Structure of CMC-TC MIB

Since the CMC-TC system is a highly modular toolkit system that can be configured for specific applications, the structure of the MIB needed the ability to cover all conceivable possible combinations. Obviously this requires a certain degree of complexity. This document is designed to enable the user – starting from an existing configuration of a CMC-TC system – to determine the SNMP variables relevant to analysis.

First the most important SNMP variables of the MIB of the CMC-TC processing unit are described one after the other. This part serves to explain the meaning of the variables and enable one to understand the basic structure of the MIB. In the second part there is a list showing how many table entries are relevant for the various sensor units and which variables are relevant for the various sensor types. It also indicates particularities in the meaning of individual variables for the various units. At the end of the second part there is a list of the various sensor and output types. In the third part, an access unit and a PSM unit are used to once again explain the process for determining the relevant variables.

The document refers to the MIB Version 1.1b of the processing unit of 11 March 2004, which is used by the processing unit as of software version 1.20.

Part 1: Description of the SNMP variables

The following designations of the SNMP variables proceed from the OID `internet.private.enterprises.rittal.cmcTc`. The processing unit of the CMC-TC system can be identified uniquely by this OID. Only the section `mib-2.system` refers to the starting point `internet.mgmt`.

`mib-2.system`

Relevant are the variables `sysDescr`, `sysContact`, `sysName`, and `sysLocation`, which contain basic information on the processing unit. `sysDescr` is a unique designation made up of device type, serial number, hardware version number, and software version number. `sysContact`, `sysName`, and `sysLocation` are texts that can be specified by the user and serve to identify the device. Furthermore, the variable `sysUpTime` could be of interest, which contains the runtime of the device since the last new start in 100ths of a second.

`cmcTcMIBRev`

`cmcTcMIBMajRev` contains the main revision number of the MIB version; currently it is 1. `cmcTCMIBMinRev` contains the secondary revision number, which shows minor additions to the MIB; currently that value is also 1. Differences between Version 1.0

and 1.1 of the MIB are a few additional variables and a supplementary description of new sensor and status codes. This document will not deal with the differences in any further detail. It is recommended that one always use the most up-to-date software version of the processing unit. The variable `cmcTcMIBCondition` contains a status code that describes the status of the processing unit, corresponding to the status of the alarm LED on the device. If the variable has the value 5 (configChanged), it is necessary to first confirm the configuration change done on the system before the SNMP variables represent the status of the new configuration. After a configuration change for the system, the selected SNMP variables may need to be exchanged, since other variables can be relevant with new sensor units and changed sensors.

`cmcTcStatus`

Each of the four connectable sensor units has an area with the designation cmcTcStatusSensorUnitX; the ´X´ is to be replaced by a number from 1 to 4 – for the four possible sensor units. The four areas all have an identical structure; therefore in this document only the first of these areas is described; each of the others is to be handled identically. For a special module there is one more additional area `cmcTcStatusExtUnit`, which will also be described below.

`cmcTcStatus.cmcTcStatusSensorUnit1`

In `cmcTcUnit1TypeOfDevice` the type of the unit is encoded. Using this value it is possible to determine how many entries have to be analysed in the following tables. The figures are summarised further below. However, the number of table entries can also be called up elsewhere, so that this variable is not needed for this purpose. However, in part the different units have to be handled very differently in creating profiles. The peculiarities of each unit and the necessary differences in processing the unit types will be described in more detail in Part 2.

The tables in the areas `cmcTcStatusUnit1Sensors`, `cmcTcStatusUnit1Outputs`, and `cmcTcStatusUnit1Msg` contain all the data of the sensors and actors connected to the respective sensor unit. Common to all three areas is that at first there is one variable in which the number of existing table entries is stored. These variables are designated by the names `cmcTcStatusSensorUnit1Sensors.cmcTcUnit1NumberOfSensors`, `cmcTcStatusSensorUnit1Outputs.cmcTcUnit1NumberOfOutputs`, and `cmcTcStatusSensorUnit1Msg.cmcTcUnit1NumberOfMsgs`. Depending on the type of sensor unit, the number of entries can be between 1 and 40. The table is read out by entering the table index as the last digit of the OID. In each case the first entry begins with the index 1.

