XT Upgrade process

**PACKAGE INFORMATION:**

Changes are made and captured. Each U\*\*\*\*\* package has a list of Jira issues that the package fixes.

Packages include:

1. **Configuration changes**

**Configuration artifacts (in node001\delta\data)**

**TASK\_CONTEXT\_COMN.txt** will be used to update the table of the same name.

Fields include:

**image\_id** (I01/ in my db, unsure why),

**task id** (XT, as XT process is updating the table),

**context\_type** (?),

**context\_key**, (PRULES, SRULE, presumably P for prod environment, s for s environment?)

**context\_data** (version and package number)

**Database table and column changes**

Txt files listed in CXOLCRDL.txt template will be used to update database

1. **Various Template Changes** as needed in current package.

**Template artifacts located in node001\delta\source**

**CXOLCRDL.txt** contains database changes.

Fields include:

**First byte**: # install and replace, @ install,~ replace will remove all rows

**Table**

**Member**

**Key length**

**U\*\*\*\*\*.txt** contains database ALTER statements (copied from project\Dn\Platform\Common\Rule)

1. **Server code changes**

**Server code changes located in node001\delta\bin**

1. **Rules changes**

**Rules artifacts (in node001\delta\rules)**

**ALLRULES.txt** contains updates to EMS Rules tables

**PROCESS INFORMATION:**

The customer will download the package from the FIS secure ftp server. They will need to decrypt using a SHA-512 hash value. The customer may wish to upgrade custspec and sitespec. The customer will port changes to their appropriate application server.

The customer will run osql -S {Database server} -U *userid* -P *password* -i U\*\*\*\*\*.txt -o U\*\*\*\*\*\*.stdout -e -m -l to apply database schema changes.

To apply database configuration or rule changes, the customer will, from the mmc console, select DataNavigator XT : Reset : CR : Upgrade.

This process will first update tables with values from node001\delta\data, then update rules from node001\delta\rules\ALLRULES.txt.

**Configuration Data Process**

XT will call CommonTableLoad->upgrade (XF04). If this flow is not successful, the database will be rolled back.

upgrade() will read CXOLCRDL.txt for a list of tables, members, and key lengths that it will cross reference in DDATA (set in XT.txt) in the modify() method to identify changes that will be made.

modify() will attempt to open each CXOLCRDL.txt member field’s corresponding file in node001\delta\data.

modify() will then check to make sure that the delta\data\table.txt file is properly sorted from least to greatest. Lines must be sorted as they will be compared later.

modify() checks the database catalog column information for the table being changed against its listing in alpha\control.

modify() compares, line by line, information contained in alpha\data\table.txt with delta\data\table.txt. These files were previously sorted, so when lines are equal for key length, update the database using an update statement in CommontableSegment.loadRow() using the delta value

When alpha is less than delta, delete the row, corresponding to the alpha value, in the appropriate table

Else, add an entry with an insert statement to the table.

**Rules Process**

upgrade() will call EMSRuleSet->upgrade (XF08). If this flow is not successful, the database will be rolled back.

upgrade() checks the database for MAX EMSR\_DATA\_DEF\_ID, EMSR\_ACTION\_INSTR, EMSR\_DATA\_GRP\_ID, EMSR\_DEFINED\_VALUE to use as base to set new ones

upgrade() reads CXOXEMDL.txt , directed to location by DFILES SCRIPT \*\Source\%member.txt in XT.txt. CXOXEMDL gives members which are passed to load()

load() opens ALLRULES.txt in the DRULES location provided in XT.txt

load() processes rules listed in ALLRULES.txt

load() inserts the rules listed in ALLRULES.txt into appropriate tables

upgrade() moves members from delta to alpha