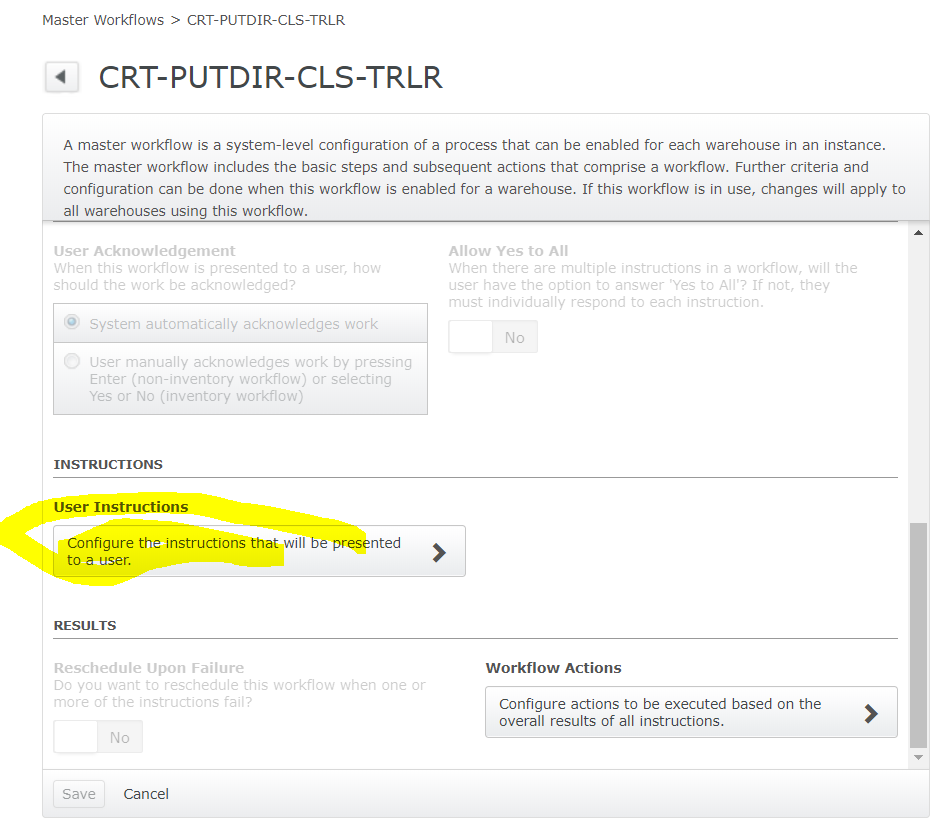
## Workflow configuration:

1. go to Master workflow to create a new workflow, named CRT-PUTDIR-CLS-TRLR:



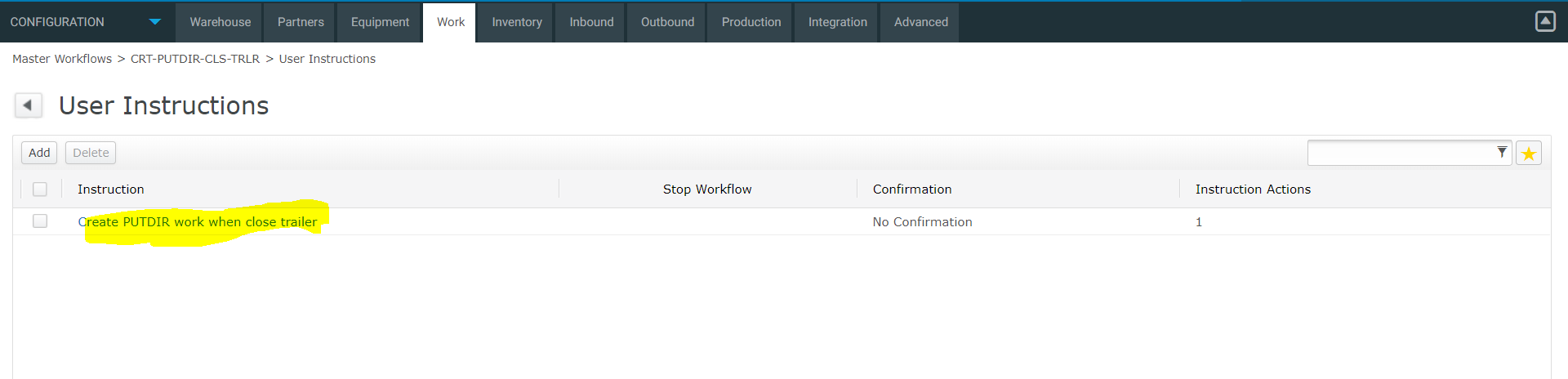
2. detail about CRT-PUTDIR-CLS-TRLR:



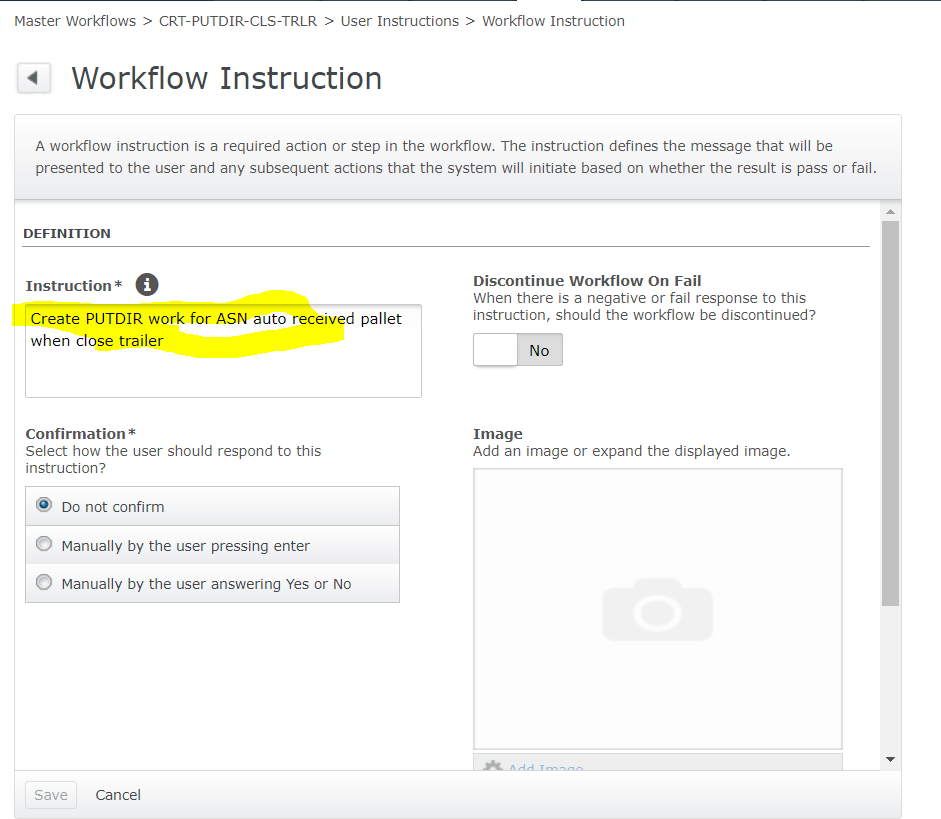


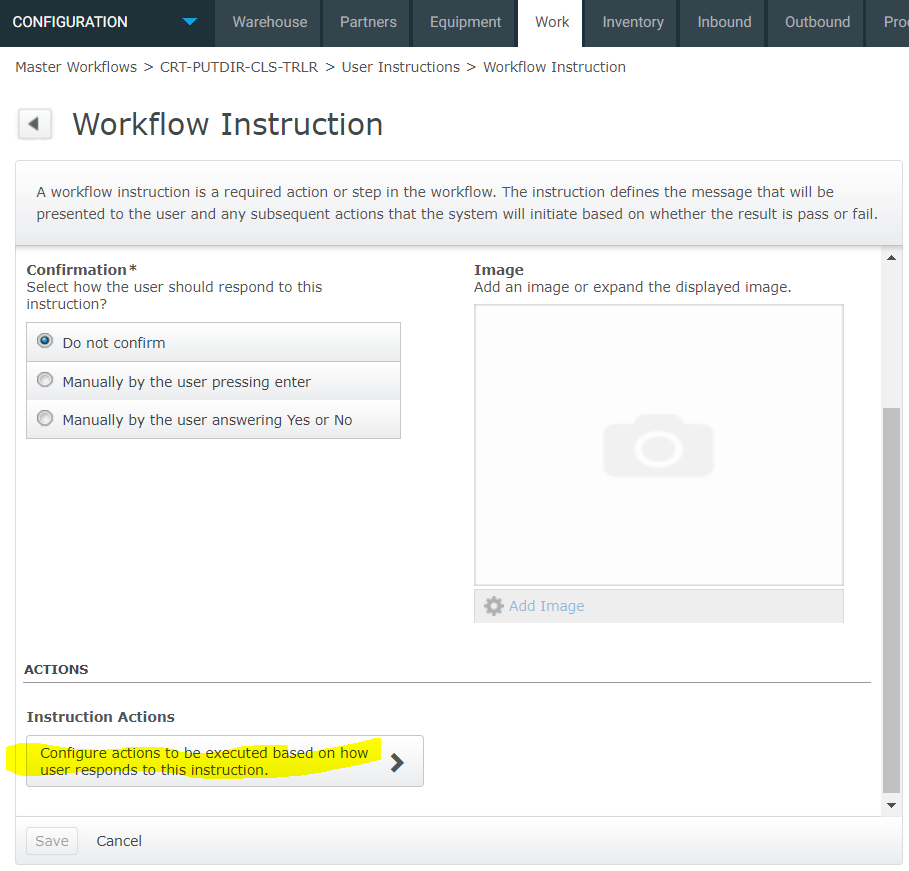
3. click ‘User Instructions’:

Add a config:



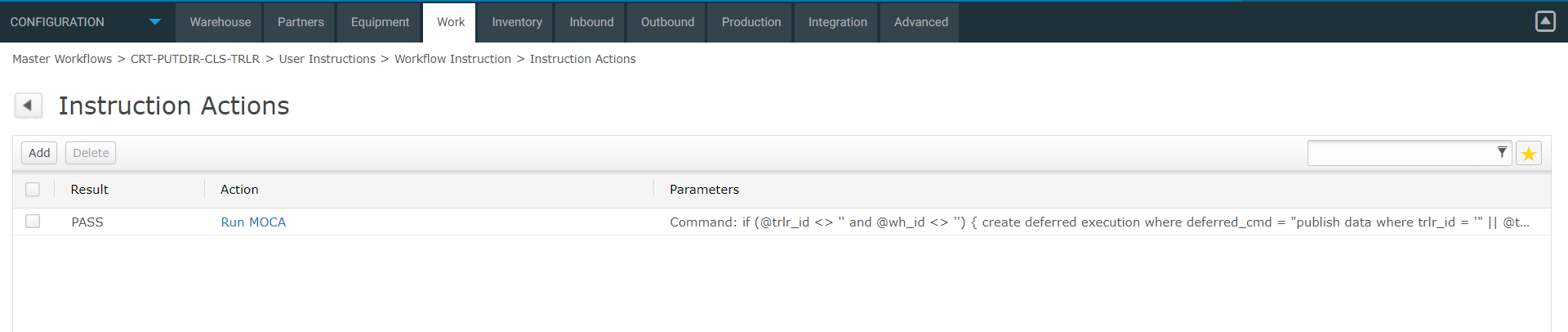
With detail:





4. Click ‘Instruction Actions’:

Add a new config:



The code to run as:

if (@trlr\_id <> '' and @wh\_id <> '')

