**Table of Contents**

[JFrog 2](#_Toc67643778)

[Sybase REP Server 3](#_Toc67643779)

[Sybase ASE 4](#_Toc67643780)

[BCP 5](#_Toc67643781)

[icaccControl Script 6](#_Toc67643782)

[CMS Set Proxy: 9](#_Toc67643783)

[EXCEL: 10](#_Toc67643784)

[JQL: (Jira Query Language) 11](#_Toc67643785)

[Other Reference: 11](#_Toc67643786)

# JFrog

**Upload files to artifactory**

|  |
| --- |
| jfrog rt u "\*" "int\_cmspromote\_generic/EAP/2021/2021-05/" --flat=false --recursive=true |

|  |
| --- |
| **/\* Template Path \*/**  http://idfartif:55742/artifactory/int\_cmspromote\_generic/EAP/2021/2021-05/EAR/cms-bedpanel-main-app-1.0.9.ear |

# Sybase REP Server

**Connect to REP server (cp850 rep server)**

|  |
| --- |
| isql -Ucs9sa\_tmp -Pcms\_tmppwd -SHA\_CDC26\_RT1 |

**To view REP server log content**

|  |
| --- |
| tail -f /appl/rs/home/dbmscfg/HA\_CDC26\_RT1.log |

**Check server running status**

|  |
| --- |
| admin who\_is\_up  go |
| admin who\_is\_down  go |

**Resume/ Suspend rep server**

|  |
| --- |
| suspend connection to <ASE>.<DB>  go |
| resume connection to <ASE>.<DB>  go |
| eg.  resume connection to CMS\_V2CORP\_ST1\_SB.econsent\_db  go |
| **/\* Skip transaction, Execute transaction \*/**  resume connection to data\_server.database [skip transaction | execute transaction] |

**\*\*\*Alternative log in for (utf8 rep server) \*\*\***

|  |
| --- |
| **/\* Connect to REP server (utf8 rep server) \*/**  isql -Ucmscorp2 -PPpmadm09~ -SHA\_TST02\_RT4 |

|  |
| --- |
| **/\* To view REP server log content \*/**  tail -f /appl/rs/home/dbmscfg/HA\_TST02\_RT4.log |

|  |
| --- |
| isql -Ucs9sa\_tmp -Pcms\_tmppwd -S**HA\_CDC26\_RT1**  isql -Ucs9sa\_tmp -Pcms\_tmppwd -S**HA\_TST02\_RT4** |
| **Check rep/sub/connection** |
| chk\_que.sh -rs HA\_TST02\_RT4  chk\_que.sh -rs HA\_CDC26\_RT1 |
| chk\_repsub.sh -all HA\_CDC26\_RT1 CMS\_V2CORP\_ST1\_LN corp |
| chk\_repsub.sh -all HA\_CDC26\_RT1 CMS\_VH2\_ST1\_LN cms |
| chk\_ln.sh -detail HA\_CDC26\_RT1 ALL > check\_ln.log |
| chk\_ln.sh -detail HA\_TST02\_RT4 CMS\_V2CORP9\_ST2\_SB |
| **View replication server log** |
| tail -f /appl/rs/home/dbmscfg/HA\_CDC26\_RT1.log  tail -f /appl/rs/home/dbmscfg/HA\_TST02\_RT4.log |
| **Check REP SUB DESP** |
| isql -Urscheck -S**RSSD\_CDC26**\_ST1 -Prscheck  rs\_helprep <repdef name>  rs\_helpsub <repsub name> |

# Sybase ASE

Dump transaction log to free log space

|  |
| --- |
| print ‘before image’  go  use <DB>  go  sp\_helpdb <DB>  go  use master  go  **/\* [ with truncate\_only] dump transaction without making copy \*/**  dump tran <DB> with truncate\_only  go  print ‘after image’  go  use <DB>  go  sp\_helpdb <DB>  go |

# BCP

bcp [[database\_name.]owner.]table\_name [:[ partition\_id | slice\_number ] |

partition partition\_name] {in | out} [datafile]

**About BCP –t –r:**

|  |
| --- |
| * **bcp** in native format for data recovery or salvage or to resolve an emergency situation. * **bcp** in native format to transport data between different hardware platforms, different operating systems, or different major releases of Adaptive Server. * **Field terminators (-t) or row terminators (-r) with bcp in native format.**   Results are unpredictable and data may become corrupted.  Using **bcp** in native format can create flat files that cannot be reloaded into Adaptive Server and it may be impossible to recover the data. If you cannot rerun **bcp** in character format (for example, a table was truncated or dropped, hardware damage occurred, a database was dropped, and so on) the data is unrecoverable. |