In order to read out all sensors and actors successively, the two tables `cmcTcUnit1SensorTable` and `cmcTcUnit1OutputTable` must be searched. For further action it is necessary to locate appropriate data contained in the `cmcTcStatusUnit1MsgTable`. As a rule, the entry for a sensor/actor located in

the cmcTcStatusUnit1MsgTable is found under the same index as stored in the cmcTcUnit1SensorTable or cmcTcUnit1OutputTable. However, there is an exception for the climate unit. The fan output is stored in the cmcTcUnit1OutputTable under Index 1; however, the associated entry in the cmcTcStatusUnit1MsgTable is found under Index 3. This peculiarity will be dealt with again in Part 2.

cmcTcStatus.cmcTcStatusSensorUnit1.cmcTcStatusUnit1Sensors

In cmcTcUnit1SensorTable.cmcTcUnit1SensorEntry.unit1SensorType the type of a sensor is encoded that is present under the index used in the query. If the value equals 1, then no sensor is present; otherwise the sensor type decides which SNMP variables are relevant for the respective sensor. Part 2 of this document contains a tabular summary of the variables to be used for the possible sensor types.

For analogue sensors the following four variables from this table are relevant:
cmcTcUnit1SensorTable.cmcTcUnit1SensorEntry.unit1SensorValue,
cmcTcUnit1SensorTable.cmcTcUnit1SensorEntry.unit1SensorSetHigh,
cmcTcUnit1SensorTable.cmcTcUnit1SensorEntry.unit1SensorSetLow,
cmcTcUnit1SensorTable.cmcTcUnit1SensorEntry.unit1SensorSetWarn.
The first variable gives the measured analogue value; the other three are setpoints that can be set: two different upper setpoints and one lower setpoint.

cmcTcStatus.cmcTcStatusSensorUnit1.cmcTcStatusUnit1Outputs

In this table the type of actor must also first be analysed from the variable cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputType.
The analysis of the variable is analogous to the analysis of the sensor types. If the variable has a value of 1, then no actor is present; otherwise the output type determines which SNMP variables are relevant for the respective actor. Part 2 of this document contains a tabular summary of the variables to be used for the possible output types.

Depending on actor type, the following variables from this table are relevant:
cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputValue,
cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputConfig,
cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputDelay,
cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1TimeoutAction.
The first variable must be analysed only for certain sensors; the respective meanings are described in Part 2. The second variable is relevant for door control and makes it possible to set whether the door control is to be administered by the processing unit, so that the door, for example, can be unlocked by a card reader or an external digital signal.

The third variable contains a delay time, which is relevant for switching outputs. The last variable can be used to set the status an actor is to take when communication between sensor unit and processing unit breaks down. The names of the sensors and actors are normally stored in the MsgTable described in the following. For one unit, however, the names of actors are stored here in the OutputTable, in the variable `cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputText`.
Part 2 of this document describes which unit that is. To execute switching , the OutputTable contains the variable `cmcTcUnit1OutputTable.cmcTcUnit1OutputEntry.unit1OutputSet`, which must have an integer value to do any switching.

cmcTcStatus.cmcTcStatusSensorUnit1.cmcTcStatusUnit1Msg

The variable `cmcTcUnit1MsgTable.cmcTcUnit1MsgEntry.unit1MsgText` contains a designation for the respective sensor/actor that can be specified freely by the user and can, for example, contain the position of the sensor in the enclosure.

In `cmcTcUnit1MsgTable.cmcTcUnit1MsgEntry.unit1MsgStatus` the current status of the respective sensor/actor is available. The status is encoded in an integer variable. This variable is the most important one for analysis of the statuses of the monitoring system.

Two other variables can be used to set whether a critical status of the respective sensor is to be reported to the processing unit via the alarm relay or the beeper. These two variables are designated `cmcTcUnit1MsgTable.cmcTcUnit1Entry.cmcTcUnit1MsgRelay` and `cmcTcUnit1MsgTable.cmcTcUnit1Entry.cmcTcUnit1MsgBeeper`.

There are also four variables used to set the trap receiver to which a message is sent when the status of the respective sensor/actor changes. These four variables are designated: `cmcTcUnit1MsgTable.cmcTcUnit1Entry.cmcTcUnit1MsgTrapX`, where ´X´ is replaced by a number from 1 to 4.

`cmcTcUnit1MsgTable.cmcTcUnit1Entry.cmcTcUnit1MsgQuit` can be used to set whether an alarm status is automatically reset when the triggering condition no longer exists, or whether the alarm status has to be confirmed by the user.

cmcTcStatus.cmcTcStatusExtUnit

In this area the variables `cmcTcValuesRelay` and `cmcTcValuesBeeper` are used for a central setting of how a critical status is reported at the inputs. Here too, there are four variables which can be used to specify to which trap receiver the status changes are reported. The variables are called `cmcTcValuesTrapX`, where ´X´ is replaced by a number between 1 and 4. In `cmcTcNumberOfValues` it is possible to call up how many inputs are shown via the following table. With these inputs – unlike

the sensor units – it is not necessary to analyse different tables; all available information is summarised there.

