2.7.11 Capture ALARM STATISTICS

Often it is necessary to capture alarm statistics from the DCS. Such a requirement may, for example, be required as part of an alarm rationalization project.

The following code facilitates capturing the number of acknowledged and unacknowledged alarms per alarm priority (emergency / high / low) to CDS parameters. The alarm data could then be accessed externally by mapping the CDS parameters to a historian or writing the alarm data from the CDS parameters to a text file on the HM which could then be file transferred to the required folder destination.

Alarm data is captured by accessing specific PDSP parameters of a US node. The PRSTS address of the US that will be referenced is set to parameter US_NOD. For example if the US node to be accessed is node number 03 then $PRSTS03 should be set to US_NOD.

```
PACKAGE
--=====================================================
--  PROJECT  -  TPS ALARM STATISTICS GENERATOR
--              ------------------------------
--
--  Block:      ALMSTAT1.CL
--
--  Function:   Determines the alarm statistics for an area
--              and writes these to CDS parameters.
--
--              This is achieved by pulling the required
--              alarm data from one US station so the alarm
--              data being collected depends on the area the
--              US is running.
--
----=================================================
--  CUSTOM DATA SEGMENT PARAMETER DEFINITIONS
--  ----------------------------------------
--
--    US_NOD     US node to reference PDSP parameters from.
--               Note that the format required is $PRSTSxx
--               where;
--
```

```
--               xx = US (GUS) node number
--
--    UAEM       Number of unacknowledged EMERGENCY alarms.
--
--    AEM        Number of acknowledged EMERGENCY alarms.
--
--    UAH        Number of unacknowledged HIGH alarms.
--
--    AH         Number of acknowledged HIGH alarms.
--
--    UAL        Number of unacknowledged LOW alarms.
--
--    AL         Number of acknowledged LOW alarms.
--
--    TOT_UN     Total number of unacknowledged alarms.
--
--    TOT_ACK    Total number of acknowledged alarms.
--
--    LAST_RUN   Time and date of the last block run.
--
--=======================================================
--  PARAMETER LIST
--  --------------

PARAM_LIST  US_ALMS

Parameter   ACKEM     :    NUMBER     Array (1..36)
Parameter   ACKHI     :    NUMBER     Array (1..36)
Parameter   ACKLO     :    NUMBER     Array (1..36)
Parameter   NAME      :    STRING
Parameter   UNACKEM   :    NUMBER     Array (1..36)
Parameter   UNACKHI   :    NUMBER     Array (1..36)
Parameter   UNACKLO   :    NUMBER     Array (1..36)

END US_ALMS

--=======================================================
--  CUSTOM DATA SEGMENT PARAMETERS
--  ------------------------------

CUSTOM (CLASS GENERAL  ;  ACCESS ENGINEER  ;  NOT BLD_VISIBLE)
```

```
Parameter           US_NOD      :     US_ALMS
BLD_VISIBLE

Parameter           UAEM        :     NUMBER
Value 0

Parameter           AEM         :     NUMBER
Value 0

Parameter           UAH         :     NUMBER
Value 0

Parameter           AH          :     NUMBER
Value 0

Parameter           UAL         :     NUMBER
Value 0

Parameter           AL          :     NUMBER
Value 0

Parameter           TOT_UN      :     NUMBER
Value 0

Parameter           TOT_ACK     :     NUMBER
Value 0

Parameter           LAST_RUN    :     TIME

END CUSTOM

--=======================================================
BLOCK   ALMSTAT1  (Generic; at Backgrnd(5) )
-------------------------------------------------------
%RELAX Linker_SDE_Checks
-------------------------------------------------------
--  LOCALS
--  ------

--  Local numbers.

Local   i                           --  loop counter
```

```
Local   loc_uaem                        --  unack E alarms
Local   loc_aem                         --  ack E alarms
Local   loc_uah                         --  unack H alarms
Local   loc_ah                          --  ack H alarms
Local   loc_ual                         --  unack L alarms
Local   loc_al                          --  ack L alarms
Local   loc_totun                       --  total unack alarms
Local   loc_totack                      --  total ack alarms

--  Local other types.

Local   current_time_date    : Time     --  current system time
                                        --  and date

------------------------------------------------------------
--  SET LOCAL PARAMETERS
--  --------------------

--  Store the current time and date.

    Set    current_time_date     =  DATE_TIME

--  Set alarm locals to zero.

    set  loc_uaem    =  0
    set  loc_aem     =  0
    set  loc_uah     =  0
    set  loc_ah      =  0
    set  loc_ual     =  0
    set  loc_al      =  0
    set  loc_totun   =  0
    set  loc_totack  =  0

--============================================================
--  EXECUTABLE CODE
--  ---------------
--  Exit if there are no US nodes configured.

     If   EQUAL_NULL_POINT_ID(US_NOD)  then  goto EXIT_SCHEME
     If   Not EXISTS(US_NOD.NAME)      then  goto EXIT_SCHEME

--  Store the alarm statistics for the area to locals.
```

```
ALM_STATS:       Loop for i in 1..36

                 Set  loc_uaem  =  loc_uaem + US_NOD.UNACKEM(i)
                 Set  loc_aem   =  loc_aem  + US_NOD.ACKEM(i)
                 Set  loc_uah   =  loc_uah  + US_NOD.UNACKHI(i)
                 Set  loc_ah    =  loc_ah   + US_NOD.ACKHI(i)
                 Set  loc_ual   =  loc_ual  + US_NOD.UNACKLO(i)
                 Set  loc_al    =  loc_al   + US_NOD.ACKLO(i)

END_ALM_STATS:  Repeat ALM_STATS

--  Now calculate the total acknowledged and unacknowledged
--  alarms for the area.

    Set   loc_totun   =  loc_uaem  +  loc_uah  +  loc_ual
    Set   loc_totack  =  loc_aem   +  loc_ah   +  loc_al

--  now store all to CDS parameters.

    Set   UAEM     =  loc_uaem
    Set   AEM      =  loc_aem

    Set   UAH      =  loc_uah
    Set   AH       =  loc_ah

    Set   UAL      =  loc_ual
    Set   AL       =  loc_al

    Set   TOT_UN   =  loc_totun
    Set   TOT_ACK  =  loc_totack
-------------------------------------------------------------
--  The program has finished so exit after storing the current
--  run time.

EXIT_SCHEME:
&
&   Set    LAST_RUN         =  current_time_date

    Exit
-------------------------------------------------------------
END  ALMSTAT1
END  PACKAGE
```