Ref: <http://infocenter.sybase.com/help/index.jsp?topic=/com.sybase.infocenter.dc30191.1573/html/utilityguide/X14951.htm>

# icaccControl Script

Login to eaptst72j (12C Admin Server)

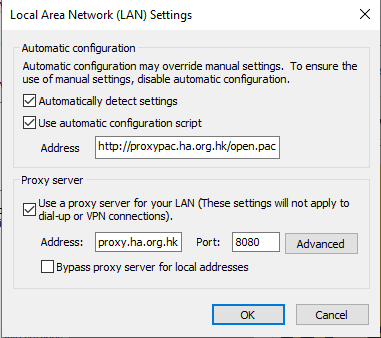
|  |
| --- |
| Username: cmsicacc  Passowrd: Fpsicacc@Dec  \*\*PS: password will be updated regularly |

|  |
| --- |
| export wlst\_username=’wls user-name’  export wlst\_password=’wls-password’  \*\*PS: please modify the value in ***Notepad*** |

|  |
| --- |
| icaccControl **ENV DOM PROJ FUNC func\_arg1 func\_arg2 … func\_argN**  **/\* <12C DOM> \*/**   1. WLS1 2. wls1\_cmspri\_ppm 3. wls1\_cmssec\_ppm 4. wls1\_cmstert\_ppm 5. wls1\_cmsmisc\_ppm 6. wls1\_cmsxpo\_ppm   **/\* Check Server Status \*/**  icaccControl PPM DC1 ALL getServerStatus  icaccControl PPM WLS1 ALL getServerStatus  **/\* Restart Servers \*/**  icaccControl PPM DC1 <PROJECT> restartServers eaptst2v,eaptst2w  icaccControl PPM <12c DOM> <PROJECT> restartServers eaptst72c,eaptst72d  **/\* CheckDS \*/**  icaccControl PPM dc1\_cms\_ppm ALL checkDS <DS Name>  icaccControl PPM <12C DOM> ALL checkDS <DS Name>  **/\* restartDS \*/**  icaccControl PPM dc1\_cms\_ppm ALL restartDS eaptst2v,eaptst2w <DS Name>  icaccControl PPM <12C DOM> ALL restartDS eaptst72c,eaptst72d <DS Name>  **/\* Add DS Targert (Not yet tested) \*/**  icaccControl PPM DC1 WLS\_DS\_TARGET <DS Name>  **/\* WLS\_DS\_TARGET UntargetDS \*/**  icaccControl PPM dc1\_cms\_ppm <PROJ> wls\_ds\_target unTargetDS <DS Name>  icaccControl PPM <12C DOM> <PROJ> wls\_ds\_target unTargetDS <DS Name>  **/\* WLS\_DS\_TARGET undeployDS \*/**  icaccControl PPM dc1\_cms\_ppm <PROJ> wls\_ds\_target unDeployDS <DS Name>  icaccControl PPM <12C DOM> <PROJ> wls\_ds\_target unDeployDS <DS Name>  /\* GETAPPINFO \*/  icaccControl PPM DC1 <PROJ> GETAPPINFO <APP\_NAME>  icaccControl PPM WLS1 <PROJ> GETAPPINFO <APP\_NAME>  **/\*** CREATE\_USER\_GROUP\_ROLE \*/  icaccControl PPM DC1 <PROJ> CREATE\_USER\_GROUP\_ROLE  /\* Start/Stop Admin Server \*/  /appl/oc/job/eap/util/scripts/opnsv2/wls\_control.sh cmsicacc start wls1\_cmsxpo\_ppm AdminServer |

|  |
| --- |
| -----------------------------------------------------------------  Currently Available Functions:  =================================================================  [1] CHECKDS  [2] CLASSPATH\_MOD  [3] CREATEDATASOURCE  [4] CREATEMULTIDATASOURCE  [5] CREATE\_USER\_GROUP\_ROLE  [6] DATASOURCENORMALSCANDAILY <<< Schedule Task = True  [7] DSCONFIGDASHBOARDUPDATE  [8] DSFULLSCAN  [9] DSINFO  [10] DSPROP\_MOD  [11] EMSSETALLDSINITOONE  [12] EMSSETALLDSINITOZERO  [13] FORCESHUTDOWNCLUSTERS  [14] FORCESHUTDOWNDS  [15] FORCESHUTDOWNSERVERS  [16] GENERATEFIREWALLEXCETIONLIST  [17] GETALLDSINICONTOCSV  [18] GETAPPINFO  [19] GETJMSSERVERSTATUS  [20] GETLIBINFO  [21] GETSERVERSTATUS  [22] INSEQROLLINGRESTARTCLUSTERS  [23] JAVA\_ARG\_MOD  [24] JMSMIGRATE  [25] LISTDATASOURCES  [26] LISTDATASOURCEWITHOUTTARGET  [27] REMOVEDATASOURCEWITHOUTTARGET  [28] RESETDS  [29] RESETJMSSERVERS  [30] RESTARTCLUSTERS  [31] RESTARTDS  [32] RESTARTSERVERS  [33] RM\_APP\_TARGET  [34] ROLLINGRESTARTCLUSTERS  [35] ROLLINGRESTARTCLUSTERSWITHMIGRATION  [36] SETDSCONFIG  [37] SHUDOWNDS  [38] SHUTDOWNCLUSTERS  [39] SHUTDOWNSERVERS  [40] STARTCLUSTERS  [41] STARTDS  [42] STARTSERVERS  [43] WLS\_DS\_TARGET |