Currently, only entries for voltage measurement are possible as “sensors” in this table. The following variables are relevant. `mcTcValuesTable.cmcTcValuesEntry.valuesText` is used as the designation for the entry; the status is represented in the variable `cmcTcValuesTable.cmcTcValuesEntry.valuesStatus`. Three variables are used for the measured value and the two setpoints that can be set: `cmcTcValuesTable.cmcTcValuesEntry.valuesValue`, `cmcTcValuesTable.cmcTcValuesEntry.valuesSetHigh`, and `cmcTcValuesTable.cmcTcValuesEntry.valuesSetLow`.

cmcTcSetup and cmcTcTrapControl

In the two areas `cmcTcSetup` and `cmcTcTrapControl` basic settings are made that normally have to be changed only during start-up.

cmcTcControl

The only variable contained in this area `cmcTcQuitUnit` is used to acknowledge configuration changes and alarms that do not reset automatically. For this purpose the variable is given an integer value.

Part 2: Tables on the relevance of variables to components

Number of table entries for the various unit types

Unit Type	Number of Units SensorTable	Number of Units OutputTable	Number of Units MsgTable	Comment
I/O Unit	4	4	4	Per index there can never be more than one sensor or one actor.
Access Unit	8	6	4	There are two groups of variables, each assigned to a door group. Per group there are 3 entries in the SensorTable and the OutputTable and 2 entries in the MsgTable. The entries 7 and 8 of the SensorTable refer to reading devices, but are normally not relevant as SNMP variables.
Climate Unit	2	1	3	The first entry in the OutputTable corresponds to the third entry of the MsgTable and refers to the fan to be connected.
PSM Unit	Currently 3, 6, 9 or 12; later 15, 18 and 21 also possible	Currently 3, 6, 9 or 12; later 15, 18 and 21 also possible	Currently 3, 6, 9 or 12; later 15, 18 and 21 also possible	In each case three entries must be viewed together; the sensor types are fixed, so that the variables to be analysed are also fixed.
FCS Unit	3	1	3	The sensor types are fixed, so that the variables to be analysed are also fixed; however the first sensor can be left out.
RTT-I/O Unit	40	20	40	For each connected climate unit, 4 variables each of the SensorTable and MsgTable and 2 variables of the OutputTable are relevant.

Relevant variables for the various sensors/actors and units

Unit Type	Sensor Types	Output Types	Relevant Variables
I/O Unit			Each sensor/actor is assigned in a table index and is analysed separately.
	4, 6, 7, 8, 13, 14, 17, 19, 23		`unit1MsgText`, `unit1MsgStatus`, `unit1MsgRelay`, `unit1MsgBeeper`, `unit1MsgTrapX`, `unit1MsgQuit`
	10, 11, 12, 18		As above, plus: `unit1SensorValue`, `unit1SensorSetHigh`, `unit1SensorSetLow`, `unit1SensorSetWarn`
	5		As one line above; however, the value `unit1SensorSetLow` has a different meaning here: the value is used as “Alarm Delay”.
		9, 11	`unit1MsgText`, `unit1MsgStatus`, `unit1MsgTrapX`, `unit1OutputDelay`, `unit1TimeoutAction`, `unit1OutputSet`
Access Unit	4	4, 5, 6, 10, 12, 13	For both of the door systems controlled by the access unit there are 2 variables in the MsgTable. For the first door system the variable with Index 1 indicates the status of the door; the variable with Index 2 indicates from where the door was last unlocked. For the second door system the variables with Index 3 and 4 are used. A door system consists of at least one of the various actors and one sensor (for door system 1 within the first 3 entries; for door system 2 within entries 4 to 6 of the respective tables). When these conditions are fulfilled, the variables `unit1MsgText`, `unit1MsgStatus`, `unit1MsgRelay`, `unit1MsgBeeper`, `unit1MsgTrapX`, and `unit1MsgQuit` must be taken from the MsgTable twice for each door system. For each door system the three variables `unit1OutputDelay`, `unit1OutputConfig` and `unit1OutputSet` from the OutputTable must be used. It is necessary to use the table index under which the first actor of the door system was found.

