

Checking values of destination columns

The command enables to compare the values of target columns (of structured variables), configured in the eval tags, with the values that were calculated by [D2000 Calc](#). The calculation is done by calling the Tell command `CHECK_DESTID_VALUES [DestId [ColIndex]] [DETAIL] [FlagList]` in [D2000 Calc](#).

Warning: Tell command CHECK_DESTID_VALUES is supported only in the version D2000 V8.0.5.

When specifying the particular object of *Structured variable* type, for which the comparison should be done, the value of parameter *DestId*, or also *ColIndex* for the particular column, must be a non-zero. These parameters are followed by the optional key words (named as **FlagList** parameter) that enable to organize the final output.

Note: The term "value of eval tag" (or abbreviated "value") will be used below in terms of the value that is intended for calculation for row of target column in the structured variable. One eval tag has several values. It depends on the number of row of target column in the structured variable.

List of key words **FlagList** (written in consecutive order and separated by blank space, the detailed meaning is described in the article [Procedure to check the eval tags](#)):

Key word	Meaning
Detail	Displays detailed information about difference in the compared values (they contains error or special flags).
Debug	The same meaning as <i>Detail</i> , but when writing <i>ColIndex</i> , it displays also the compared value without flags.
Ignore_Time	Ignores the time stamps of the compared values (for SBS this option is always enabled).
ReCalc0s	Compares the values no matter when the eval tag was calculated. Default value is 10s.
ReCalc1m	Does not compare the values if eval tag was calculated before less than 1 minute and the response has not been received from D2000 System yet. Default value is 10s.

Procedure to check the eval tags

It checks only those eval tags with the defined reference to the target column of structured variable.

1. type of log (neither **Detail** nor **Debug** parameters are set)

When necessary, you may narrow the selection by defining *DestId* parameter for the structured variable, or *ColIndex* parameter for the particular target column.

Examples:

Example 1

```
Tell Command: CHECK_DESTID_VALUES
Start Time: 31.05.2013 15:41:44.763
C_Index;C_ID;C_Name;C_Type;D_Column;COUNT;NoEval;ReCalc;ErrInv;ErrCnv;ErrDif
1:10216;P.SV.ALARM.AKTIV_DRN;CHANGE;$14491^29;45;45;0;0;0;0
...
41:11715;P._SysProf.Server.ThreadAlarm;CHANGE;$11427^11;154;154;0;0;0;0
42:11717;P._SysProf.Server.UsedMemPerc;CHANGE;$11427^5;154;154;0;0;0;6
43:11719;P.SV.B.LFC_SED.TG_FILTER_T10_MINUS_ALARM;CHANGE;$12280^86;60;0;0;0;0;0
...
705:71184;P.TEST_XXX_Priznak;PERIODIC;$71179^2;2;0;2;0;0;0
Duplicated Destination Column: $11427^5 - 2 occurrences!
  Calc 'P._SysProf.Server.UsedMemPerc' with Type CHANGE
  Calc 'P._SysProf.Hpux.Server.UsedMemPerc' with Type CHANGE
Duplicated Destination Column: $13104^4 - 2 occurrences!
  Calc 'P.SIMU.AGC_Unit_LFC.Manual.PbAct' with Type CHANGE
  Calc 'P.SIMUL.AGC_Unit_LFC.PbAct' with Type CHANGE
SUMMARY: 705 Calc(s) with 23658 Shadow Calc(s) - 65 Error(s) and 11113 Flag(s)
```

At the beginning of the log there is always the called TELL command and time, when it was started. This is followed by the specific record, which contains header of CSV data. Just one row with data in CSV format is assigned to each eval tag with the defined target column. This row represents either debugging information (green color) or error message (red color).

Each eval tag contains a consecutive number for row (number of rows is equal to number of eval tags with the defined target column of structured variable), ID of eval tag, its name, type and filled target column in the format $\ColIndex . The special counters comes after this column in this order:

- **COUNT** - total number of values for given structured eval tag,
- **NoEval** - value that has not been calculated yet,
- **ReCalc** - value that was calculated before less than n seconds (default is n=10s, it can be change to 60s by *ReCalc1m* or disabled by *ReCalc0s*,
- **ErrInv** - any information about the given value has never been received from D2000 Server yet,
- **ErrCnv** - the recalculated value is invalid because there does not exist the conversion from the value that is calculated by D2000 Calc to value that is filled in structured variable because it is of other type,
- **ErrDiff** - the value that has been calculated by D2000 Calc is compared with the value which is registered in D000 Calc from D2000 System.

Rules:

- none of the counters can be greater than value in COUNT,
- the flags are evaluated sequentially from top to bottom,
- if the value has NoEval flag, it is also compared with the value from D2000 System (the value may have still ErrCnv flag or ErrDiff),
- if the value has ReCalc flag, other flags will not be evaluated,
- if the value has ErrInv flag, other flags will not be evaluated,
- if the value has ErrCnv flag, it compares the value registered in D2000 System and invalid value (the value can even acquire ErrDiff flag),
- if the value has ErrDiff flag, it means that the values differ and it is the main debugging error
- when comparing the values, there are ignored these flags:
 - whether the value is manual,
 - value for indirect,
 - if it is SBS or Ignore_Time has been entered, it ignores also the value time;
- the row in the log represents the debugging information if ErrInv, ErrCnv and ErrDiff are null, otherwise it represents the error message.

At the end of log there is a summary. Before starting the tell command with other parameters, you should check the error flags (cumulative ErrInv, ErrCnv and ErrDiff) and special flag (cumulative NoEval and ReCalc). If there are the eval tags that fill the same destination column of structure variable, a warning occurs even before this log.

