Analyse the content of a package

- [SCMBeans] |
- [ZMF Administrator] |
- [1.0-Orval]

Release info**Release date:** Mon, 26/09/2011
Applicable ChangeMan ZMF: Any ChangeMan ZMF release

Many ChangeMan ZMF processes (like checkout, promote, install, etc) perform **special processing** for specific library types, and this via special jobsteps. Quite often this is implemented using all sorts of ISPF skeleton customisations. That's why in skeletons like CMN20, CMN30, CMN50, CMN$$RPM, etc you will find a lot of coding with lots of ')SEL' and ')DOT' statements, to process specific library types. As time goes by, more and more of such coding is added, which makes those skeletons hard to maintain, and which also makes performance (of FTINCL) go down (not to speak about increasing CPU resources). So what can you do to keep control over this type of skeleton customisations?

Here are a few samples of such **special processing**:

- if a package contains DDL components, then include a number of extra jobsteps in promote, demote, install or backout, where these extra jobsteps invoke a 3rd party software to correctly process these DDL components in the selected target DBMS system.
- during a (batch) checkout of QMF components, also perform an upload of the selected component (from baseline 0, -1 or a promotion environment) into a target development environment or subsystem.
- if a package contains ABC components, then add an extra jobstep in the ChangeMan ZMF audit job to perform some custom audit processing.

All samples above can be implemented in ChangeMan ZMF fairly easy. However, the skeleton coding to test if such special processing for a specific library type is needed, isn't always obvious (and usually a pain for ChangeMan ZMF newbies ...). This because variable names or ISPF table names vary depending on the specific process to be tailored. Here are 2 samples to illustrate this (out of dozens of similar samples ...):

- The **variable name** of a component library type is as follows:
  - &CMPTYPE (for stage or checkout).
  - &LIBTYPE (for promote or demote).
  - &STGLTYP (for distribute, install, baseline ripple and backout).
- The **ISPF table** name containing the above mentioned variable name for library types:
  - &CMPMBRTB (for checkout).
  - none (for stage, such table does NOT exist).
  - RPMLIBTB (for promote or demote).
  - &RMLBTBL (for distribute, install, baseline ripple and backout).

Consider the approach to address this issue as documented in the Z-Clues (login required).

Source URL (retrieved on *2026-11-04 04:04*):
http://dr.chgman.com/z-factory/z-issues/scmbeans/s008

Copyright © 2000-2025 AbitMORE ® BV. All rights reserved

Copyright © 2000-2025 AbitMORE ® BV. All rights reserved