## Proposed

File watcher on these file types.

### Request Files

Cycle: 99

Month: 66

Year: 2033

Rerun: 4

**Type: BillCycle\_cfg**

Parallel: 16

### MAPS

The codes as per map are:

BillConfirm\_cfg BCF C

BillConfirmNonBT\_cfg NBC T

BillCycleRerun\_cfg BCR X

**BillCycle\_cfg** **BLC** **B**

BillPreRerate\_cfg BPR R

BillPrep\_cfg BLP P

BillQA\_cfg BQA Q

BillUndoS\_cfg BUS S

BillUndo\_cfg BLU U

BillProd\_cfg BPD D

Create a directory:

C99\_M66\_Y2033\_R4\_P16\_BLC

## Logical Job Definition Only

### OPPROG

The main job definition information

* job\_name
* description
* app\_script - command
* pre\_run - has to be successful meaning new ctm job
* post\_run – only run on job success meaning new ctm job
* post\_error – only run on job failure (new ctm job)
* login\_name – default = oper, not in use at Rogers
* app\_job\_own – who owns the batch (application)
* active\_ind – can be run
* default\_run\_time

### OPPROG\_DB

Pairs to OPPROG

* job\_name
* prefix - CAN
* request\_flag – has to be set to 1
* run\_id - ??

### JOBDBCONNECT and DBCONFIG

DBCONFIG contains database specific information that jobs can connect to. JOBDBCONNECT Correlates Job Name and Databases job should connect to by the DB\_CONNECT\_CODE

* job\_name
* market – correlates to prefix in OPPROG (CAN)
* run\_mode – replicating the property for &source\_job variable?
* application\_id
* dl\_server\_code – similar to app\_job\_own from OPPROG
* db\_connect\_code – allows us to use &source\_job value and replicate values into columns of new entry

### OP\_APP\_DESC

Only used if the job requires parameters – this table describes the parameters that are available to be used in the data table.

APP\_DESC holds the structure of the parameters. APP\_DATA holds the values.

* table\_name – refers to job name
* struct\_seq\_num
* max\_rec\_num
* field\_name, field\_type (ie int/varchar/date), field\_size

### OP\_APP\_DATA

* table\_name – job name
* JOB\_REC – from OPPAR
* struct\_seq\_num
* field\_seq\_num – parameter name (field\_name)
* PHASE\_NUMBER – parameter type (field\_type)
* DATA – actual data of the parameter, less than allowed size from field\_size

## Job Instance Definition

### OPPAR

Similar to OPPROG – describes instance of a job – has JOB\_REC (same jobname can have multiple JOB\_REC (JobName and JOB\_REC are PRIMARY KEYS)

* job\_name
* job\_rec -> ByRec, ENDDAY, ENDWEEK(1/2), ENDMONTH(01/02) -> JOB GROUPINGS
* run\_mode -> 0 in batch mode, 1 run immediately
* run\_time -> when the job should be run (see table – AMDOCS operational)
* cycle\_key – used when job recs (of the same job) run on different CPU’s – I think HOST OVERRIDE

#### RUN TIMES

Available RUN\_TIMES

0 – By request

1 - Daily

2 - Weekly

4 - Monthly

h – Short EOD

r - Billing - Pre-Rerate

p - Billing - Preparation-Production

d - Billing - Production

b - Billing - Full

c - Billing - Confirmation

u - Billing - Undo

s - Billing - Selective Undo

x - Billing - Full-Rerun

q - Billing - QA Only

t - Billing – Confirmation-NonBT

The configuration for these modes is stored in :

$TLG\_ETC/.Screens.ini

and is used by the Scheduler APIs ... (V21 – only the Billing ones are in use)

### OPPAR\_DB

Similar to OPPROG\_DB but for the instance of the job (Active Job)

* job\_name
* job\_rec – same as the job\_Rec is OPPAR
* request\_flag – operation status of the job
* run\_id - ??

#### REQUEST\_FLAG

0 – not requestd

1- Requested (wasn't picket YET by the scheduler)

2 – Picked by the scheduler (wasn't started yet)

3 – The JOB\_REC is running

4 – The JOB\_REC has finished SUCCESSFULLY

5 – The JOB\_REC has FAILED

### OPHOST

Which Host (CPU) a ‘FAMILY’ (Job Name) should run.

* Usually all JOB\_RECS of JOB NAMES run on the same host
* CYCLE\_KEY is used to differentiate where these jobs run from
* OPPAR\_CYCLE\_KEY in OPPAR will reference this CYCLE\_KEY

Fields

* job\_name
* bb\_prefeix
* cycle\_key – see above/below
* cpu\_id – where does the job run
* login\_name – user name override

#### CYCLE\_KEYS

NULL - Default

C# - Billing Cycle Code, represents a #

USG# - Usage Server 1 or 2

US# - Usage Server 1 or 2

U# - Usage Server 1 or 2

## Queries

### Job Definitions

Retrieve run information from logical tables.

select \* from opprog where opprog\_job\_name like '**< JOB NAME >**';

select \* from opprog\_db where opprog\_db\_job\_name like '**< JOB NAME >**';

select \* from ophost where job\_name like '**< JOB NAME >**';

select \* from jobdbconnect where job\_name like '**< JOB NAME >**';

Getting distinct default start times

select distinct opprog\_default\_run\_time from opprog;

Getting distinct run\_id’s (not sure what these are)

select distinct opprog\_db\_run\_id from opprog\_db;

Getting distinct operator’s (job creators)

select distinct operator\_id from ophost;

Getting distinct run modes – may be several

select distinct run\_mode from jobdbconnect;

## Billing Only (Not Control-M)

* OPPROG\_SHC
* OPPROG\_SHCFOL
* OPPAR\_SCH
* OPPAR\_SCHFOL

## Destination Tables

* OPDIST – container for structure of destinations (ftp/sftp/print)
* OPCONFIG – Data within the structure defined on OPDIST