2. type of log (Detail or Debug parameter is entered, but the particular destination column of structured variable is not defined)

Tell command without the parameters allowed to find out the number of error and special flags (see the summary at the end of log of 1. example). If their sum is convenient, because it defines the maximum number of rows of next log, the tell command without DestID can be called for the structured variable. In the output, the one row will correspond to each value of structured eval tag, which contains error or special flags. If it contains only special flag, it will be in the form of warning, but if it contains both error and special flags, the row will represent the error message.

Example 2

```
Tell Command: CHECK_DESTID_VALUES 11427 DETAIL
Start Time: 03.06.2013 11:02:55.233
CNT_NAME;D_VOBJ;FIRST_EVAL;C_ID;C_NAME;C_TYPE;CV_REDEF;CV_TYPE;CV_VALUE;CV_TIME;DV_UNDEF;DV_TYPE;DV_VALUE;
DV_TIMEID
CALC-ERRDIF;$11427[1]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.520
CALC-ERRDIF;$11427[2]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.520
CALC-ERRDIF;$11427[3]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.520
CALC-ERRDIF;$11427[4]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.520
CALC-ERRDIF;$11427[5]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.521
CALC-ERRDIF;$11427[6]^5;FALSE;11717;P._SysProf.Server.UsedMemPerc;CHANGE;FALSE;INT;(invalid);30.05.2013 09:55:
08.499;FALSE;INT;(invalid);30.05.2013 09:55:11.521
...
```

At the beginning of the log there is always the called TELL command and time, when it was started. This is followed by the specific record, which contains header of CSV data. A one-row record, which starts with flag that was evaluated as the last (it is the most important and starts with CALC- prefix) corresponds to each value that contains the error or special flags. This is followed by the destination value of structured variable in the format \$[RowNumber]^CollIndex, the information about a first evaluation of I/O tag (FIRST_EVAL: on SBS it is still True), its ID, name and type, whether the value is recalculated (CV_REDEF: on SBS it is still True after the first recalculation), type, value and date of recalculation, whether the destination value has never been received from D2000 System (DV_UNDEF), its type, value and time stamp.

3. type of log (Detail or Debug parameter is entered and the destination column of structured variable is exactly defined)

The output file will contain either the detailed information for each value of structured I/O tag, which contains the error flags, if *Detail* parameter is entered, or detailed information about all values if *Debug* parameter is entered.

Example 3

```
Tell Command: CHECK_DESTID_VALUES 11427 5 DETAIL
Start Time: 03.06.2013 12:21:34.864
CNT_NAME;D_VOBJ;FIRST_EVAL;C_ID;C_NAME;C_TYPE;CV_REDEF;CV_TYPE;CV_VALUE;CV_TIME;DV_UNDEF;DV_TYPE;DV_VALUE;DV_TIME
[Calc]
ID                : $11717[1]
FirstEval         : FALSE
bDestValUnDef     : FALSE
bDestValReClc     : FALSE
[Info]
Id               : 11717
Name            : P._SysProf.Server.UsedMemPerc
Descript        : Server - % vyuzitia pamate
ParentId        : 28
Value_Type      : CE
ActRefNr        : 1
PasRefNr        : 0
CreateTime      : (null)
```

```

    ModifyTime          : 24.08.2010 17:48:17.263
[Data]
    RECAL_REQ           : CHANGE
    PERIOD               : 0
    OFFSET               : 0
    START_VAL            :
    TriggerId           :
    TriggerTyp           : TVALUE
    InvertTrig           : FALSE
    MonotonicTime        : FALSE
    UTC_Offset           : 1
    DestId               : $11427^5
    IgnoreInvalid        : FALSE
    TimeFilter           : 0
[Alarm Data]
    Row                 : 0
    Col                 : 0
    Alarm               :
    Monitor              :
    KvitReq              :
    Critical             :
    Timeout              : 0
    Silent               : FALSE
    SignalMonitor        : FALSE
    SignalAlarm          : FALSE
    SignalCritical       : FALSE
    IgnoreInvalid        : FALSE
    ProcAlarmShowMasks :
[Value]
    GValTyp              : INT (Integer)
    Status               : NOACKVALUE
    LimitStatus          : INLIMIT
    ProcAlarmStatus      : NOALARM
    ValTyp               : CE
    Flags                : F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F
    ValTime              : 30.05.2013 09:55:48.296
    AlarmTime            : (null)
    Value                : (invalid)
[LastVal]
    GValTyp              : INT (Integer)
    Status               : NOACKVALUE
    LimitStatus          : INLIMIT
    ProcAlarmStatus      : NOALARM
    ValTyp               : CE
    Flags                : F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F
    ValTime              : 30.05.2013 09:55:08.499
    AlarmTime            : (null)
    Value                : (invalid)
[DestVal]
    GValTyp              : INT (Integer)
    Status               : NOACKVALUE, MANUAL
    LimitStatus          : INLIMIT
    ProcAlarmStatus      : NOALARM
    ValTyp               : INT
    Flags                : F,F,F,F,F,F,F,F,F,F,F,F,F,F,F,F
    ValTime              : 30.05.2013 09:55:11.520
    AlarmTime            : (null)
    Value                : (invalid)
LL_Stat                : TRUE
VLL_Stat               : TRUE
HL_Stat                : TRUE
VHL_Stat               : TRUE
Trigger                : FALSE
calcCounter            : 3
nvCounter              : 1
ShadowList             : 0
OnlyMyRow              : TRUE
CalcTime               : (null)
LastCalcTime           : (null)
bAllwaysInLimits       : TRUE
FirstAfterRDSw         : FALSE
ActEvalTime            : 30.05.2013 09:55:11.509
PocitajTime            : 30.05.2013 09:55:48.296
...

```