Climate Unit	4, 6, 7, 8, 13, 14, 17		unit1MsgText, unit1MsgStatus, unit1MsgRelay, unit1MsgBeeper, unit1MsgTrapX, unit1MsgQuit
	10		As above, plus: unit1SensorValue, unit1SensorSetHigh, unit1SensorSetLow, unit1SensorSetWarn; however, the value unit1SensorSetWarn is used as setpoint for the fan.
		7	unit1MsgText, unit1MsgStatus, unit1MsgTrapX, unit1TimeoutAction The two entries of the MsgTable have Index 3; the entry of the OutputTable has Index 1.
PSM Unit	30, 31, 32	18, 19, 20	It is always possible to analyse three table entries together; the sensors and actors listed are always present in the order indicated. It suffices to process the indicated sensor types; the output types need not be analysed separately. In each of the three cases the variables unit1MsgText, unit1MsgTrapX, and unit1MsgStatus from the MsgTable are to be used; for the second index the variables unit1MsgRelay, unit1MsgBeeper and unit1MsgQuit are also relevant. The entries for the first index give the status of power measurement; the second index shows the overall status of the unit; the third index shows the mounting position. Furthermore, the unit1SensorValue with the first index is used to show the power measurement; this value is multiplied by a factor of 10 and shown as an integer. The value unit1OutputDelay with the first index and all three values unit1OutputValue must also be used. The entry unit1OutputValue with the first index shows the switching status of the unit; the second has an upper and the third a lower alarm setpoint for power measurement. The value unit1OutputSet with the first index is used to switch the device.

FCS Unit	10		Like temperature sensor (sensor type 10) for the climate unit
	21		Fixed for Index 2: `unit1MsgText`, `unit1MsgStatus`, `unit1SensorValue`
	22		Fixed for Index 3: `unit1MsgText`, `unit1MsgStatus`, `unit1SensorValue`, `unit1MsgRelay`, `unit1MsgBeeper`, `unit1MsgTrapX`, `unit1MsgQuit`
		14	No analysis needed
RTT-I/O Unit			Four entries of the MsgTable and SensorTable plus two entries of the OutputTable can always be analysed together, since the listed sensors and actors are always present.
	24		Fixed for the first index: `unit1MsgText`, `unit1MsgStatus`, `unit1SensorValue`, `unit1MsgRelay`, `unit1MsgBeeper`, `unit1MsgTrapX`, `unit1MsgQuit`
	25		Fixed for the second index: `unit1MsgText`, `unit1MsgStatus`, `unit1SensorValue`, `unit1MsgRelay`, `unit1MsgBeeper`, `unit1MsgTrapX`, `unit1MsgQuit`
	10		Fixed for the third index: like temperature sensor (sensor type 10) for the I/O unit
	26		Fixed for the fourth index: `unit1MsgText`, `unit1MsgStatus`, `unit1SensorValue`
		14	Fixed for the first index of the OutputTable: `unit1OutputText`, `unit1OutputValue`; the value `unit1OutputValue` is used to hold the setpoint for temperature regulation of the climate unit.
		15	Fixed for the first index of the OutputTable: `unit1OutputText`, `unit1OutputValue`; the value `unit1OutputValue` is used to hold the alarm setpoint for the temperature difference at the filter.

Relevant Sensor Types

Sensor Types	Description
4	Zugangssensor / Access
5	Vandalismussensor / Vandalism
6	Bewegungsmelder / Motion
7	Rauchmelder / Smoke
8	Luftstromsensor / Airflow
10	Temperatursensor / Temperature
11	Analogsensor / Analogue
12	Feuchtesensor / Humidity
13	Digitaleingang NO / Digital In NO
14	Digitaleingang NC / Digital In NC
17	Spannungswächter / Voltage Detector
18	Spannungssensor / Voltage
19	Lüfteralarm / Fan Alarm
21	Lüfterdrehzahl / Fan Speed
22	Lüfterstatus / Fan State
23	Leckagesensor / Leakage
24	Warnung RTT / Warning RTT
25	Alarm RTT / Alarm RTT
26	Filter RTT / Filter RTT
30	Strommessung PSM / Current PSM
31	Status PSM / State PSM
32	Einbaulage PSM / Position PSM

Relevant Output Types

Output Types	Description
4	Türverriegelung / Door Lock
5	Türverriegelung / Door Lock
6	Türverriegelung / Door Lock
7	Lüfter / Fan
9	Digitalausgang / Digital Out
10	Raumtür / Room Lock
11	Spannungsschalter / Power Out
12	Türverriegelung / Door Lock
13	Türverriegelung / Door Lock
14	Schwellwert / Setpoint
15	Temperatur-Alarmschwelle / Setpoint Max. Temp.
18	Schaltausgang PSM / Relay PSM
19	Obere Alarmschwelle PSM / Setpoint Max. Current
20	Untere Alarmschwelle PSM / Setpoint Min. Current

Part 3: Example for determining the relevant variables

The example deals with a CMC-TC device to which an access unit is connected as third sensor unit and a PSM unit as second sensor unit. In the access unit, a door system is active; the door handle is connected to the second port and the associated access sensor to the third port. The following table shows the contents of the SNMP variables. The fields containing relevant data are highlighted in green. The yellow items are also to be analysed to determine the relevant variables. Following the table there is a short explanation.