{

create deferred execution

where deferred\_cmd = "publish data where trlr\_id = '" || @trlr\_id || "' and wh\_id = '" || @wh\_id || "'

|

/\* If this trailer has ASN pallet sits in receive stage \*/

[select distinct iv.stoloc srcloc

from inventory\_view iv

join locmst lm

on iv.stoloc = lm.stoloc

and iv.wh\_id = lm.wh\_id

join loc\_typ lt

on lm.loc\_typ\_id = lt.loc\_typ\_id

and lm.wh\_id = lt.wh\_id

and lt.rcv\_stgflg = 1

join rcvlin rl

on iv.rcvkey = rl.rcvkey

and iv.wh\_id = rl.wh\_id

and iv.asnflg = 1

join rcvtrk rk

on rk.wh\_id = rl.wh\_id

and rk.trknum = rl.trknum

where rk.wh\_id = @wh\_id

and rk.trlr\_id = @trlr\_id] catch(-1403)

|

if (@? = 0)

{

[select count(lodnum) tot\_lod\_cnt

from invlod

where stoloc = @srcloc

and wh\_id = @wh\_id]

|

/\* Check if there is sufficient PUTDIR work for these pallets, if not

\* we create PUTDIR work as needed.

\*/

[select count(reqnum) exist\_wrk\_cnt,

@srcloc srcloc

from wrkque

where srcloc = @srcloc

and wh\_id = @wh\_id

and oprcod = 'PUTDIR']

|

if (@exist\_wrk\_cnt < @tot\_lod\_cnt)

{

do loop

where count = @tot\_lod\_cnt - @exist\_wrk\_cnt

|

create work

where wh\_id = @wh\_id

and oprcod = 'PUTDIR'

and lodlvl = 'L'

and srcloc = @srcloc

}

}"

}

Note to above logic:

This logic creates PUTDIR work based on:

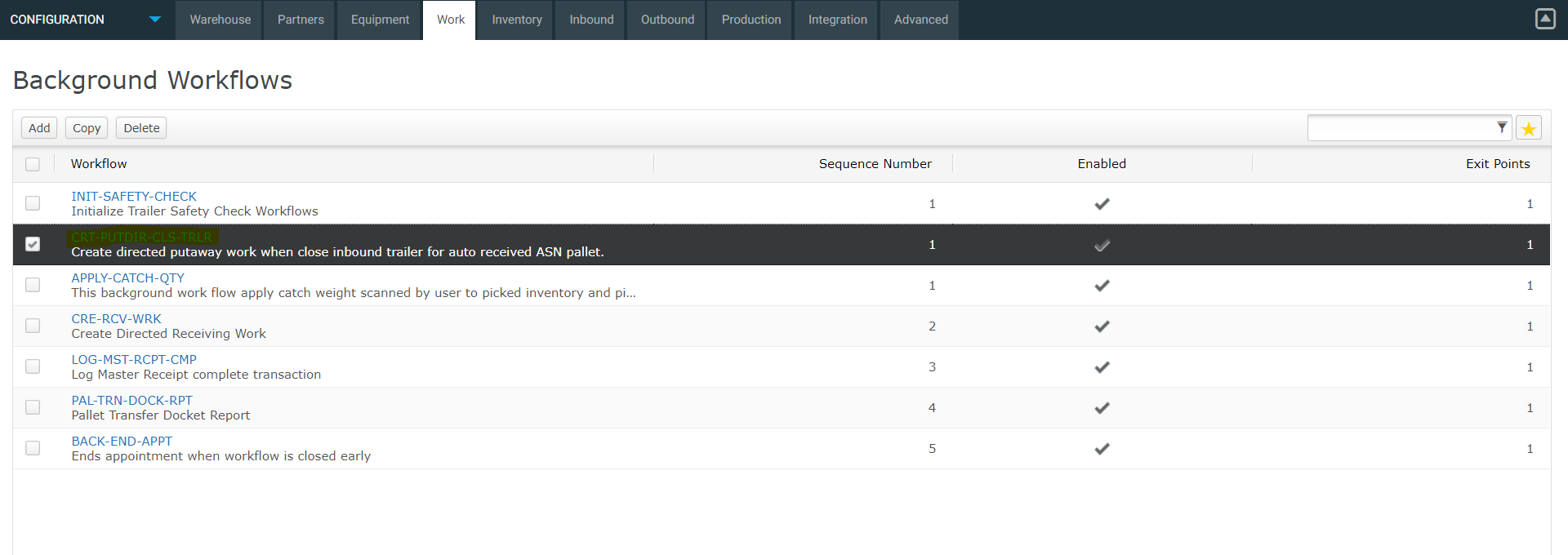
**1. Get location where have ASN pallet deposited.**