# CMS Set Proxy:



|  |
| --- |
| /\* Shwo Proxy Setting \*/  netsh winhttp show proxy  /\* Import Proxy Setting from IE \*/  netsh winhttp import proxy source=ie |
| /\* NPM set Proxy \*/  **npm config** set proxy http://proxy.ha.org.hk:8080  **npm config** set https-proxy https://proxy.ha.org.hk:8080  **npm config** set https-proxy http://proxy.ha.org.hk:8080 |
| git config --global http.proxy http://proxyUsername:proxyPassword@proxy.server.com:port  git config --global http.proxy http://<username>:<password>@<proxy server:port> |
| PIP Install:  pip install --proxy http://llm234:85167787887Ss!@proxy.ha.org.hk:8080 pandas |

# EXCEL:

**Ref:** [**https://www.ablebits.com/office-addins-blog/2016/04/07/how-to-count-distinct-and-unique-values-in-excel/**](https://www.ablebits.com/office-addins-blog/2016/04/07/how-to-count-distinct-and-unique-values-in-excel/)

|  |
| --- |
| Count distinct values in Excel (unique and 1st duplicate occurrences)  To get a count of distinct values in a list, use the following formula:  **=SUM(1/COUNTIF(*range*, *range*))**  Remember, it's an array formula, and therefore you should press the Ctrl + Shift + Enter shortcut instead of the usual Enter keystroke.  Alternatively, you can use the SUMPRODUCT function and complete the formula in the usual way by pressing the Enter key:  **=SUMPRODUCT(1/COUNTIF(*range*, *range*))**  For example, to count the distinct values in range A2:A10, you can go with either:  **=SUM(1/COUNTIF(A2:A10,A2:A10))**  Or  **=SUMPRODUCT(1/COUNTIF(A2:A10,A2:A10))** |

|  |
| --- |
| **Count if contain a specific string**  **=Countif(range, “String”)** |

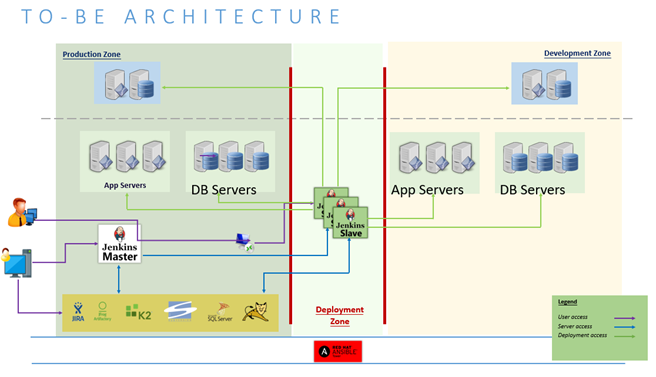
# JQL: (Jira Query Language)

|  |
| --- |
| **project = "IT Operation – Clinical Management System" AND "PPM Type" = Bi-Weekly AND summary ~ "PPM2021\_0\*" AND "PPM Revise counting" > 0 ORDER BY summary ASC** |
| **project = "IT Operation – Clinical Management System" AND "PPM Type" = Service AND summary ~ "PPM2021\_\*" AND "PPM Revise counting" > 0 AND "Ready for PRD Promotion" <= now() ORDER BY summary ASC** |

|  |
| --- |
| **project = "IT Operation – Clinical Management System" AND "PPM Type" = Bi-Weekly AND summary ~ "PPM2021\_04\*" AND "Promotion Documents" ~ "https://wfeng-svc" ORDER BY summary ASC** |
| **project = "IT Operation – Clinical Management System" AND "PPM Type" = Bi-Weekly AND summary ~ "PPM2021\_04\*" AND "Promotion Documents" ~ "https://hatool.home" ORDER BY summary ASC** |

# Other Reference:

**K2 E-form x Jenkins Pipeline Architecture**

****