Access unit as third unit

`cmcTcUnit3NumberOfSensors` = 8

unit3 Sensor Index	unit3 Sensor Type	unit3 Sensor Text	unit3 Sensor Status	unit3 Sensor Value	unit3 Sensor SetHigh	unit3 Sensor SetLow	unit3 Sensor SetWarn
1	1	not available	1	0	0	0	0
2	15	Doorlock Sensor	4	1	0	0	0
3	4	Access Sensor	4	1	0	0	0
4	1	not available	1	0	0	0	0
5	1	not available	1	0	0	0	0
6	1	not available	1	0	0	0	0
7	20	Cardreader/ Keypad	5	-1	0	0	0
8	20	Cardreader/ Keypad	5	-1	0	0	0

The first three columns refer to the first door system. The first door system consists of the required two sensors. The door handle is assigned to the second index.

`cmcTcUnit2NumberOfOutputs` = 6

unit3 Output Index	unit3 Output Type	unit3 Output Text	unit3 Output Status	unit3 Output Value	unit3 Output Set	unit3 Output Config	unit3 Output Delay	unit3 Output Timeout
1	1	not available	1	0	1	1	0	1
2	4	Handle Lock	6	1	3	2	20	1
3	1	not available	1	0	1	1	0	1
4	1	not available	1	0	1	1	0	1
5	1	not available	1	0	1	1	0	1
6	1	not available	1	0	1	1	0	1

In the SensorTable it was determined that the door handle is assigned to the second index; therefore the values marked here in the second column are to be used.

cmcTcUnit2NumberOfMsgs = 4

unit3 Msg Index	unit3 MsgText	unit3 Msg Status	unit3 Msg Relay	unit3 Msg Beeper	unit3 Msg Trap1	unit3 Msg Trap2	unit3 Msg Trap3	unit3 Msg Trap4	unit3 MsgQuit
1	Door Lock 1	13	2	2	2	2	2	2	1
2	Last Access 1	17	2	2	2	2	2	2	1
3	not available	1	2	2	2	2	2	2	1
4	not available	1	2	2	2	2	2	2	1

In the SensorTable it was determined that it is a complete door system; therefore the variables marked in the second column must be used.

PSM unit as second unit

cmcTcUnit3NumberOfSensors = 3

unit3 Sensor Index	unit3 Sensor Type	unit3 Sensor Text	unit3 Sensor Status	unit3 Sensor Value	unit3 Sensor SetHigh	unit3 Sensor SetLow	unit3 Sensor SetWarn
1	30	Current	4	1	0	0	0
2	31	Status	5	4	0	0	0
3	32	Position	4	1	0	0	0

The table has three entries; thus it is a single PSM unit. If multiple PSM units were connected in series, there would be a correspondingly higher number of entries; then the variable must be used according to that multiple. Each PSM unit is handled in the tables with three entries each. The variable marked in the SensorTable contains the power measurement multiplied by a factor of 10. Thus the current power measurement in this case is 0.1A.

cmcTcUnit2NumberOfOutputs = 3

unit3 Output Index	unit3 Output Type	unit3 Output Text	unit3 Output Status	unit3 Output Value	unit3 Output Set	unit3 Output Config	unit3 Output Delay	unit3 Output Timeout
1	18	PSM On/Off	6	1	1	1	0	1
2	19	PSM Setpoint High	4	11	1	1	0	1
3	20	PSM Setpoint Low	4	0	1	1	0	1

The marked variables can simply be taken over. When multiple PSM units are present, the corresponding variables of the following table lines must also be taken. The meaning of the marked variables is explained in Part 2 of the document.

cmcTcUnit2NumberOfMsgs = 4

unit3 Msg Index	unit3 MsgText	unit3 Msg Status	unit3 Msg Relay	unit3 Msg Beeper	unit3 Msg Trap1	unit3 Msg Trap2	unit3 Msg Trap3	unit3 Msg Trap4	unit3 MsgQuit
1	Current	4	2	2	2	2	2	2	1
2	Status	5	1	2	2	2	2	2	1
3	Position	20	2	2	2	2	2	2	1

The marked variables can simply be taken over. When multiple PSM units are present, the corresponding variables of the following table lines must also be taken.