**2. Get total pallet that sits in receive stage lane.**

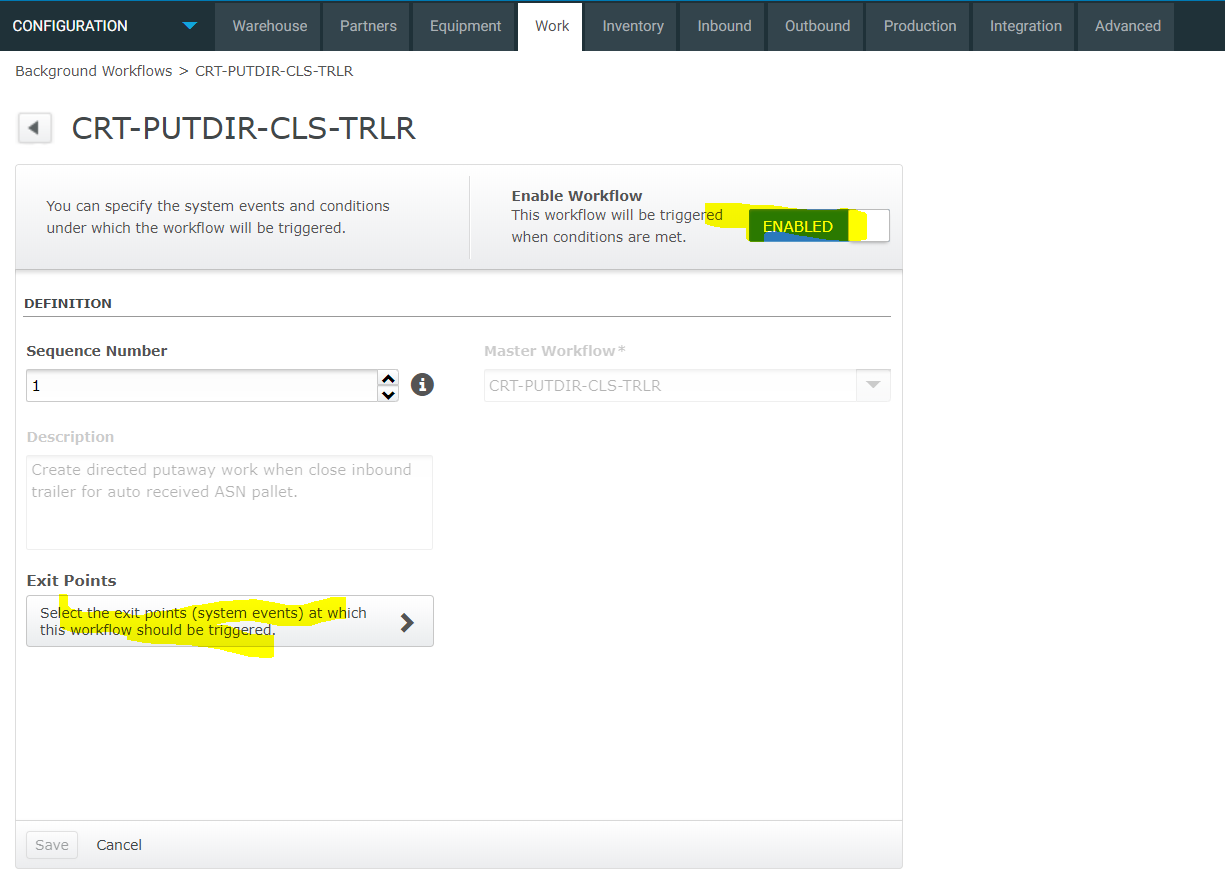
**3. Calculate how many work still needed by using the number of pallets – the number of work available (same as standard logic).**

