AAPIJOB.PROCESSJOBQUEUE

Job verwerkt opdrachten uit de tabel ATJOBQUEUE. Vanuit het applicatie-proces worden hier PART-NO in weggeschreven die verwerkt moeten worden. Per **PART-NO** komt er een apart JOB in deze tabel.

Name Null? Type

------------- -------- -----------------

JOB\_ID NOT NULL NUMBER

JOB\_TYPE NUMBER(4)

JOB\_CREATED DATE

JOB\_STARTED DATE

JOB\_FINISHED DATE

PART\_NO VARCHAR2(18 CHAR)

REVISION NUMBER(4)

USER\_ID VARCHAR2(8 CHAR)

RESULT\_CODE NUMBER(8)

PARENT\_JOB\_ID NUMBER

Er komen meerdere JOBS in deze tabel (vanuit AAPIJOBDELEGATES aangestuurd):

JOBTYPE\_FINALIZE CONSTANT NUMBER := 0;

JOBTYPE\_CLEAN CONSTANT NUMBER := 1;

JOBTYPE\_CATIA CONSTANT NUMBER := 2;

JOBTYPE\_MESH CONSTANT NUMBER := 3; TODO

JOBTYPE\_CCODE CONSTANT NUMBER := 4; TODO

JOBTYPE\_TOSUBMIT CONSTANT NUMBER := 5;

JOBTYPE\_FEA CONSTANT NUMBER := 8;

aapiJob.ProcessJobQueue;

MODULE: DBMS\_JOB  
ACTION : 1274

AAPIUTILS.startalldbjobs

aapiJob.StartJobQueue()

LOOP AtJobQueue (select job\_id from **atJobQueue** where **job\_started is null**)

END LOOP AtJobQueue

ProcessJob(job\_id)

COMMIT

iapiGeneral.SetConnection(psJobName=JOBQUEUE)

StartJob(anJobID=>anJobID,anJobType=>lnJobType,asPartNo=>lsPartNo,anRevision => lnRevision)

aapiJobDelegates.ProcessJob(anJobID=>anJobID, anJobType=>lnJobType, asPartNo => lsPartNo, anRevision=>lnRevision)

COMMIT

iapiGeneral.GetLastErrorText();

SetJobResult(anJobID, lnRetVal);

UPDATE atJobQueue SET **job\_finished = SYSDATE**,result\_code = anResult WHERE job\_id = anJobID;

dbms\_alert.signal(psJobName, anJobID);

COMMIT

Select count(\*) from dba\_jobs

where what='aapiJob.ProcessJobQueue'

Indien count=0 DAN dbms\_job.submit(job => lnJob  
, what=>'aapiJob.ProcessJobQueue',  
next\_date=>SYSDATE, interval => 'SYSDATE + 1/24/60'

**Aapijob.CreateJob**(anJobId, sPartNO  
,anRevision,anJObType,anPartentJobId

ATJOBQUEUE

INSERT INTO ATJOBQUEUE

INTERSPEC

Update ATJOBQUEUE set **JOB\_STARTED=SYSDATE** where job\_id = anJobID

IF anJobType = JOBTYPE\_FINALIZE THEN

lnRetVal := ProcessFinalizeJob(anJobID, asPartNo, anRevision);

ELSIF anJobType = JOBTYPE\_CLEAN THEN

lnRetVal := ProcessCleanJob(anJobID, asPartNo, anRevision);

ELSIF anJobType = JOBTYPE\_CATIA THEN

lnRetVal := aapiCatia.CreateCatiaDatafiles(asPartNo, anRevision);

ELSIF anJobType = JOBTYPE\_MESH THEN

~~lnRetVal := iapiConstantDBError.DBERR\_JOBNOTFOUND; --TODO~~

ELSIF anJobType = JOBTYPE\_CCODE THEN

~~lnRetVal := iapiConstantDBError.DBERR\_JOBNOTFOUND; --TODO~~

ELSIF anJobType = JOBTYPE\_TOSUBMIT THEN

lnRetVal := ProcessToSubmitJob(anJobID, asPartNo, anRevision);

--~~ELSIF anJobType = JOBTYPE\_CANCEL THEN~~

~~-- lnRetVal := ProcessCancelJob(anJobID, asPartNo, anRevision);~~

ELSIF anJobType = JOBTYPE\_FEA THEN

lnRetVal := aapiFea.CreateMaterialFiles(asPartNo, anRevision);

END IF;