**4. Create a deferred execution command to create the work.**

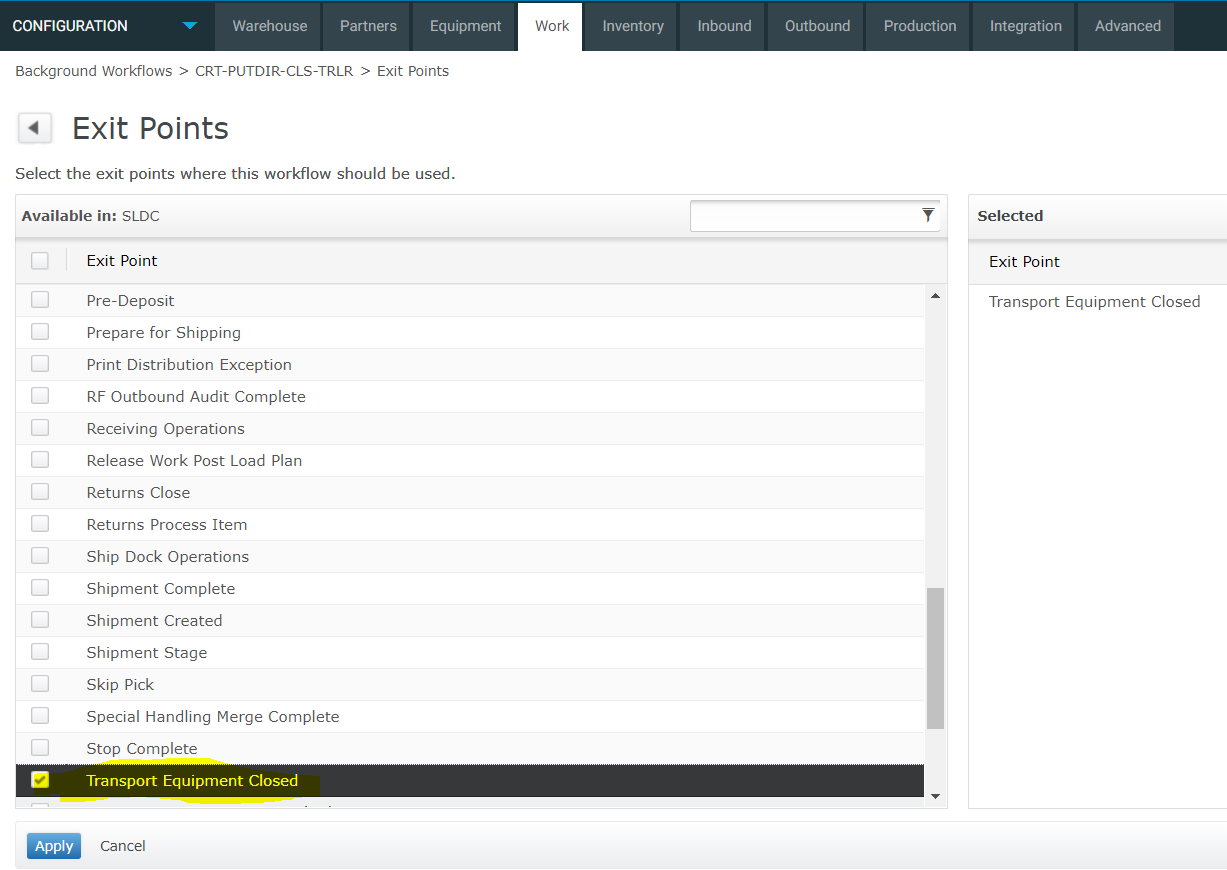
5. go to background workflow to add a new record:



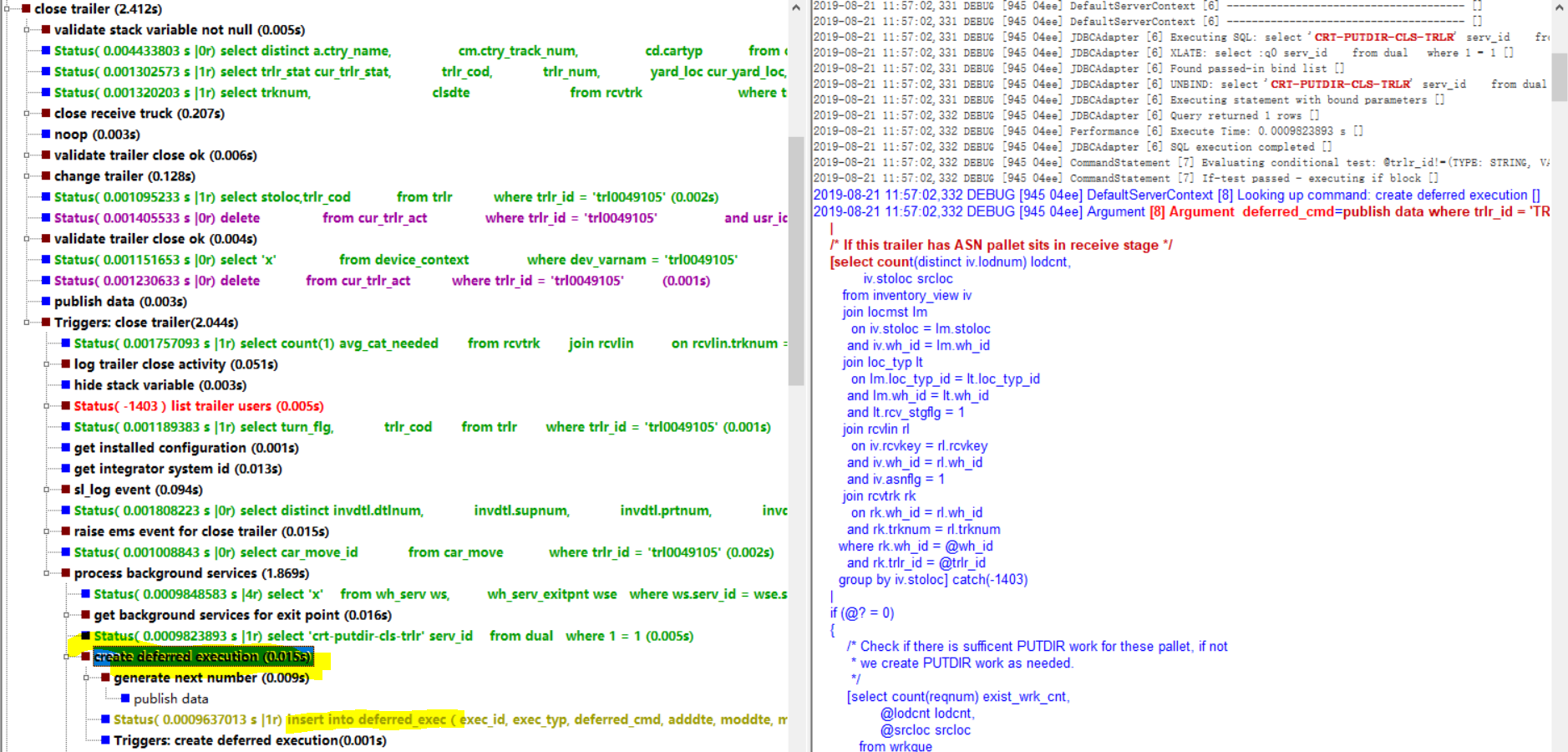
6. Detail page:



7. click ‘Exit Points’:



8. a snapshot from local test:



Trace of the deferred execution:

