MSR Sub-project MEDOC

Structure Principles

Scope: Networks

MSR MEDOC, Dipl.-Ing. R. Reimer

Abstract

This document describes the principles for the structure of the MSR development documentation MEDOC for *Networks*. The description can also be used as a guideline for preparing structured development documentation.

Table of Contents

List of Figures

List of Tables

Introduction

Companies

MSR MEDOC [MEDOC]

Name Roles	Departement	Address	Contact
Dipl.-Inform. H. Gengenbach			
Dipl.-Ing. E. Hinderer			
Dipl. Ing. E. Jakobi			
Dipl.-Math. M. Krause			
Dipl.-Ing. R. Mertl			
Dipl.-Ing. D. Neumann			
Dipl.-Inform. P. Rauleder			
Dipl.-Ing. T. Riegraf			
Dipl.-Ing. B. Weichel			
Dipl.-Inform. J. Wieland			
Dipl.-Ing. R. Reimer			

Version Information

Document Part	Editor			
	Company	Version	State	Remarks
2002-02-07	Dipl.-Ing. R. Reimer			
For details refer to nr. 1, Page 85	MEDOC	3	RD	

1 Introduction

This document describes the principles for the structure of for the MSR development documentation MEDOC for *Networks*. The description can also be used as guideline for preparing structured development documentation.

The description is given in, amongst others, the structures of graphics that have been acquired from implementation of the *MSRNET.DTD* prototype. The meanings for the symbols used in the graphics are given in Figure 24 Meaning of symbols in the structural graphics p. 74.

It shall be noted at this point that MSR does not pursue standardization and will not pursue any in the future. Only content and structure of existing standards/in-house standards are used in the requirements documentation and in the MEDOC development documentation. MEDOC shall provide the possibility of storing the contents of (inter)national standards and in-house standards as well as non-standardized documents, for describing data of relevance to development documentation.

A simplified presentation of this is given in the following table:

Table 1: Delimitation of MEDOC activities from standardization activities

MEDOC	Standard	In-house standard	No standard
Structure	Structure of standards can be reproduced by MEDOC	Structure of in-house standards can be reproduced	Agreed by MSR group
Content	Master data taken from standards	Master data can be supplemented in specific cases	

MEDOC allows inclusion of the contents and structure of international, national and in-house standards. Fundamentals for the structure definitions are standards that are valid at the time of printing. In addition to this, data that cannot be reproduced by any standard, can be recorded using MEDOC. The structure for the data will be agreed by the MSR group, contents will not be subjected to any standardization whatsoever. Tables and contents of tables that are given in this document are only to be regarded as examples of data. The contents have been taken in some instances from international standards since different limiting values apply for individual companies within the consortium.

2 Objective and procedure

The objective of the study group *MEDOC Networks* is to define one format for all possible bus descriptions and descriptions for networks. The data modeling is performed initially on the basis of the CAN (Full CAN and Basic CAN). The transfer to further network formats is made after the data modeling has been completed.

1 General

The *MSRDOC.DTD* modeled up to now always represented one entity, a system including the associated *control unit*s or the associated *control unit* (e.g. an ABS system). Since a network generally links several *control unit*s in various systems (e.g. the *engine control unit* and the *ABS control unit*), it is meaningful to display the description of networks in a separate DTD.

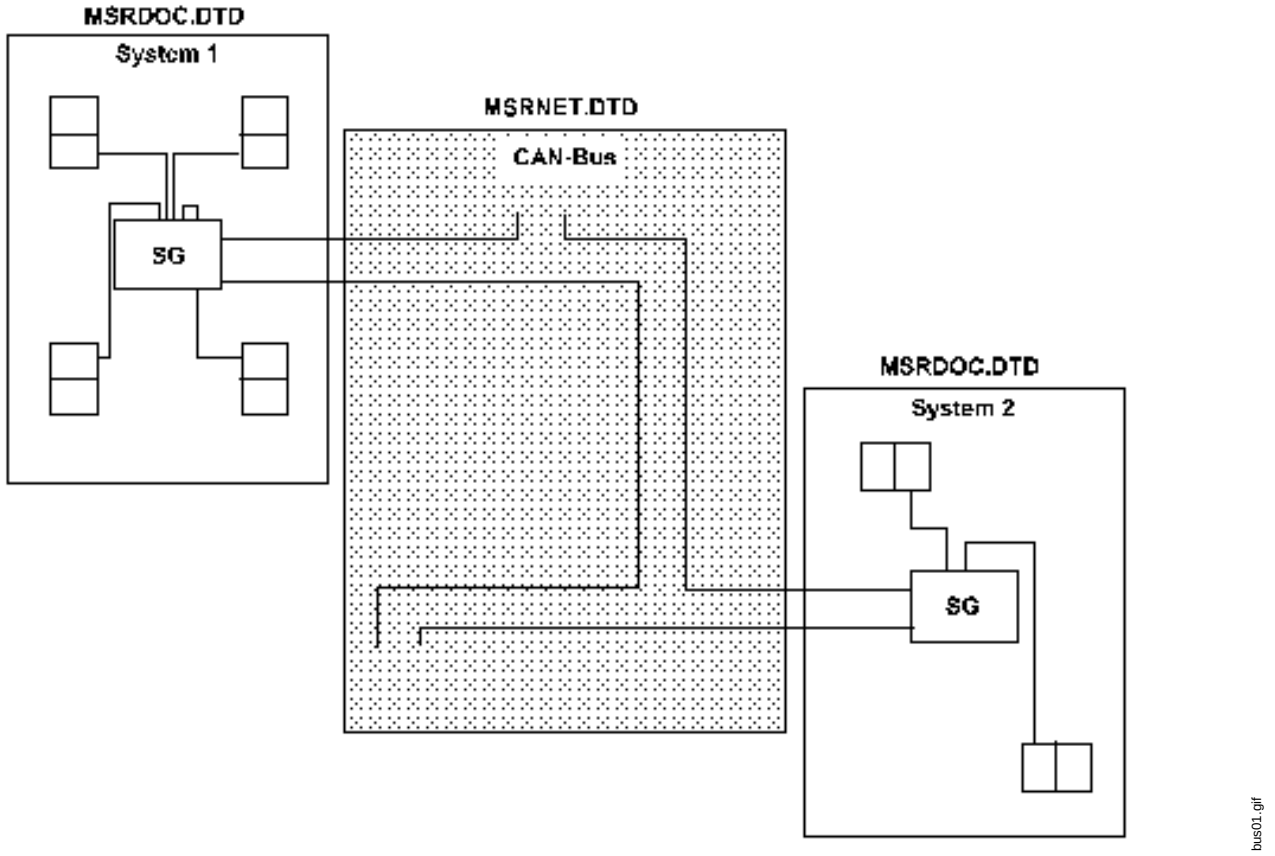


Figure 1: MSRDOC.DTD and MSRNET.DTD

Entity-overriding links are needed for a description of coupling the control units to the network. A corresponding variant concept based on the *MSRDOC.DTD* is foreseen for this, such that the references between entities of the *MSRDOC.DTD* and the *MSRNET.DTD* can be written with reference to variants. A series-related or series-overriding description of networks must be possible.

If a *vehicle system* possesses several network systems in parallel, then these shall be accordingly described in several entities of the *MSRNET.DTD*. A clear distinction (engineering as well as organization) between the differing networks is attained by this[1].

The supplier receives the description of the entire network topology in the types series in question from the manufacturer.

A link between the *MSRNET.DTD* and the *MSRDOC.DTD*'s via the short description for the net-signals is conceivable.

[1] Gateways initially propose a compilation of several networks in one entity. The gateway functionality is however realized in a specific system or unit, and is therefore to be described in this performance or software.

2 Arrangement of network and control units

2.1 Requirements

Coupling a *control unit* with one or more networks can be realized by differing ways and means. Three of these possibilities are shown in Figure 2 Linking network and control unit p. 15:

a) Single feeder and network system

b) Branched feeder or star point in the *control unit* (multiple n-conductor lines)

c) Connections to several networks (e.g. gateway)

Common to all three approaches is that the arrangement is on the basis of control-unit signals to names for network lines[2]. An arrangement to a control-unit port does not appear to have sufficient flexibility for use as a network description for several families of control units.

[2] The term net-signal is not used here as this appears elsewhere in another context in conjunction with messages.

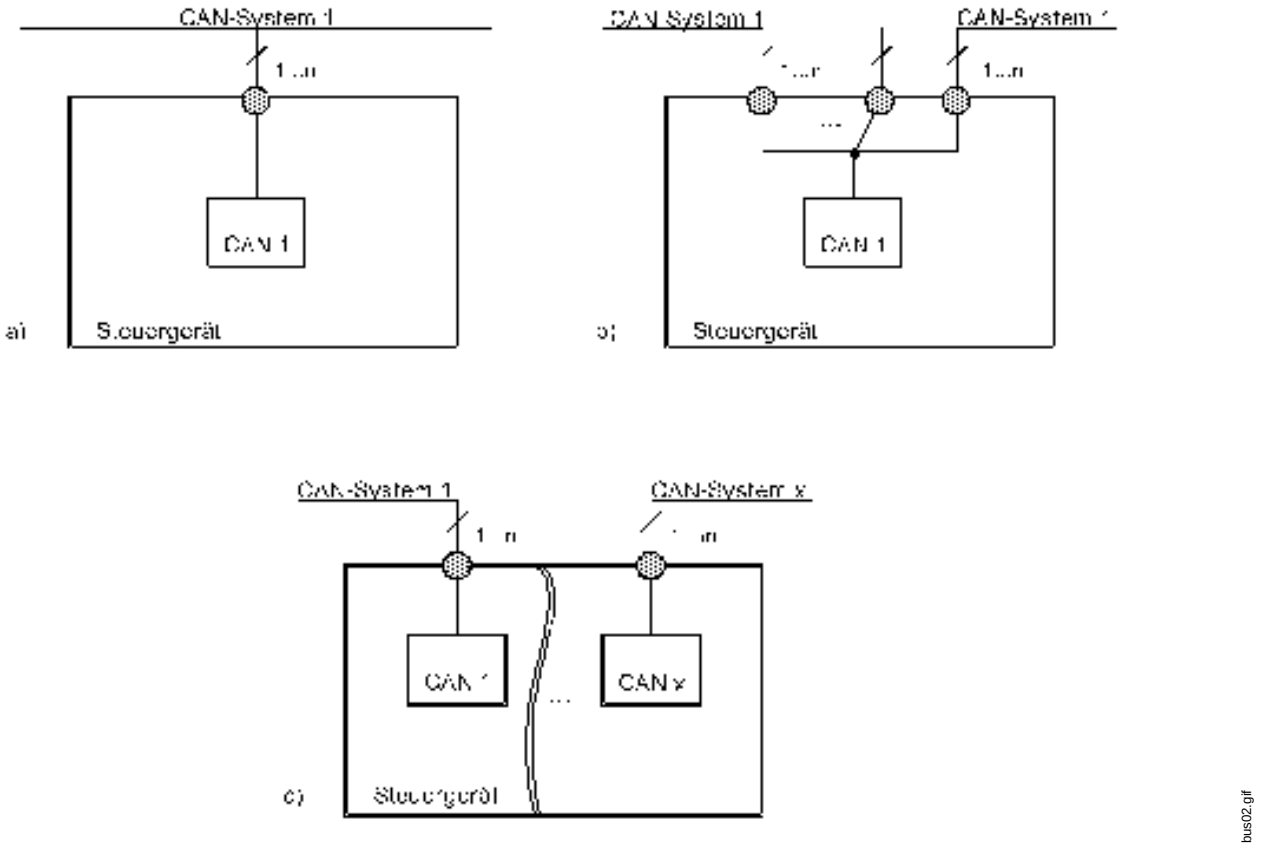


Figure 2: Linking network and control unit

Whereas in figure a) the *control unit* is linked to the network via an n-conductor single feeder (e.g. CAN_High, CAN_Low, CAN_Shield), there are a number of these n-conductor signal lines in figure b). Case b) illustrates a branched feeder or star point of the network in the *control unit*.

A *control unit* can furthermore, as illustrated in figure c), be connected to differing networks (e.g. gateway). Versions of the interfaces to each of the networks to a) and b) are possible here.

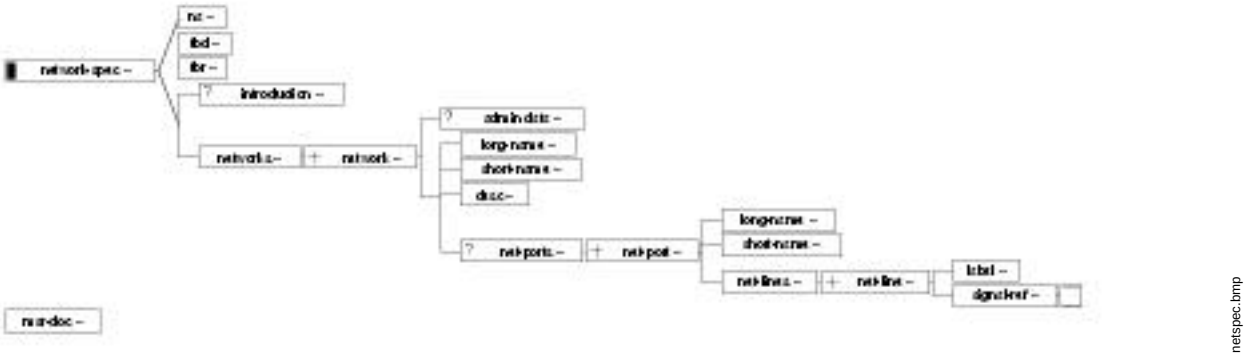
The alternatives in Topic App. D Alternatives considered for linking control unit and network p. 75, were also considered alongside the solution chosen for the description of *control unit* and network. The solution in Topic 2.2 Description of network connections in MSRDOC p. 15) is applied.

2.2 Description of network connections in MSRDOC

A new element **<network-spec>** (refer to Figure 3 Network connections in MSRDOC p. 16) is introduced in the **<architecture>** in *MSRDOC.DTD* in order to be able to define the network links for the component types. This modeling was chosen for the following reasons:

- The objectives for a (semantic execution) referencing from *MSRNET.DTD* entities are concentrated in **<network-spec>**.
- Network-related characteristics and interrelationships of signals can be concentrated here without a general expansion of the contents model of **< signal>** and at the same time, so as to be able to handle net-signals like all other signals.

A possible sematic overlapping with **<signal-class>** is intentionally accepted since the latter has been designed for company and process-specific classifications.



netspec.bmp

Figure 3: Network connections in MSRDOC

Example for an entry in the MSRDOC entity

A *MSRDOC.DTD* entity (for control unit 2 in Figure 4 Examplary scenario p. 17) looks in principle like the following (the counterpart in the *MSRNET.DTD* entity is given in Topic 3.2.2.4 Example of a description for the network topology p. 23):

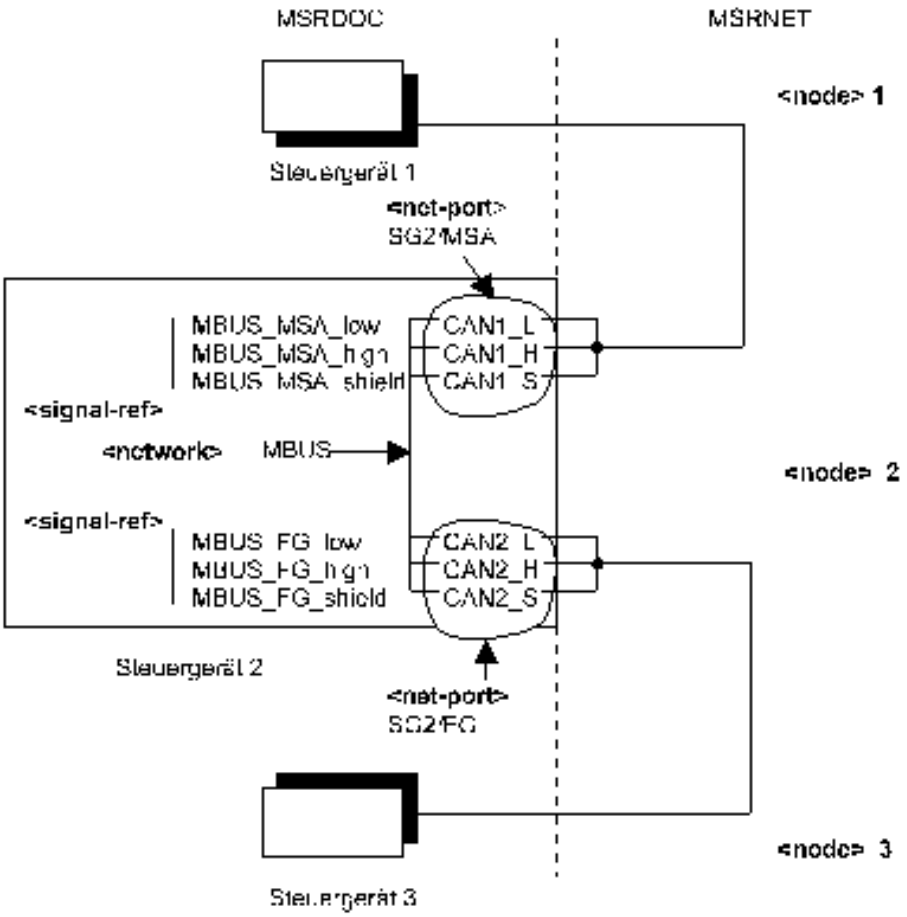


Figure 4: Examplary scenario

```
<part-type>
   <long-name>control unit 2</long-name>
   <short-name>SG2</short-name>
...

 <architecture>
   <scheme-diagrams> ...
   <interface-spec>  ...
   <signal-spec>     ...
   <network-spec>
    <networks>
      <network><long-name>CAN bus for engine management</>
        <short-name>MBUS</>
        <desc>This bus primarily handles engine management</>
        <net-ports>
           <net-port>
              <long-name>engine-side connection</>
              <short-name>MSA</>
              <net-lines>
                <net-line>
                   <net-line-name>CAN_LOW</>
                   <signal-ref sref="MBUS_MSA_LOW">
                </>
                <net-line>
                   <net-line-name>CAN_High</>
                   <signal-ref sref="MBUS_MSA_HIGH">
                </>
```

```
                <net-line>
                   <net-line-name>CAN_shield</>
                   <signal-ref sref="MBUS_MSA_SHIELD">
                </>
             </net-port>
             <net-port>
                <long-name>connection on passenger-compartment side</>
                <short-name>FG</>
                <net-lines>
                  <net-line>
                     <net-line-name>CAN_LOW</>
                     <signal-ref sref="MBUS_FG_LOW">
                  </>
                  <net-line>
                     <net-line-name>CAN_HIGH</>
                     <signal-ref sref="MBUS_FG_HIGH">
                  </>
                  <net-line>
                     <net-line-name>CAN_SHIELD</>
                     <signal-ref sref="MBUS_FG_SHIELD">
                  </>
              </net-port>
          </net-ports>
      </network>
     </networks>
   </network-spec>
  ...
</part-type>
```

Table 2: Summary of net-port, net-line, and signal name

Net-Port	Net-Line	Signal
MSA	CAN_LOW	MBUS_MSA_LOW
	CAN_HIGH	MBUS_MSA_HIGH
	CAN_SHIELD	MBUS_MSA_SHIELD
FG	CAN_LOW	MBUS_FG_LOW
	CAN_HIGH	MBUS_FG_HIGH
	CAN_SHIELD	MBUS_FG_SHIELD

This variant completely models the component-specific network characteristics. Included in this is also the explicit arrangement of component-signal name and network line[3].

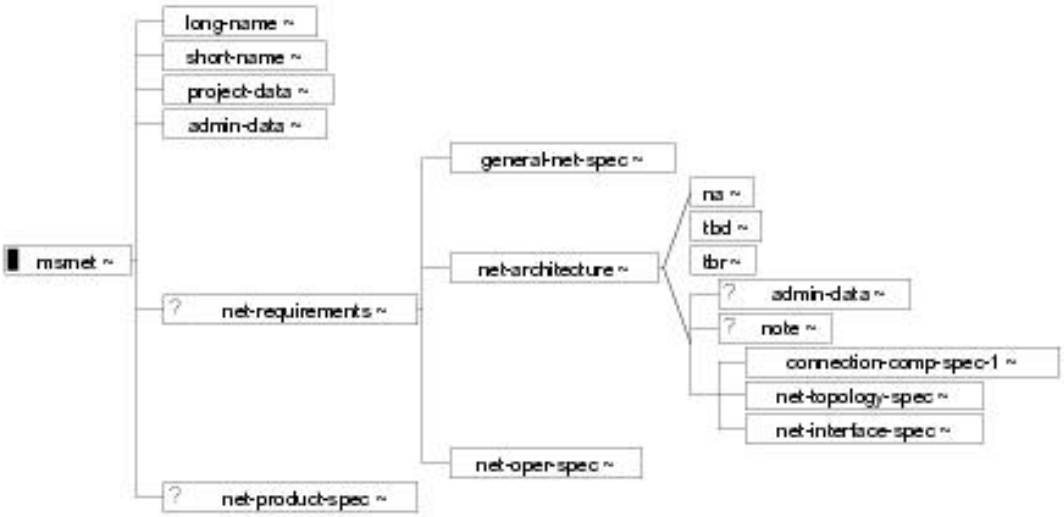
Net-ports are introduced in order to be able to cover Figure 2 Linking network and control unit p. 15 case b). The network is applied to a component at several points in this case (i.e. at several ports) and is linked within the component. It can thereby be assumed that the network lines are assigned different designations at the differing pins. A specific net-port arrangement of network-line names to component-internal signal names must therefore be made. This arrangement can in principle be realized by two variants:

- Arrangement of network-line and component signal name per net-port
- Arrangement of net-port und component signal name per network-line[4]
-

[3] The network-line identifiers are not explicitely normed. It is the task of a semantic check to ensure consistency by the network-line identifiers for the differing net-ports.

[4] This solution was not chosen and an example is therefore not given.

3 Structuring of MSRNET.DTD



dtdbasic.bmp

Figure 5: Basic structure of MSRNET.DTD

The description of a CAN network is made up of the following parts:

- Long and short designation (**<long-name> <short-name>**)
- Administration information **< admin-data>**
- Project information **<project-data >**
- General global description **<general-net-spec>** Topic 3.1 General description p. 20
- Network architecture (**<net-architecture>**) (topology, cables, network parameters, driver concept, network EMC design)
- Network operation (**<net-oper-spec>**) (data traffic, signals, messages).

An approximate pictorialization in the form of a layer model (similar to the ISO/OSI reference model for communication) is provided by the following illustration:

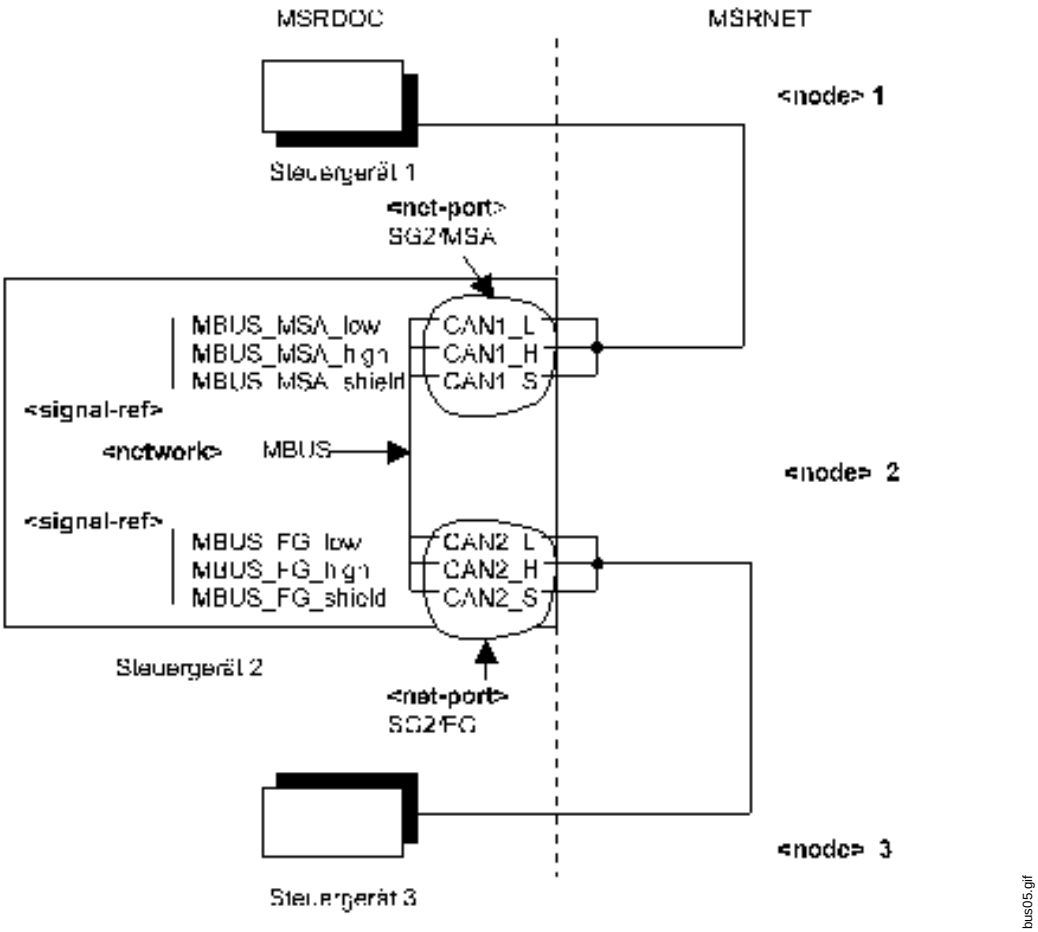


Figure 6: Layer model for times/transmission times

The first version of the DTD contains no formal description of required transmission times[5].

The parameters *Latency* and *Cycle time* are only foreseen for the initial description of messages, i.e. parameters in the lower layer within the illustration.

3.1 General description

Generally speaking, global descriptions of a network can be stored in **<general-net-spec>** . Described in this are for example, the application for the network (drive train, comfort area ...).

[5] Such a description of a formal nature is not be found in currently applicable specifications either.

3.2 Network architecture

The description of the network architecture (**< net-architecture>**) comprises the topics of ”Connection component” **<connection-comp-spec-1>**, ”Topology” (**< net-topology-spec>**) and ”Network interface” **<net-interface-spec >**. Both fixed as well as independent parameters can be described here for the network and the network hardware.

<add-spec> serves to define additional specifications nor foreseen in the DTD.

3.2.1 Connection components

The network cables used for the connection components are described as **< connection-comp-spec-1>** (refer to Figure 7 Structure for connection components p. 21), similar as in *MSRDOC.DTD* .[6].

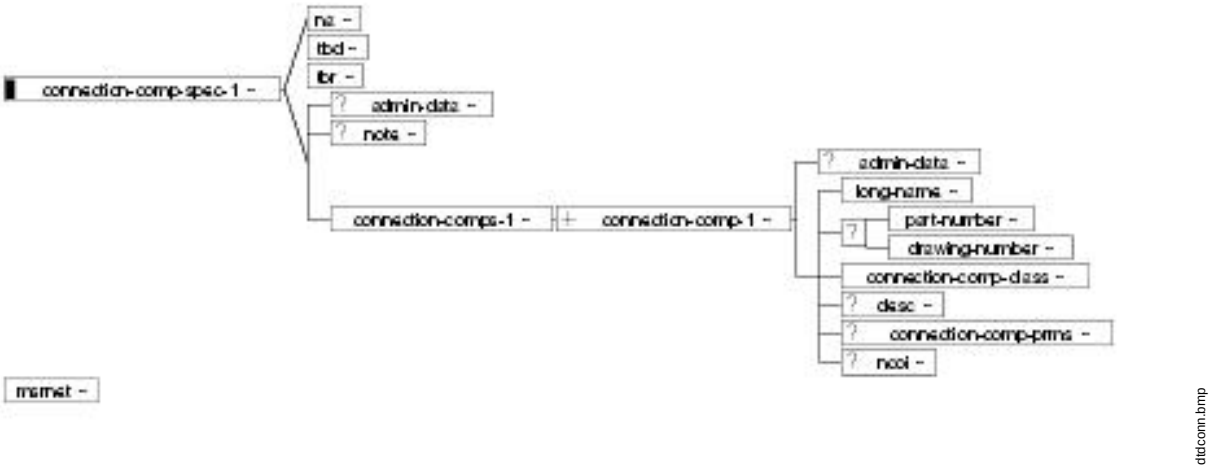


Figure 7: Structure for connection components

3.2.2 Topology

The type, the nodes as well as the segmentation, can be documented within the topology (**<net-toplogy-spec >** refer to Figure 8 Structure of the topology p. 22). The type (**<topology-type>**) can exhibit characteristics such as ”ring”, ”star”, ”bus”, ”bus with single feeder” or even ”mixed form” and must be supplemented by graphics.

Nodes **<net-nodes>** (Topic 3.2.2.1 Network lines p. 22) and segmentation **<segmentation>** (Topic 3.2.2.3 Description of segmentation p. 23) can be described in addition to this.

[6] A modification is used that does not permit use of a part-type reference.

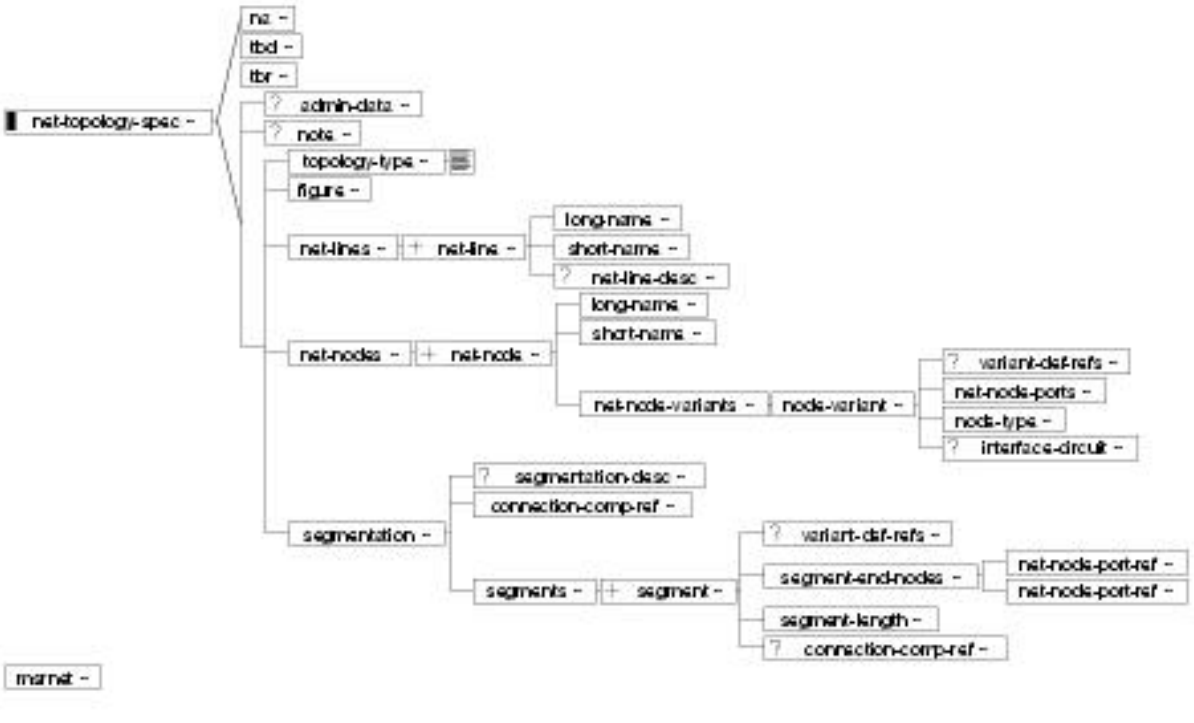


Figure 8: Structure of the topology

3.2.2.1 Network lines

The wiring in the network itself consists as a rule of many lines (called net-line here. An example for this is (*CAN_LOW*, *CAN_HIGH*, *CAN_SHIELD*). These signals can be described in topology (**<net-lines>**) as well. The description comprises as a minimum, a **<short-name >** and a **<long-name>**. A detailed presentation (**<netline-desc>**) can be given with tables, graphics, etc., as an option.

In the *MSRDOC.DTD* entities, these net-lines are arranged according to the component internal signals (compare Topic 2.2 Description of network connections in MSRDOC p. 15, **< net-line>**). Thus it is possible by this to check the consistency[7] (semantics).

An explicit allocation of these leads to the conductors (e.g. by color code) of the connection cable is not made as this cannot be described by a formalized hierarchy.

3.2.2.2 Description of network nodes

The description of the node comprises:

< short-name> Number or short designation for the node in the network

[7] Formal referencing of net-lines between msrdoc and msrnet was dispensed with for pragmatic reasons since the same designations are used as a rule in all entities.

< long-name> Designation for the node in the network

<node-type> The node type serves to differentiate between *participators* and *auxiliary nodes* . Auxiliary nodes do not participate in the network traffic but rather only serve the physical structure or presentation of the topology.

< net-node-port>s The ports for the node are defined here (compare Topic 2.2 Description of network connections in MSRDOC p. 15). This is the only point where the referential consistency to *MSRDOC.DTD* entities must be assured. The **<net-node-port>** references the component (**< part-ref>**) as well as the net-port (**<net-port-ref>**) associated with the component type from the *MSRDOC.DTD*. [8]

<interface-circuit > Information on the terminal protective circuitry. **<interface-circuit >** can be used here. This information is redundant to *MSRDOC.DTD*. The structure is retained in order to be able to give an autonomous description for a MSRNET entity. The entire terminal protective circuit can be given for a node in addition to this, whereas for *MSRDOC.DTD* only the circuit for a single pin is described. The information is optional.

3.2.2.3 Description of segmentation

Documentation of the segmentation can be both of a general nature (by text and graphics) (**<segmentation-desc>**) as well as by a formal (**<segments>**) specification.

The type of cable used as standard for the network can be given in **<connection-comp-ref>** .

The formal specification describes all segments (**<segment>**):

- Segments can be given as being variant-specific (specified in **<variant-def-refs>**).
- Details on the start-nodes, end-nodes, segment lengths (**<segment-length>**) as well as cable types for the individual segments. The start and end nodes are defined by referencing one of the **< net-node-port>**s of the node in question (**< net-node-port-ref>**).
- The information in the SU unit of measure of meters is foreseen for the segment length[9].
- A segment-specific type of cable can be given (**<connection-comp-ref>** if this deviates from the type of cable used as standard for the network.

[8] All node-ports of a node must reference the same part. The reference to the part has not be "taken out" of the node-ports in spite of this. This would have meant a distrubition for the definition of the the semantics for the reference. This should be avoided for handling reasons. It is the task of a semantic check to determine any inconsistencies.

[9] The application of the standard parameters model was dispensed with here in order to enable the presentation of segmentation as a closed table.

3.2.2.4 Example of a description for the network topology

The following example[10] corresponds to the example given in Figure 4 Examplary scenario p. 17:

```
<net-topology-spec>
  <net-topolgy-type>bus</>
  <connection-comp-ref connection-comp="nyfaz"></>
  <nodes>
    <node>
       <long-name>control unit 1</>
       <short-name>1</>
       <node-variants>
         <node-variant>
           <node-ports>
             <node-port id="sg1.can">
               <long-name>SG1 - CAN</>
               <short-name>can</>
               <net-port-ref netport="SG1/CAN">
             </node-port>
         </node-variant>
       </node-variants>
       <node-type>participator</>
    </node>
    <node>
       <long-name>control unit 2</>
       <short-name>SG2</>
       <node-variants>
         <node-variant>
           <node-ports>
             <node-port id=sg2.msa>
               <long-name>engine compartment CAN</>
               <short-name>MSA</>
               <net-port-ref netport="SG2/MSA">
             </node-port>
             <node-port id="sg2.fg">
               <long-name> passenger compartment CAN</>
               <short-name>FG</>
               <net-port-ref netport="SG2/FG">
             </node-port>
         </node-variant>
       </node-variants>
       <node-type>participator</>
    </node>
    <node>
       <long-name>control unit 3</>
       <short-name>3</>
       <node-variants>
         <node-variant>
           <node-ports>
             <node-port id=sg3.can>
               <long-name>control unit3</>
               <short-name>can</>
               <net-port-ref netport="SG3/can</>
             </node-port>
             <node-type>participator</>
         </node-variant>
       </node-variants>
     </node>
   </nodes>

   <segmentation>
     <segments>
        &lt;segment&gt;
           <segment-end-nodes>
             <node-port-ref node-port="sg1.can"></>
             <node-prot-ref node-port="sg2.msa"></>
           </segment-end-nodes>
```

[10] The example has to be updated.

```
            <segment-length>1</>
         &lt;/segment&gt;
         &lt;segment&gt;
            <segment-end-nodes>
              <node-port-ref node-port=”sg2.fg”></>
              <node-prot-ref node-port=”sg3.can”></>
            </segment-end-nodes>
            <segment-length>1</>
         &lt;/segment&gt;
       </segments>
</net-topology-spec>
```

3.2.3 Description of network interfaces

The network interface (**<net-interface-spec>**) constitutes the second part of the hardware description. This describes the driver concept (**<driver-concept>**), the network parameters globally defined for all participants (**< net-interface-prms>**), notes on the EMC design for the network (**< net-emc-design>**) as well as supplementary information (**<add-info >**).

The description of the bus interface, or bus link, consists of a bus-global physical data sheet (requirements); either according to *[External Document: ISO Norm CAN High-Speed / URL: / Relevant Position:] [External Document: ISO Norm CAN Low-Speed / URL: / Relevant Position:]* or individually characterized. It shall then be established on the participant side (control unit side) that chip that shall be used (**<part-type-spec >** in *MSRDOC*).

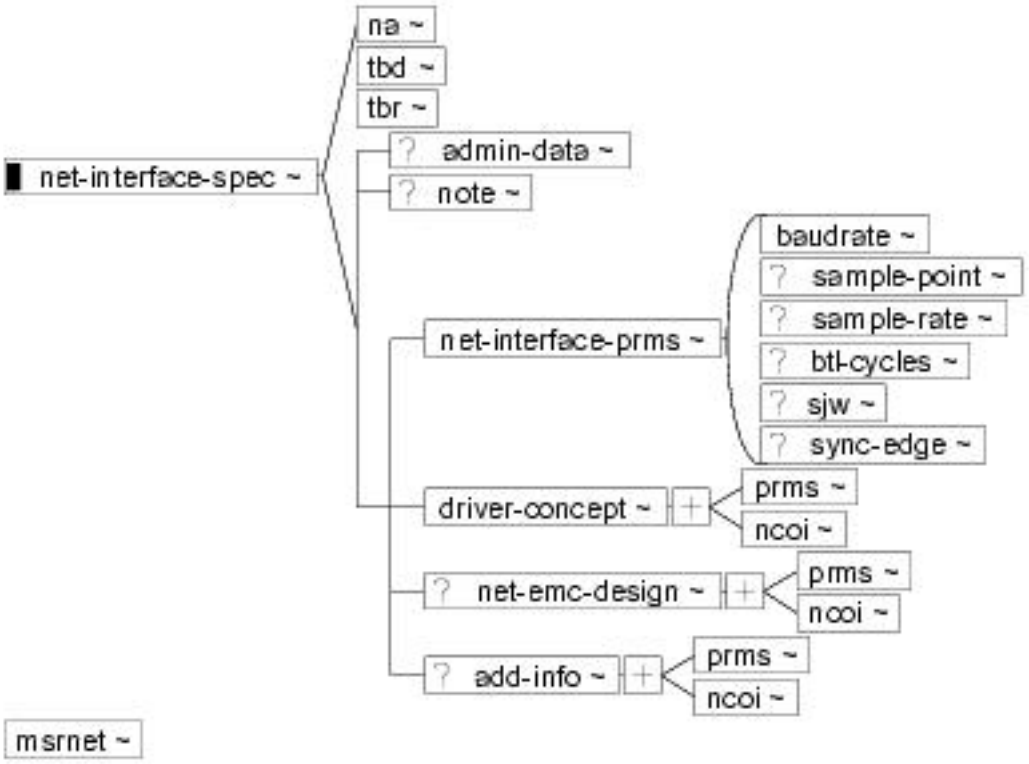


Figure 9: Structure of the network interface

The network parameters defined globally for all participants are summarized as a parameters table (**<net-interface-prms >**)[11]:

<baudrate> The baud rate (Definition Baud rate p. 70) in Hz. The **<abs>**-**<tol>** model must be used. The tolerance shall be given in %.

The baud-rate programming for the *CAN-controller* can be derived from the above parameter, is not however unique (e.g. *Register occupation*). It concerns here a description of the bus link at a higher level (higher than the physical level). This can also be termed parameterization of the CAN controller and is to be documented for the CAN controller or for the *control unit*.

[11] A specific parameter model should really be used for each of these parameters. The MSR standard parameters model has however been used in the interests of generalization. If this is executed in future as the architectural form, then more specific model models could be used. The user must however first select the correct model, or the author environment should propose the correct model.

<sample-point> The sample point for a single bit in a message. The information is given as a percentage. (**<abs>**-**<tol>** model without completed tolerance information)

This information is optional.

This parameter indicates the point in time when a bit will be sampled. It is given as a percentage of the bit time (transmission time for one bit). A typical value is 75 %.

<sample-rate> Numple of sampling operations per bit

This information is optional.

<btl-cycles> Number of Definition BTL cycles p. 70 BTL cycles as a whole-number value, i.e. the resolution of one bit (**< abs>**-**<tol>** model without completed tolerance information).

This information is optional.

<sjw> SJW (Synchronization Jump Width) as whole-number value (Definition SJW p. 72). The parameter ”SJW” (Syncro Jump Width) is given as a precentage of the bit time as well.

This information is optional.

<sync-edge> Synchronization flank, to be given as a **<text>** parameter

This information is optional.

The **<baudrate>** is global to the bus. The parameters **<sample-point>**, **<sjw >** and **<btl-cycles>** are generally global, can however be participant-specific in exceptional cases. These special circumstances are not covered in *MSRNET.DTD*.

The driver modules used or to be used can be described within the scope of the optional driver concept (**< driver-concept>**).

Just as for the driver concept, an arbitrary structure (**<net-emc-design >**) is available for describing the network EMC design.

Additional physical parameters such as flank steepness, input resistance and signal level shall be documented as required for the additional information (**< add-info>**).

3.3 Network operation

This section (**<net-oper-spec >**) describes the data traffic for the differing network services (Topic 3.3.1 General network management p. 28 as well as the behavior when faults occur (Topic 3.3.3 Error handling p. 29) and similar. The structures for the block-wise transmission of data are illustrated in Topic 3.3.5 Block transmission modes p. 29.

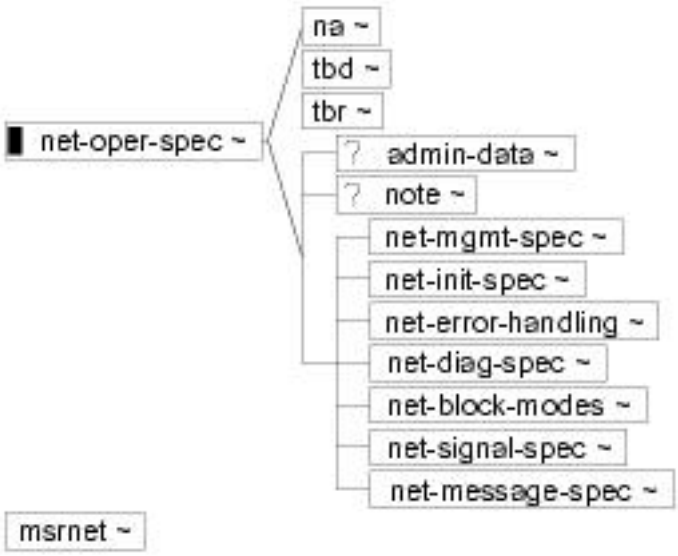


Figure 10: Structuring the network operation

A message that can be transmitted via CAN contains several signals as a rule, whereby it is possible that one one signal is being used in more than one message. It is for this reason that the signals (**<net-signal-spec>**, refer to Topic 3.3.6 Network signals p. 29) are specified first, and then the messages (**<net-message-spec>** refer to Topic 3.3.7 Messages p. 31).

3.3.1 General network management

Information can be provided in this section (**<net-mgmt-spec >**) on, amomgst others, sleep/wake-up mechanisms. If the network can run in different operational modes, then these (and the transition mechanisms) are to be described here as well.

All net-signals and messages used for network management must be specified as net-signals (**<net-signal>** or **<net-message>**) (compare Topic 3.3.6 Network signals p. 29). Grouping single messages into groups (compare Topic 3.3.7 Messages p. 31) serves this purpose in particular . Network management signals should be summarized as a separate table (CALS table) when preparing the documentation[12].

The provsion of standby values is not included in the network description (e.g. for *net-signal group*) because this is to be executed as a rule by the recipient of the signal.

There are also activities for *ASAP* and for *OSEK* regarding the standardization of network management. Formal modeling is initially dispensed with for this reason, and is only foreseen for the structure proposed for the chapter.

[12] An extension of the DTD at a later point in time could provide the structural basis for automation of this.

3.3.2 Initialization

The initialization (**<net-init-spec >**) describes services and protocols that serve to bring a network participant to a state capable of communication following power-up or after a reset.

3.3.3 Error handling

The section **<net-error-handling >** documents the mechanisms for handling errors.

3.3.4 Diagnostics

The section **<net-diag-spec>** documents the mechanisms for processing diagnostic routines in the network.

3.3.5 Block transmission modes

The block transmission modes (**<net-block-modes >**) describe application-specific protocols (initialization, diagnostics and similar) for the transfer of data packages.

It is recommended to reference the messages used for the block transmission modes. Signals are also used to set up these messages. These signals are identified as such by the net-signal class (**< net-signal-class>**).

There are also activities for *ISO TC22 SC3 WG1* regarding the standardization of block transmission modes. Formal modeling is initially dispensed with for this reason, and is only foreseen for the structure proposed for the chapter.

3.3.6 Network signals

<net-signal-spec > [13] (refer to Figure 11 Network signals p. 30) specifies the signals transmitted in the network.

Network signals are related to the variables in control-unit software. It is for this reason that optional parts of the software data dictionary (**< sw-units>** and **<sw-compu-methods>**) are included in the DTD.

The net-signal groups **<net-signal-group >**s serve to group those signals having exactly the same characteristics. A redundant description is no longer necessary due to the grouping into a group[14] [15].

[13] The term network signal is used here for differentiation in MSRDOC from the term signal.

[14] The structure is similar to that for a signal group in MSRDOC.DTD

[15] This is a general principle for MSR

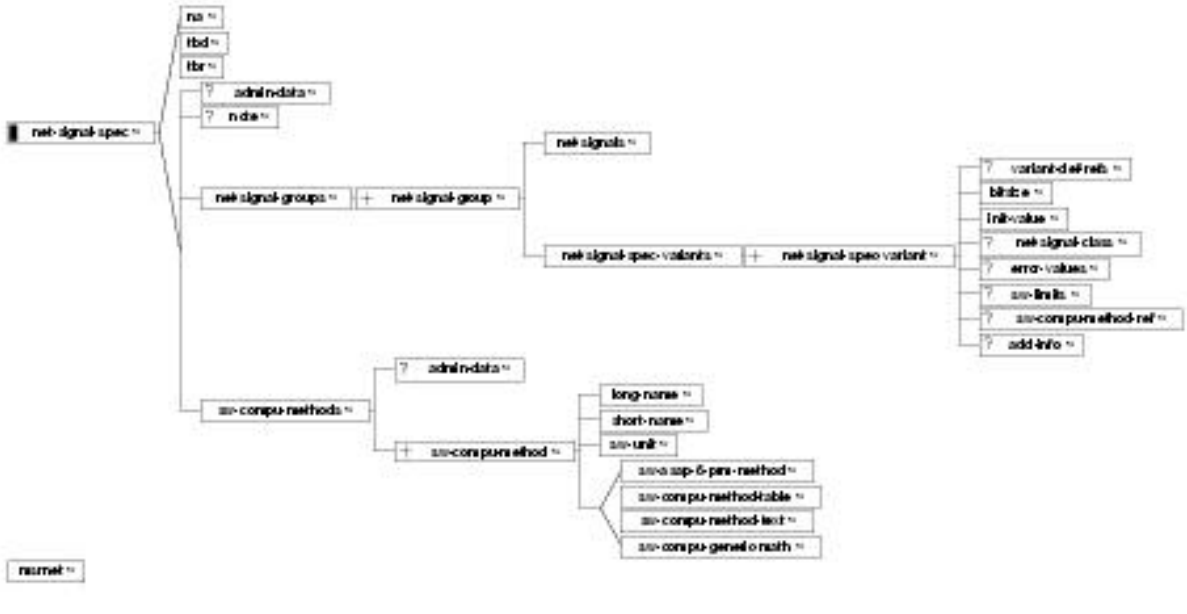


Figure 11: Network signals

The signal characteristics (**<net-signal-spec-variant >**) comprise the following parts[16]:

<bitsize> Length of the signal in bits

<init-value> Optional initialization value - this value is transmitted if the associated physical signal has not yet been measured.

<net-signal-class> The optional net-signal class (**<net-signal-class>**) can be an " *Application signal*" or " *Network management signal*".

<error-values > This value is transmitted if the associated physical signal is not available because of an error situation (e.g. sensor fallen off). Several **<error-value>**s can be given in order to differentiate between differing error situations. This information is optional.

<sw-limits> Optional information on the range of values for the signals. The information can be provided both as a network-internal (**<coded>**) as well as a physical (**<phys>**) presentation.

<sw-compu-method-ref > The conversion documents how the network-internal presentation (coded or internal values) is converted into the corresponding physical values (external values). This is documented by a reference to the **<sw-compu-methods >**. Refer also to: *[External Document: Fundamental structures for*

[16] Some of this information is optional, in order to be able for instance, to apply name and size to these block mode signals as well.

software / State: 1.1.0 / Date: 16.12.97 / Publisher: MSR AG-MEDOC / URL: / Relevant Position:]

This information is optional.

A recipients list is not foreseen for the definition of network signals since such a definition can be generated by the information on the messages (**<receivers>** within **<net-message-signals >** within **<net-message-spec>**).

3.3.7 Messages

The signals in the network are transmitted as groups of messages. These messages can be specified in **<net-message-spec>** (refer to Figure 12 Structure of messages p. 31).

Network messages can be grouped in **<net-message-set>**s. The grouping is **not** intended for messages that are structurally identical but rather serves for documentation of the grouping.

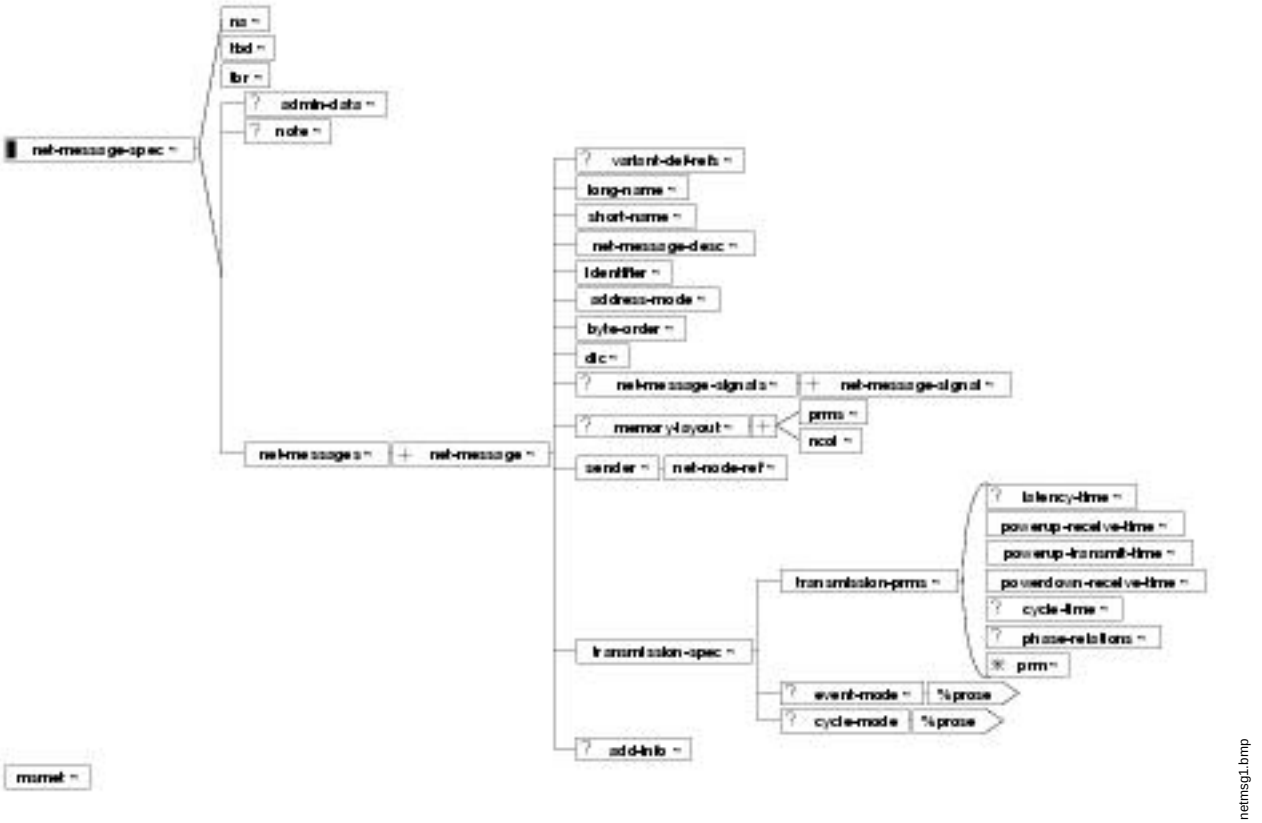


Figure 12: Structure of messages

Each message **<net-message >** is specified as a function of the variant and identified by **< long-name>**, **<short-name>**. The description consists of:

<net-message-desc> The option is given here of describing the message in prose.

<identifier> This a hex. value for identification of the message in the network.

Two mechanisms are available for determining the identifiers of a specific **<net-message >** (for a network node):

- Discrete information for the identifiers **<net-message-identifiers >**.
- A computation of the identifier (**<calc-net-message-identifiers >**) as the sum of a basis and **<Identifier-base-address >** and an offset given in nodes. The **< msg-identifier-offset>** is given by means of the respective **<sender >**.

  This serves to describe structurally identical (abstract) messages. Such messages can only then be transmitted on the bus when they have received a unique allocation to an identifer.[17]

There exists a 1-to-n relationship between **<identifier >** and **<sender>**, in order to be able to describe variants (**<variant-def-ref>**).

<address-mode> The address mode can be ”*STD*” (Standard) or ”*XTD*” (Extended).

A more appropriate designation is ”CAN data transmission format”. This is a very global definition for a bus and can be ” *Standard-Format*” or ”*Extended-Format*”. Specifications from the Bosch company are available for this. It can be defined in the case of ”*Extended-Format* ” for each *control unit*/CAN controller, whether this is ”*active*” or ”*passive* ”, i.e. whether communication with the extended format is possible or not. [18]

<byte-order> Designates the byte order in the message. Possible values are: *motorola*, *motorola forwards*, *motorola backwards* or *intel*.

<dlc> The information DLC (Data Length Code) (refer to Definition DLC p. 71) is given in bytes and indicates the length of the message. A message can include gaps and hence the length cannot be determined from the sum of the signal lengths.

<net-message-signals> The signals of a message are listed in **<net-message-signals >** as (**<net-message-signal>**). This information is optional. **<net-message-signal>** can either be direct **<net-signal>**s or an **<offset>** can be applied.

Multiplexed signals can also be given. These are specified as **< multiplex-signal-set>** that specify a **< multiplexer>** with included an **<offset>** [19] and length **<bitsize>**. A signal list (**<net-**

[17] Such messages are e.g., command messages by which a certain function within a control unit can be triggered from different nodes. Further examples are network management and diagnostic messages.

[18] It still has to be clarified where this is defined.

message-signals>) is in turn assigned in the **<multiplex-entry >** in the **<multiplex-signal-list>** to each value of the multiplexer. Any number of interleaved multiplex signals can be established in this way[20].

The associated receivers **<receivers>** shall also be documented for each signal of a message (direct or multiplexed since these can also depend on the message.

<net-message-layout> The information **<net-message-layout >** is foreseen as an optional, redundant supplement to the signal list (**< net-message-signals>**). Pre-processed information can be stored here that cannot be derived by the *SGML formatter* from the **<net-message-signals>**. Valid is always the information given in **<net-message-signals>**.

<sender> Designates the node that the signal sends.

<transmission-spec> Describes the transmission procedure for the message. The following parameters apply

<powerup-receive-time> Definition Readiness to receive following power-up p. 71
Readiness to receive following power-up

Transmission parameter for a message that describes the point in time following *power-up* as of which the message can be received.

<powerup-transmit-time> Definition Readiness to transmit following power-up p. 72
Readiness to receive following power-down

Transmission parameter for a message that describes the point in time following *power-up* as of which a message can be sent.

<powerdown-receive-time> Definition Readiness to receive following power-down p. 71
Readiness to receive following power-down

Transmission parameter for a message that describes the time

[19] The offsets refer principally to the beginning of the message and are summated (i.e. beginning at 0)

[20] This hierarchy is possible as a variant-dependent structure at all levels. Variant-dependencies should however only be introduced at the highest level.

after a *power-down message* during which a reaction to incoming messages is possible even though sending is not possible. It can be prevented by this that a component goes into the sleep mode just when a message is en route to the component.

<latency-time>

Definition Latency time p. 71
Latency time

Optional transmission parameter for a message that describes the transmission time in the network. The latency time is the time by which the sending process can be delayed by messages having a higher priority. In technical terms, the latency time is period of time between setting the TransmitRequestBit and receiving the AckMessage. The maximum latency time for this message can thus be specified for this message[21].

<cycle-time>

Definition Cycle time p. 72Cycle time (also cycle tolerance time), optional transmission time for a message that describes the tolerance time for a message for a cyclic control system (cycle sending of messages)[22].

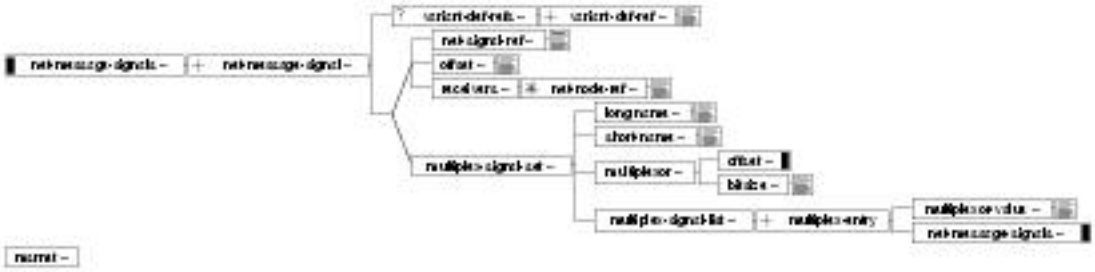
<phase-relations>

Definition Phase relation to other messages p. 71Phase relations to other messages

Transmission parameter for a message that describes whether there is a phase relation with other messages present.

[21] Only of relevance for cyclic control systems, hence optional.

[22] Only of relevance for cyclic control systems, hence optional.



netmsg2.bmp

Figure 13: Structure of signals in a message

The consistency of the entered data shall be assured by a semantic check.

3.4 Variants

Variant dependencies exist for:

- Nodes (**< net-node>**)
- Network interfaces**< interface-circuit>**
- Network segments (**< segment>**)
- Net-signals **< net-signal>**
- Net-signal characteristics **<net-signal-group>**
- Message (**< net-message>**) is then used when the message is present as variant-dependent.
- Signal entry in message (**<net-message-signal>**) is then used when the message is present as variant-dependent.
- Memory layout (task for AG-DTD %info variant-dependent could be realized with **<xref>** by variants definition).
- Transmission parameter (**<transmission-spec>**) [23]
- Two mechanisms are available for determining the identifiers of a specific **< net-message>**:
  - Discrete information pertaining to identifiers **<net-message-identifiers >**.
  - A calculation of the identifier (**<calc-net-message-identifiers >**) as the sum of a basis and **<Identifier-base-adress >** and and offset that is given in the node. The **< msg-identifier-offset>** is given using the corresponding **<sender >**. This procedure is preferentially used for network management and diagnostics.

  TBD

[23] Not yet realized in 11.0.

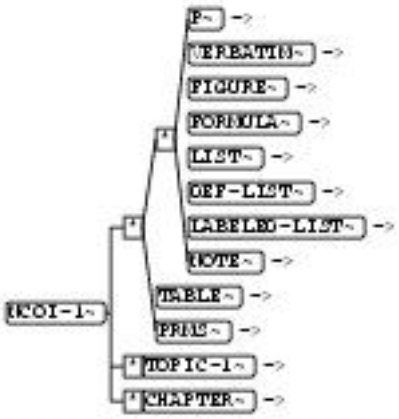
4 Basic Structures of the MSR Application Profile

All MSR DTDs are using some common data structures. These operating models are described in this chapter.

4.1 Not Content Orientated Information (ncoi)

<ncoi-1> contains all basic descriptive elements. There are also elements like **<chapter>** or **<fail-save-concept>** in the *MSRSYS DTD* which have the same content model as **<ncoi-1>**.

The figure below illustrates the structure of **<ncoi-1>**.



ncoi-1.bmp

Figure 14: Structure of <ncoi-1>

There also are two weaker ncoi models (*ncoi-2* and *ncoi-3*) with lesser elements than **<ncoi-1>**. *ncoi-2* has no **<chapters>**. *ncoi-3* has also no chapters and futhermore another ”topic” model without **<prms>**.

The components of ncoi [24] are interchangeable between all MSR DTDs[25] without any changes.

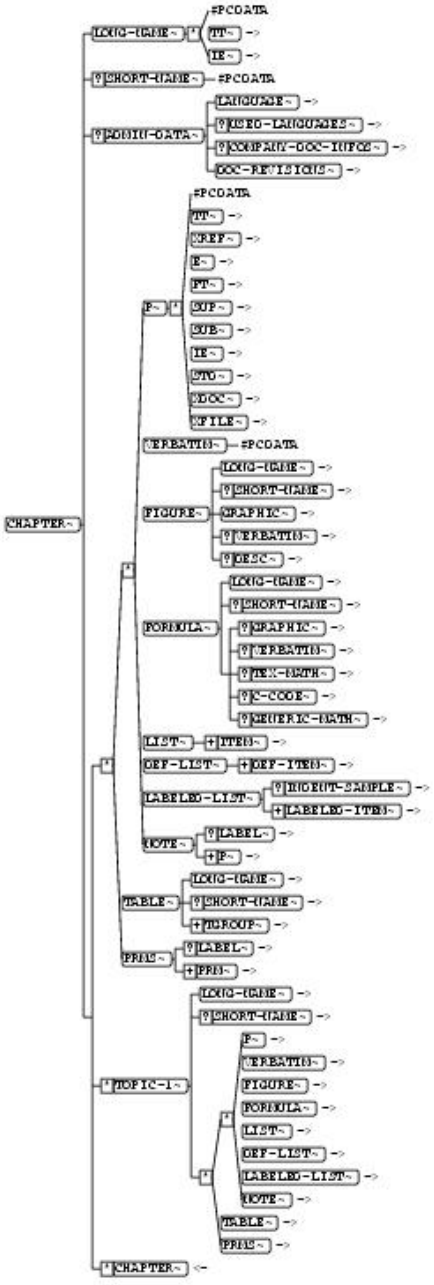
4.1.1 Chapter

<chapter> is a sequence of paragraph level elements mixed with **<chapter>**. **<chapter>**s can be nested as deeply as required. It is up to the author to make sure, that the nesting of the chapters can be handled by the processing system[26].

[24] not content orientated information

[25] DTDs with the same version of the MSR application profile

[26] In that respect, it is recommended to nest not more than four levels.



chapter.bmp

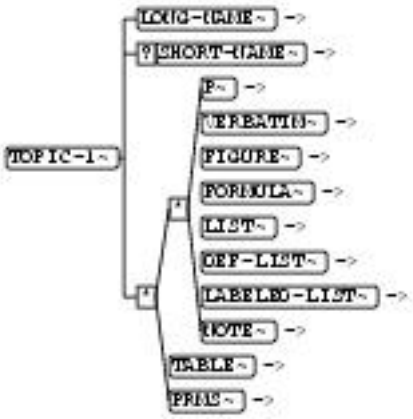
Figure 15: chapter content model

One advantage of using **<chapter>** for all levels[27] is the option to move a chapter using *cut & paste* to any place in the document at any level.

[27] the other option would be to have an extra element for each chapter level

4.1.2 Topic

Use **<topic-1>** or **<topic-2>** to create bridge titles instead of one line paragraphs with entirely emphasized contents. Note that these elements can be referenced by **<xref>**. In difference to **<topic-1>**, **<topic-2>** has no **<prms>**.



topic-1.bmp

Figure 16: Structure of <topic-1>

4.1.3 Paragraph Level Elements

”Paragraph level elements” are elements which occur on the same level as **<p>**.

The user should first look for an appropriate one among the available elements before trying to simulate things by using inadequate elements. In that respect the following hints are given:

<p>	Paragraph
<verbatim>	Preformatted text which is usually set in monospaced font. Tabs, line spaces and carrige returns are considered. Use **<verbatim>** to print program listings etc. It can even be used to show simple diagrams.
<figure>	See chapter Topic 4.1.3.2 Figure p. 41.
<formula>	See chapter Company 4.1.3.3 Formula p. 42.
<list>	A ordered or unordered list of items. For an unordered set of items, use **<list type=”unnumbered”>**. For a ordered list of items use **<list type=”numbered”>** [28].
<def-list>	Use **<def-list>** to create definition lists which might be collected into an overall definition list or a glossary. In this case **<labeled-list>** might lead

[28] Actually it is up to the rendition system if the sequence is expressed as numbers or as letters.

to the same rendition but has no information about the fact that terms are defined[29].

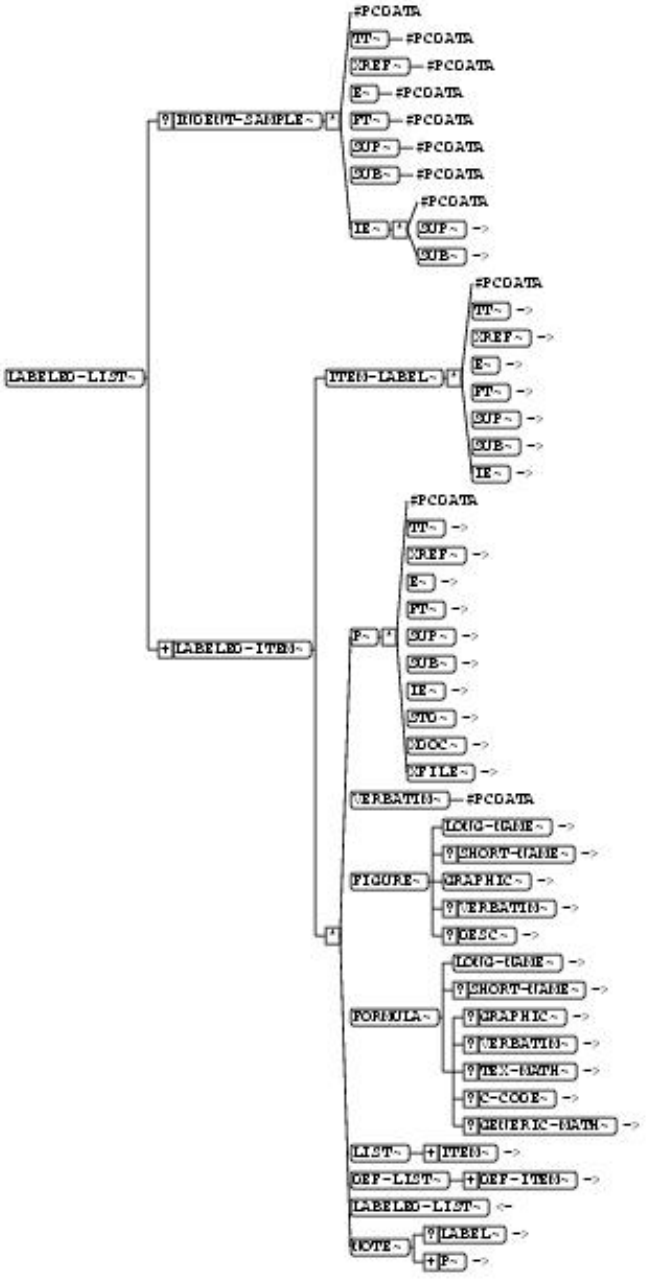
<labeled-list> Use**<labeled-list>** to create explanations or even bridge titles for very short topics instead of bulleted lists with emphasized initial words. See also Topic 4.1.3.1 Labeled List p. 39

Use **<labeled-list>** instead of two column tables if the first column cells almost contain one word.

<note> See chapter Topic 4.1.3.4 Note p. 43

[29] it is not really easy to distinguish between these two elements. As a rule of thumb, one might say that the semantics of def-list is stronger than the one of labeled-list which is more layout oriented. The items of def-list can be referenced by xref which is not possible with the items of a labeled-list.

4.1.3.1 Labeled List



labeled-list.bmp

Figure 17: Structure of <labeled list>

<labeled-list> is one of the most powerful elements. If possible it is rendered as a label followed by the item body:

XX XXXXXXXXXXXXXX
XXXXXXXXXXXXXX

xxxxxx xxxxxxxxxxxxxx
xxxxxxxxxxxxxx

xxxxx xxxxxxxxxxxxxx
xxxxxxxxxxxxxx

The indentation is determined by the *rendition system* which should take into account the biggest **<item-label>**.

Sometimes the author wants some influence to the indentation. For this respect **<indent-sample>** can receive any content which is used by the *rendition system* as a sample which must be rendered and measured to determine the indentation.

The attribute **[item-label-pos]** defines how the **<item-label>** should be handled. The default value of the attribute is **[item-label-pos]**=”no-newline”. If an **<item-label>** is wider than **<ident-sample>** the most general case is to start the item body in a new line if necessary(**[item-label-pos]**=”newline-if-necessary”):

xxxxxxxxxx

xxxxxxxxxxxxxx
xxxxxxxxxxxxxx

xxxxxx xxxxxxxxxxxxxx
xxxxxxxxxxxxxx

xxxxxx xxxxxxxxxxxxxx
xxxxxxxxxxxxxx

If the attribute has the value **[item-label-pos]**=”newline” the item-body starts generally in a new line.

Note that **<indent-sample>** can be used to adjust the indentation if there are multiple **<labeled-list>**s which should have the same indentation.

4.1.3.2 Figure

<figure> is used to insert graphics into the document. A figure can be defined in three different ways.

1. as a real **<graphic>**
2. as an ASCII graphic (**<verbatim>**)
3. as a pure textual description (**<desc>**) of the graphic [30]

The treatment of the graphic is determined by the attributes of **<graphic>**:

Do not enter annotating text to **<long-name>**in**<figure>** or **<table>**(like *Figure 1: ...*). This embellishment is the task of the processing system, not of the author. If the author adds these things, they will be there twice since the *rendition system* will add it again.

[category] Denotes the category of the graphic. This information can be used to generate more specific list of figures

[filename] Denotes the system filename where the *rendition system* can find the graphic. This is not necessarily the final format. It is up to the *rendition system* to locate the graphic in the company specific environment, to change the file extension to get the appropriate graphic representation.

[30] This can be used for systems which can’t display graphics.

The type of this attribute can be turned from *SDATA* to*ENTITY* in the DTD file in order to allow *SGML tool*s access to the file using its *entity manager*. In this case, the entity name should be chosen in the style of a filename (e.g. *crpctmt.wmf*)[31].

[fit]
- 0 figure is placed in original size. If it does not fit on the page or the available space, it is scaled down.
- 1 the figure is scaled up or down to fit the page as possible. This value will be ignored if **[width]** or **[height]** is specified in addition.
- 2 the figure is rotated counterclockwise by 90° if it is landscape and is wider than the actual text area. It is scaled down to the page size if it does not fit otherwise. This value will be ignored if **[width]** or **[height]** is specified in addition.
- 3 the figure is always rotated counterclockwise by 90°. If it does not fit on the page it will be scaled down. If **[width]** or **[height]** is specified in addition, the figure will be rotated and then scaled to the specified values.
- 4 the figure is always rotated counterclockwise by 90° and scaled up or down for best fit on the page. This value will be ignored if **[width]** or **[height]** is specified in addition.

[height] If this attribute has a value, the figure will be scaled to the defined height which is a real value with dimensions (e.g. ”10cm”, ”150mm”, ”12.5in”). If also **[width]** is specified the figure will be distorted. This value always specifies the width of the ”figure box” on the page after possible scaling/rotating.

[notation] This attribute specifies the format of the graphic file if used by an *SGML Application* supporting notations.

[scale] If this attribute receives a value, the figure will be scaled by the given factor which must be a signed real number. Numbers greater 1 increase the size of the figure, values less than 1 make the figure smaller. For example with *scale=”0.5”* the a figure of the size 10x10 cm will appear as 5*5cm.

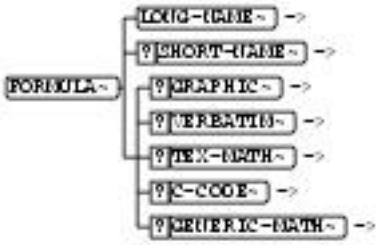
[width] If this attribute has a value, the figure will be scaled to the defined width which is a real value with dimensions (e.g. ”10cm”, ”150mm”, ”12.5in”). If also **[height]** is specified the figure will be distorted. This value always specifies the width of the ”figure box” on the page after possible scaling/rotating.

The scaling attribute precedence is:

- **[scale]** has precedence over all
- **[fit]**has precedence over **[width]**and/or **[height]**

[31] This is the way how this document is prepared. It is visible in the sgml source.

4.1.3.3 Formula



formula.bmp

Figure 18: Structure of <figure>

A formula can be described in five different ways which can exist parallel. These are:

<graphic>	A formula prerendered as a figure.
<verbatim>	A simple ASCII formula.
<tex-math>	A *TeX* math formula which can be processed by a *TeX* or *LaTeX* processor.
<c-code>	A formula which is defined as c-code.
<generic-math>	This element is intended for the definition of semantic math descriptions which can be processed by math processors. Actually there is no recommentation for the language of the formula specification or usage of a special rendering system.

It is up to the rendering system which of the available representations is used.

4.1.3.4 Note

A note is an object to express a combination of an icon with descriptive text and an additional label. This is useful for things like cautions, hints etc..

The attribute **[notetype]** defines the note category. The following values are available:

- caution
- hint
- tip
- instruction
- exercise
- other

If the attribute **[notetype]** has a value of ”other” the user has to specify a own type within the attribute **[user-defined-type]**.

A formatter has to place the right icon before the descriptive text according to the value of **[notetype]** or **[user-defined-type]**. The optional **<label >** can be used to define a title of the note.

4.1.4 Character Level Elements

Character level elements can occur within element like **<p>**, **<item-label>**. There are rendition oriented elements like **<e>** (emphasis), **<sub>** as well as semantically oriented Elements as **<tt>** (technical term) or **<std>**(referring to an external standard). It is highly recommended to use rather semantically oriented elements than rendition oriented ones.

4.1.4.1 Rendition Oriented Character Level Elements

The rendition oriented character level elements are:

<e> Emphasizes the text. The attribute **[type]**determines the rendition style.

<sub> Subscript - places the contents with smaller font below the base line.

<sup> Superscript - places the contents with smaller font above the base line.

4.1.4.2 Semantically Oriented Character Level Elements

Table 3: semantically oriented character level elements

Element	use for	example
<tt>	Use for any technical term. The type of that term is determined by the attribute **[type]** [32]. This element could be treated as a back-door to markup information which is not totally semantic. The *SGML processing system* can generate list of technical terms which makes it easier to find misspellings and other errors.	This is an SGML tag **<tt type=sgmltag>** we can collect all **<tt>**s
<xref>	Used to create links in the document. The role of the target is determined by the attribute **[id-class]** receiving the value of the target's fixed attribute **[f-id-class]** . The attributes of **<xref>** should be maintained by the *authoring system*.	

[32] it is recommended to enter the term always in singular and create plurals outside of the element (see second example).

Table 3 (Cont.): semantically oriented character level elements

Element	use for	example
<xdoc>	Used to refer to an external document which usually is not available electronically. **<xdoc>** receives a set of elements characterizing the external document	Details to architectural forms can be found in *[External Document: / URL: / Relevant Position:]* .
<ft>	Is used to create footnotes	Footnotes seem to be small and unimportant[33].
<ie>	creates index entries	It is not necessary to put SGML tags into the Index, since the processing for *MSRREP.DTD* recommends to create a list of SGML tags automatically.
<xfile>	Is used to create pointers to external files which are not to be processed by the native *SGML processing system*. The contents of **<xfile>** can be used to connect to appropriate systems in later steps of the processing chain.	The schematic is found in *[External FILE: MOTRONIC wiring diagram / URL: motronic.asc]*
<std>	Is used to refer to a standard.	SGML is defined in *[/ Standard: Information Processing - Text and Office Information Systems / Subtitle: Standard Generalized Markup Language / State: standard / Date: 1986 / URL: / Relevant Position: entire document]*

Table 4: usage of technical terms

type	use for	example
<tt type=sgmltag>	Used to describe SGML tags including attributes	To describe SGML tags use **<tt type=sgmltag>**.
<tt type=sgml-attribute>	Used to describe SGML attributes outside of tags	The sgmltag is denoted by the attribute **[type]**
<tt type=tool>	Used to mention tools used for example in a process. This can be software, as well as mechanical tools. The tool should be specified by its nature not by the specific product name.	SGML files are processed using an *SGML processing system*.

[33] But many people read footnotes first - so put the interesting stuff in footnotes

Table 4 (Cont.): usage of technical terms

type	use for	example
<tt type=product>	Used to mention specific products.	This document is processed using *MetaMorphosis*.
<tt type=variable>	Used to mention a variable informally. This is used to control the rendition as well as for generating variable lists. This is mainly for informal reports[34]. It is also possible to use this to mention a variable in the ECU software if no **<sw-data-dictionary>** is part of the document. In a later process step, this can be turned over to a formal **<xref>**	The initialization is controlled by the environment variable *MMRC*. The initial advanced angle is calculated based on *N*and*TL*.
<tt type=state>	Used to mention a state for example of a process.	The documents must at least be *revised* if they are submitted to the customer.
<tt type=prm>	Used to mention a state for example of a process. It is also possible to use this to mention a calibration parameter in the ECU software if no **<sw-data-dictionary>** is part of the document. In a later process step, this can be turned over to a formal **<xref>**	The initial advanced angle is calculated using a lookup table *KFZW* .
<tt type=material>	Used to mention material.	Furniture is usually made of *wood* and *plastic*
<tt type=control-element>	Used to mention control elements of tools like push-buttons, menu items, switches etc. as well as keyboard keys.	To finish the dialog push the *OK* button.
<tt type=code>	Used to markup program in line code sequences	*MetaMorphosis* is invoked with *mm crp.sgm*
<tt type=organisation>	Used to markup the name of an organization.	SGML is standardized by *ISO*

[34] Note that in MSRDOC.DTD there are means to formally describe variables to be used if the system software of an ECU is specified.

Table 4 (Cont.): usage of technical terms

type	use for	example
<tt type=other>	Used to mention a special term which does not fit to the other types. This is a back-door for the definition of user defined types. They have to be specified within the attribute **[user-defined-type]** . A formatter uses this user defined type only if **[type=other]** .	This is a *thing* not covered by **<tt>**.

Table 5: sub-elements for xdoc and xfile

Element	use for	example
<number>	Used to markup the document ISBN resp. the standard number	ISBN 0-7923-9432-1
<state>	Used to markup the state of the referred document resp. standard.	released
<date>	Used to markup the release date of the referred document resp. standard. This could be expressed as year only, if the exact date is not known.	1994
<publisher>	Markup the publisher of the document or the standard. This can be the author as well as the publishing organization.	Steven J. DeRose and David G. Durand / Kluwer Academic Publishers
<position>	Markup the relevant position in the referenced document resp. standard.	Chapter 5.2 - Architectural forms
<subtitle>	Used to markup the subtitle of the referenced document or standard if there is one.	HyTime
<short-name>	Used to markup the document identifier	SGML
<long-name>	Used to markup the main title of the referenced object.	Making Hypermedia work
<file>	Used to markup the file access information. This is intended to be processed by external systems.	*[External FILE: MOTRONIC wiring diagram / URL: motronic.asc]*

4.1.5 Table

<table> is implemented as *CALS table* (see *[External Document: CALS table spec / URL: / Relevant Position:]* at www.oasis.org). Capturing these kind of tables must be supported by the *SGML editor*, so only some hints are given here:

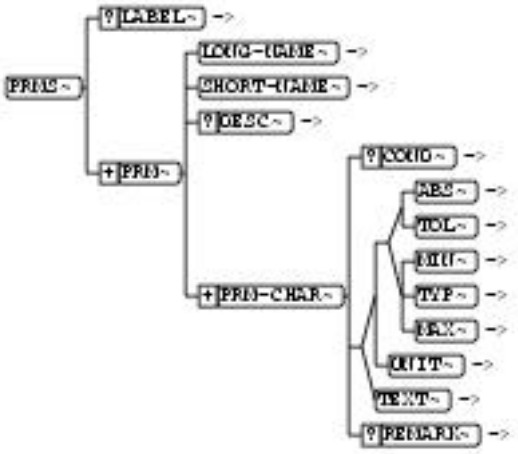
- *CALS table*s consist of mainly three parts within **<tgroup>**: **<thead>**, **<tbody>**, **<tfoot>** .
- Each part is made of **<row>**s of **<entry>**s. Each of these elements have attributes to control the layout of the table.
- **<tgroup>** also receives a set of **<colspec>**s having information about the table columns.
- One of the major problems if *CALS table*s do not work is, that the amount of **<colspec>** elements and **<entry>** does not match the value of the attribute **[columns]** in **<tgroup>** .
- Within **<entry>** most of the paragraph level elements are allowed.

Note It is highly recommended to insert **<thead>**. This creates a table heading which is repeated on each page, if a pagebreak falls into the table.

4.1.6 Parameter tables

User Definable Parameters

For structured documentation of individual numerical and/or alpha-numerical requirements, so-called parameters are available. They have the following structure:



prms.bmp

Figure 19: Structure of prms

* parameter
* long-name
* short-name
* description
* condition [36] [37]

((* absolute value and tolerance[35]or

* minimum, typical, maximum value[36])

* unit) or

* text[37]

The following representation example can be drawn from this structure:

<short-name> UB

Table 6: Parameter structure

		<prm-char>						
Element: **<long-name>**	Element: **<short-name>**	Element: **<min>**	Element: **<typ>**	Element: **<max>**	Element: **<abs>**	Element: **<unit>**	Element: **<tol>**	
Operating voltage	U_B	10,8		14,2		V		
					13,5	V	5 %	
Colour of housing		red, green and blue						
Function state		active						

- Defined Parameters

  There are many pre-defined parameters in the MSR DOC DTD. The only difference between them and user defined parameters is that the designation (long-name element) of the parameter is pre-defined.

4.2 Predefined Document Structure

The automotive systems to be described with the help of this DTD possess very different specifications. Because of this, the specification of a particular topic, e.g. ”acoustic characteristics” might not make sense or might only become necessary later on, depending on the project.

This situation was also taken into account in the DTD through the elements ”**<na>**” (not applicable), ”**<tbd>**” (to be defined) and ”**<tbr>**” (to be resolved) as shown in Figure 20 Principles of information acquisition p. 50. This is a mechanism is located at each element on chapter level and works like a check list. A user has to make a statement for each topic.

[35] For definition of an exact setting or measurement value.

[36] For definition of a typical value or value range.

[37] For definition of an alpha-numerical value.

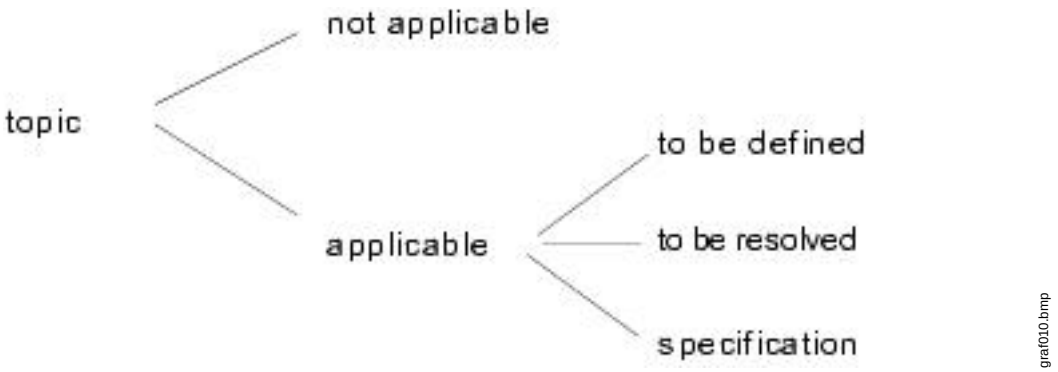


Figure 20: Principles of information acquisition

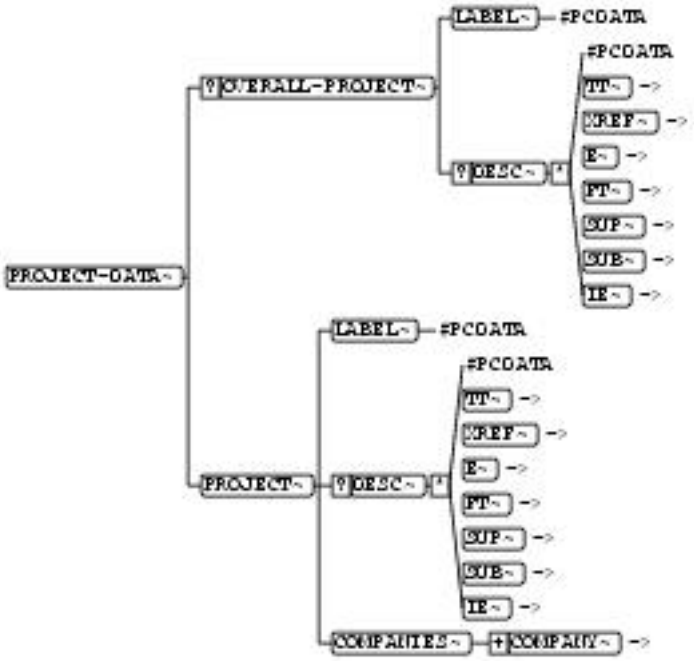
If a certain topic is not applicable it has to be marked with **<na>**. If it is applicable it can be marked with either with **<tbd>** which indicates that someone has to do a job, or it can be marked with **<tbr>** which indicates that a specification already exists but it hasn't yet been included, or a detail specification can be defined.

The elements **<na>**and**<tbr>** can be described with a short description. Within the element **<tbd>** the persons responsible for the definitions that have to been made can be specified with **<team-member-ref>**s. The schedule for the definitions can be defined within **<schedule>**.

4.3 Project Data

Registering and documenting development of a MSR system is project-oriented, whereby there may be several versions of the product data of a project. The projects can be combined with the help of main projects. This can be defined within **<overall-project>** by a **<label>** an a short description in **<desc>**. Each project is assigned to a maximum of one main project.

The documentation and continuation of project phases occurs in versions. We differentiate between active versions, the data of which can still be modified, and fixed versions, the data of which can no longer be modified. New versions can be designed on the basis of a fixed version. New versions can reuse complete fixed versions of a document or even parts of such a document. This is illustrated by the following figure:



project-data.bmp

Figure 21: Structure of <project-data>

Project data can be described by a PDM system in an integrated SGML-Editor and PDM environment. This is information on the current project and possibly the main project. Company-specific details about the project can be specified in **<general-project-data>** on the following items:

System overview **<system-overview>**	This chapter can be used to define information about a global system, e.g. a certain car model.
Order justification **<reason-order>**	This may be used to specify information about the reasons for the order of the described component resp. for making the specification of such a component.
Objectives **<objectives>**	This chapter can be used to specify information about the project objectives. E.g. ”Development and system release of the engine-managment-system for the model NEW-BEETLE”

Models **<sample-spec>**	This structure is used to define development samples like A-,B-,C-,D-sample. These samples represent the results of the different development phases.
Variant specification **<variant-spec>**	This section is used to specify all variant definitions and their corresponding variant characteristics. See also Topic 4.5 Variant Concept p. 54.
Limits to other projects **<demarcation-other-projects>**	This chapter is used to describe the demarcation to other projects.
Parallel developments **<parallel-design>**	This can be used to give an overview of the work in parallel projects.
Integration capability **<integration-capability>**	In this chapter requirements on the capabilities of integration in other systems can be described.
Acceptance conditions**<acceptance-cond>**	This chapter is used to define the general conditions for the acceptance of the described components.
Schedule and plans **<project-schedule>**	This chapter is used to define the project-schedule, e.g. project milestones, dates, time limits etc.
Purchasing conditions **<purchasing-cond>**	This is used to define purchasing conditions like amount of devices per year, delivery times, storage quantities, etc. .
Protocols, minutes of meeting **<protocols>**	This is the place where project minutes and other arrangements can be mentioned.
Handed over documents and data**<dir-hand-over-doc-data>**	This is the directory of the handed over documents and data.

Additional project specifications**<add-spec>**	Any kind of additional project description which can't be described with the chapters mentioned above.

4.4 Administrative Data

Since the respective companies explode the interchange DTD into fragments and use it for the respective acquisition DTDs (perhaps in different departments), the administrative dataappears in many places in the DTD. Each of these places can be used as such a fragment(see below).

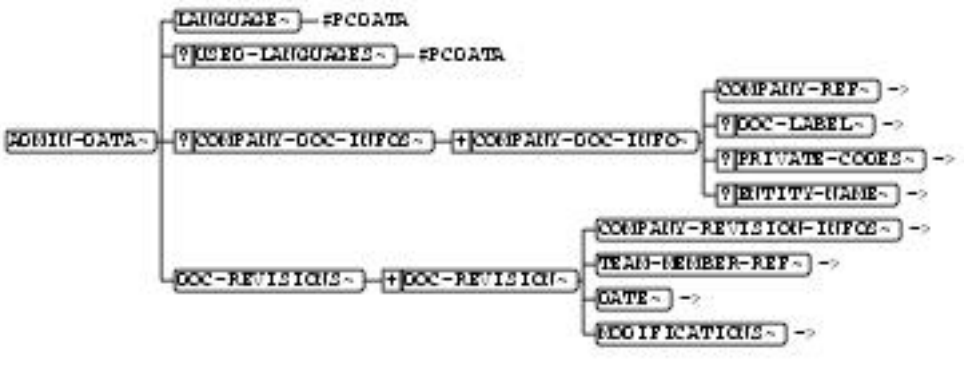


Figure 22: Support of DTD fragmentation through administrative data

The operating model is

- The document respectively the fragment is written in a certain language which can be defined in the element **<language>**. This element can be used to control a SGML system, e.g. to set the correct prefix strings for elements.
- The DTD can be configured for the multilingual operation. In this case **<language>** contains the language of the origin document. All languages used in a document have to be defined within **<used-languages>**, that is each language is defined with a **<l-10>**-element which contains the full language name and in the Attribute **[l]** the short language name (see Topic 4.6 Multilinguality p. 54).
- The document (or the fragment) is handled in all companies participating in the project.
- The data management in the various companies is different. For that reason, each participant can enter information about their document management facilities in **<company-doc-info>**:

  <doc-label> this is the label under which the document is managed in the company denoted by **<company-ref>**

  <private-code> allows to transport company specific information in a private notation. This is the place, where for example *PDMS* (*Product Data Management Systems*) can place pointers and document ids required to resynchronize after a document exchange.

<entity-name> It might be the case that each participating company uses a different fragmentation strategy. In order to support this, **<entity-name>** can receive information useable by a *split utility* which creates the desired fragments out of the entire document.

- If a new release of the document or the fragment is given, each participating site may use a specific scheme for revision numbers. For that reason, each **<doc-revision>** can receive **<company-revision-info>** which holds the participant specific information for the actual document revision.

  It is up to a *semantical check utility* to keep sure that there is only one entry per company.
- nevertheless, the actual revision is initiated by one individual denoted by **<team-member-ref>** at one certain point of time denoted by **<date>**.
- Finally the modifications made in that revision are stored in **<modifcations>** where the actual **<change>** as well as the **<reason>** for that change is notified. If possible, the change can be located by **<xref>**.
- For each **<modifcation>** the attribute **[type]** determines, if the change is made to the document only (*doc-related*) or to the subject of the document (*content-related*).

4.5 Variant Concept

Especially in the automotive sector there is a multiplicity of different variants of a part type. Normally there is not only one variant documented in the system requirements respectively the product specification of such part types.

To understand the implementation of the variant concept in the MSR DTDs, first some definitions have to be made:

Variant Characteristic: Characteristics that lead to a new variant e.g. engine, product line, country, etc. Characteristics are defined in **<variant-char>**. The characteristics have to be subdivided in three classes. These are:

- characteristics which lead to a new subject number(<variant-char **[type="new-part-number"]**>). For this only the existence of such a characteristic is enough to establish a new subject number for this variant!
- characteristics which don't lead to a new subject number (<variant-char **[type="no-new-part-number"]**>).
- characteristics which lead to a new subject number according to shaping.

Variant Definition: Definition of several variants with their variant characteristics for a part type.

Variant: A variant of a part type is defined through the values of it's variant characteristics.

Variant Coding: Allocation of all variant definitions to their corresponding subject- and drawing- numbers and the respective development versions.

4.6 Multilinguality

The MSR DTDs can be configured for multilingual operation. To use the multilingual DTD configuration the DTD switch ”multilinguality : YES or NO” have to be set.

The description of multilingual texts is made through multiple terminal elements that is multiple elements with content of #PCDATA. Multilingual elements get one of the additional language elements **<l1>**,**<l2>**, **<l3>**,**<l4>**,**<l10>** to build an aggregate of terminal elements. These language elements provide an attribute **[l]** where the language of this element can be specified. The content of the attribute **[l]** have to be defined as two-letter lower-case symbols according to the *[/ Standard: Code for the representation of names of languages / URL: / Relevant Position: Part1]*

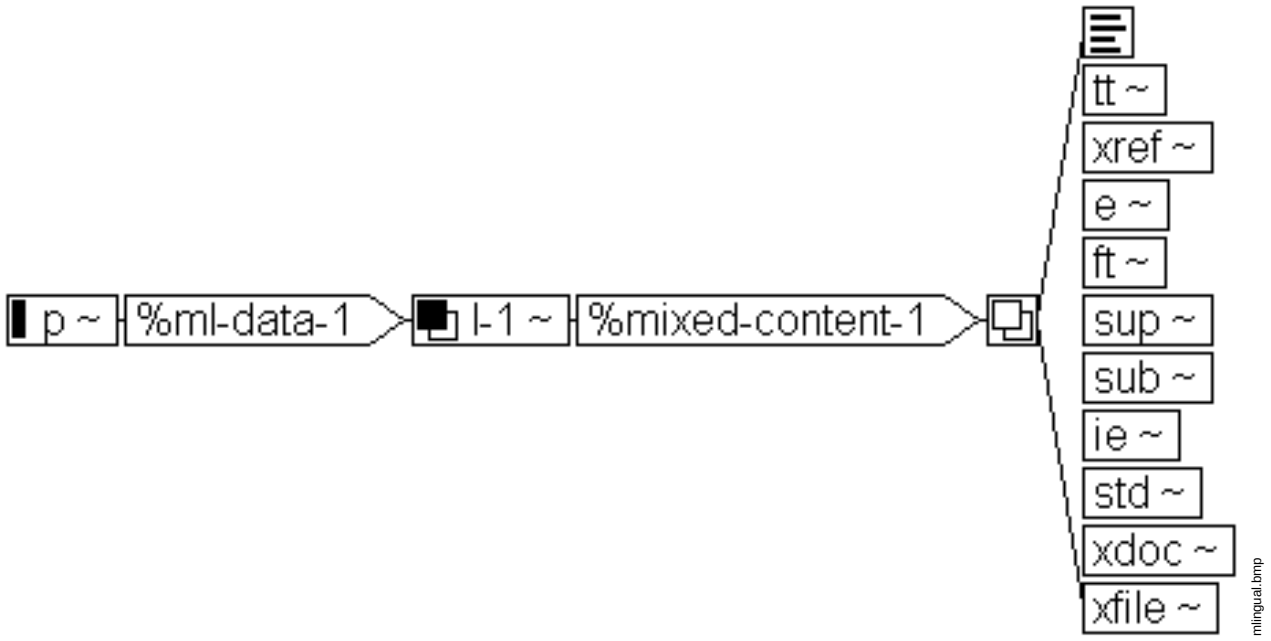


Figure 23: Multilingual Paragraph

5 Overview Changes

[snt] Structural Fundamentals of Networks

5.1 Overview to Second version with new application profile

geplant für ???

Table 7: Lösungen in Second version with new application profile

Name	Kind	Thema	State	Prio	Pd	Occurred in	P.
cmt-text		sw-compu-method-text with description	**passed**			Meeting 12.12.96	-> p. 64
resp.obj		Assignment of responsibilities for objects	**passed**			Meeting 12.12.96	-> p. 67

5.1.1 Release notes

5.1.1.1 [cmt-text] sw-compu-method-text with description

<sw-compu-method-text> can now be documented in detail by using **<desc>** within **<sw-compu-method-value-pair >**.

Migration No migration required

5.1.1.2 [resp.obj] Assignment of responsibilities for objects

<admin-data> is now allowed on **< net-port-variant>**

5.2 Overview to First version with bug-fixes

geplant für 06.02.98

Table 8: Requests zu First version with bug-fixes

Name	Kind	Thema	State	Prio	Pd	geplant für	P.
trans-mode	enh	Model transmission modes to completion	**open** H. Riegraf to submit a proposal.			First version with bug-fixes	-> p. 63

Table 8 (Cont.): Requests zu First version with bug-fixes

Name	Kind	Thema	State	Prio	Pd	geplant für	P.
docrefopt	enh	Document-overriding references optional	**passed**			First version with bug-fixes	-> p. 65

Table 9: Lösungen in First version with bug-fixes

Name	Kind	Thema	State	Prio	Pd	Occurred in	P.
trans-mode	enh	Model transmission modes to completion	**open** H. Riegraf to submit a proposal.			First version with bug-fixes	-> p. 63
basetype		SW base type in net-signal	**passed**			Meeting 12.12.96	-> p. 63
docrefopt	enh	Document-overriding references optional	**passed**			First version with bug-fixes	-> p. 65
init-opt	enh	init-value optional	**passed**			First version	-> p. 65
msg-set	enh	Messages in individual chapters	**passed**	A 1		First version	-> p. 66
msgid-mask	enh	Same message from different senders	**passed**			Meeting 12.12.96	-> p. 66

5.2.1 Release notes

5.2.1.1 enh: [trans-mode] Model transmission modes to completion

<event-mode> and **<cycle-mode>** to be removed until formal formal models are available.

5.2.1.2 [basetype] SW base type in net-signal

<net-signal-spec-variant> now has **< sw-base-type>**.

5.2.1.3 enh: [docrefopt] Document-overriding references optional

The potential document-overriding references to be optional and assigned a label. *MSR-NET.DTD* can therefore be used autonomously by this.

5.2.1.4 enh: [init-opt] init-value optional

<init-value> is now optional in **< net-signal-spec-variant>**

5.2.1.5 enh: [msg-set] Messages in individual chapters

<memory-layout> is now renamed in **<net-message-layout >**.

Network messages can now be grouped in **<net-message-set>**s. The grouping is intended for messages having the same contents and serves to group the documentation.

If such an entity is read into a non-standardized database (i.e. the **<short-name>** of the **< net-message-set>** are allocated to the **<net-message>**), then the **<short-name >** serves as the criterion for changing groups when restoring a MSRNET entity.

5.2.1.6 enh: [msgid-mask] Same message from different senders

5.3 Overview to First version

geplant für 18.11.97

Table 10: Requests zu First version

Name	Kind	Thema	State	Prio	Pd	geplant für	P.
non-can		Extension to non-CAN networks	**open**				-> p. 62
sw-var-abgl		Agreement between MSRSW variables and MSRNET signals	**open**				-> p. 62
init-opt	enh	init-value optional	**passed**			First version with bug-fixes	-> p. 65
msg-set	enh	Messages in individual chapters	**passed**	A 1		First version with bug-fixes	-> p. 66

Table 11: Lösungen in First version

Name	Kind	Thema	State	Prio	Pd	Occurred in	P.
ctrl-mode	enh	Where is controller mode specified	**passed**			Meeting 12.12.96	-> p. 64
var-trans	bug	Variants in transmission spec.	**passed**			Meeting 12.12.96	-> p. 68

5.3.1 Release notes

5.3.1.1 enh: [ctrl-mode] Where is controller mode specified

<adress-mode> is now renamed to **< identifier-type>** which can take the values *standard* or *extended*.

<node-variant> now has **<protocol-conformance >** to capture the level of conformance the node provides.

5.3.1.2 bug: [var-trans] Variants in transmission spec.

<transmission-spec> is no longer variant-dependent.

5.4 Overview to Introduction of Net-Ports

geplant für 12.6.96

Table 12: Requests zu Introduction of Net-Ports

Name	Kind	Thema	State	Prio	Pd	geplant für	P.
Netport-Node	enh	Handling nodes with several inflexible net-ports	**passed**			Meeting 22.8.96	-> p. 67

5.5 Overview to Meeting 22.8.96

geplant für 22.8.96

Table 13: Lösungen in Meeting 22.8.96

Name	Kind	Thema	State	Prio	Pd	Occurred in	P.
Netport-Node	enh	Handling nodes with several inflexible net-ports	**passed**			Introduction of Net-Ports	-> p. 67

5.6 Overview to Meeting 12.12.96

geplant für 12.12.96

Table 14: Requests zu Meeting 12.12.96

Name	Kind	Thema	State	Prio	Pd	geplant für	P.
gateway		How will gateway functionality be handled	**open**				-> p. 61
multi-arch	enh	Support of multi-architecture networks	**open** BMW, VW, Porsche to be discussed further				-> p. 61
net-class	enh	Network-protocol classes	**open**				-> p. 62
supersignal		Supersignal	**open**				-> p. 62
basetype		SW base type in net-signal	**passed**			First version with bug-fixes	-> p. 63
cmt-text		sw-compu-method-text with description	**passed**			Second version with new application profile	-> p. 64
ctrl-mode	enh	Where is controller mode specified	**passed**			First version	-> p. 64
msgid-mask	enh	Same message from different senders	**passed**			First version with bug-fixes	-> p. 66
resp.obj		Assignment of responsibilities for objects	**passed**			Second version with new application profile	-> p. 67
seg.na	enh	Segmentation is not always entered	**passed**	A1			-> p. 68
var-trans	bug	Variants in transmission spec.	**passed**			First version	-> p. 68
lnk-sw		How are signals allocated to SW variables	**rejected**				-> p. 69

6 Changes

6.1 [gateway] How will gateway functionality be handled

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	Rauleder	**open**			[11.0] p. 59	

Subject How are gateway functions described? How do links come to *MSRSW.DTD* entities. Is only the signal-signal arrangement to be considered here?

Begründung Background is the issue of the different degree of formalization in the stress field between *MSRSW.DTD MSRNET.DTD MSWDOC.DTD*

Beschlossene Lösung

Still to be resolved

6.2 [multi-arch] Support of multi-architecture networks

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	BMW H. Kramer	**open** BMW, VW, Porsche to be discussed further			[11.0] p. 59	

Subject The possibility must be given to describe networks that work using different protocols and architectures. The following shall thereby be observed

Gateway The entire network consists of several descriptions with separate signal definitions. The sub-networks are ignorant of each other.

Repeater The entire network has one consistent description and one common signal definition.

Begründung Scenarios occur in practice, in the first instance however gateways

Lösungsansatz 1

Several network specifications to be introduced, whereby one common sub-network component occurs. The sub-network components can reference the common network. Without sub-networks, the common network is the entire network. The following structure is thus given.

```
common-network
   net-signal-spec
   net-message-spec
sub-net*
   net-signal-spec
      net-signal
```

```
   net-signal-ref
net-message-spec
   net-message
   net-message-ref
```

A model for gateways shall be found in addition to this.

6.3 [net-class] Network-protocol classes

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	BMW H. Krammer	**open**			[11.0] p. 59	

Subject The possibility must be given to specify the protocol type for a network.

Begründung There are different types of networks

6.4 [non-can] Extension to non-CAN networks

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	AG-Net 18.11.97	**open**			[1.1.0] p. 58	

Subject The extension to non-CAN networks must be included.

Begründung Non-CAN can also be given

6.5 [supersignal] Supersignal

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	BMW H. Krammer	**open**			[11.0] p. 59	

Subject The possibility must be given to group different types of signals.

Begründung Improvements in the documentation.

Lösungsansatz 1

Solution analogous to **<port-groups>** in MSRDOC

Lösungsansatz 2

Signals can be grouped using a cluster. This is one element with several indicators to **<net-signal>**.

6.6 [sw-var-abgl] Agreement between MSRSW variables and MSRNET signals

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	AG Net 18.11.97	**open**			[1.1.0] p. 58	

Subject: Agreement between **<sw-variable >** and **<net-signal-spec-variant>** in future versions

Begründung: These data are closely related.

See also: **other:** Chapter 6.8 [basetype] SW base type in net-signal p. 63

6.7 [trans-mode] Model transmission modes to completion

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	Riegraf, Auf-dermauer	**open** H. Riegraf to submit a proposal.			[1.1.1] p. 56	[1.1.1] p. 56

Subject: The transmission modes **<event-mode>** bzw. **< cycle-mode>** should be modeled to completion

Begründung: In order that a formal handling in tools can follow

Beschlossene Lösung

Initially remove the informal **<event-mode>** and **< cycle-mode>**. A formal proposal will be included when proposed.

Release notes

1.1.1enh: [trans-mode] Model transmission modes to completion p. 57

6.8 [basetype] SW base type in net-signal

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	BMW H. Krammer	**passed**			[11.0] p. 59	[1.1.1] p. 56

Subject: The possibility should be provided to give a base type for a signal

Begründung It cannot be determined from the conversion formula alone whether this can be a value with a prefixed sign.

Beschlossene Lösung

<sw-base-type> to be included in **<net-signal-spec-variant >**.

Matching between **<sw-variable >** and **<net-signal-spec-variant>** must be possible in future versions

Release notes

1.1.1[basetype] SW base type in net-signal p. 57

6.9 [cmt-text] sw-compu-method-text with description

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	BMW H. Krammer	**passed**			[11.0] p. 59	[2.1.0] p. 56

Subject **<sw-compu-method-text>** should have a further element **<cmt-desc>** where additional information on a certain value can be described.

Begründung It can be that **<cmt-text>** is a formal text that shall be described in detail in the documentation.

Beschlossene Lösung

An element **<desc>** will be introduced within **< sw-compu-method-value-pair>**.

Release notes

2.1.0[cmt-text] sw-compu-method-text with description p. 56

6.10 [ctrl-mode] Where is controller mode specified

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	Riegraf / Weichel	**passed**			[11.0] p. 59	[1.1.0] p. 58

Subject Statement from Team-Member Dipl.-Ing. T. Riegraf p. 9 on **<identifier-type>**: A better designation is ”CAN-data transmission format”. This is a very global definition for a bus and can be ” *Standard-Format*” or ”*Extended-Format*”. Specifications from the Bosch company exist for this. It can be defined in the case of ”*Extended-Format* ” for each *control unit*/CAN controller, whether it is ”*active*” or ”*passive* ”, i.e. whether communication with the Extended-Format is possible or not. Clarification is required as to where this definition is made.

Begründung A gap is otherwise given

Beschlossene Lösung

- **<adress-mode>** to be renamed in **<identifier-type >**. This can take the value of *standard* or *extended*.
- **<node-variant>** receives a new element **<protocol-conformance>** and can take the following values

CAN-2.0A-active	Definition CAN conformance 2.0A p. 70CAN Conformance 2.0A
CAN-2.0B-active	Definition CAN conformance 2.0B p. 70CAN Conformance 2.0B
CAN-2.0B-passive	Definition CAN conformance 2.0B passive p. 70CAN Conformance 2.0B passive
NA	The network described knows no conformity classes. The value can therefore not be filled out.

Release notes

1.1.0enh: [ctrl-mode] Where is controller mode specified p. 59

6.11 [docrefopt] Document-overriding references optional

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	Klein	**passed**			[1.1.1] p. 56	[1.1.1] p. 56

Subject Document-overriding references should be optional.

Begründung Simplifies the handling in simple author environments. Resolution by the AG-DTD

Beschlossene Lösung

The potential document-overriding references to be optional and assigned a label. *MSR-NET.DTD* can therefore be used autonomously by this.

Release notes

1.1.1 enh: [docrefopt] Document-overriding references optional p. 57

6.12 [init-opt] init-value optional

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	riegraf	**passed**			[1.1.0] p. 58	[1.1.1] p. 56

Subject **<init-value>** shall be optional.

Begründung The information cannot always be provided, especially in the case of block modes.

Beschlossene Lösung

<init-value> is now optional in **< net-signal-spec-variant>**

Release notes

1.1.1 enh: [init-opt] init-value optional p. 57

6.13 [msg-set] Messages in individual chapters

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	Kramer	**passed**	A 1		[1.1.0] p. 58	[1.1.1] p. 56

Subject The possibility should be given to categorize network messages such that these can be distributed amongst individual chapters for printed documentation.

Begründung Better overview of the documentation

Beschlossene Lösung

A group concept will be introduced for messages **< net-message-set>**. ”Group” was not used since ”group” groups conceptually identical types together. It is only for *MSRSYS.DTD* that the possibility ”group” is given for ports to use contents and logical grouping.

Release notes

1.1.1enh: [msg-set] Messages in individual chapters p. 58

6.14 [msgid-mask] Same message from different senders

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	BMW H. Kram-mer	**passed**			[11.0] p. 59	[1.1.1] p. 56

Subject It must be possible to allocate several senders to one message. These differ in variants and identifiers.

Begründung It is possible that the same message can be sent by different senders and differing identifiers. The receivers respond to a masked identifier. Example is a window lifter where each contact pushbutton depresses the same message.

Furthermore, variant-handling is to be supported.

Lösungsansatz 1

Two mechanisms are provided for determining the identifier for a specific **< net-message>**:

- Discrete information on the identifiers **<net-message-identifiers >**.
- A computation of the identifiers (**<calc-net-message-identifiers >**) as the sum of a basis and **<Identifier-base-adress >** and an offset that is given in the node. The **< msg-identifier-offset>** is given by the respective **<sender >**. This procedure is used preferentially for network management and diagnostics.

Beschlossene Lösung

A group concept will be introduced for messages as well **< net-message-group>**

Release notes

1.1.1enh: [msgid-mask] Same message from different senders p. 58

6.15 [Netport-Node] Handling nodes with several inflexible net-ports

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	??	**passed**			[10.01] p. 59	[10.02] p. 59

Subject It is currently such that each net-port appears as a separate node in the segmentation.

Begründung This is inflexible. The relation between net-ports is is not transparent in the network document.

Lösungsansatz 1

The mention of net-ports in the node could be dispensed with. The net-port could be included in the segmentation when referencing the node.

Beschlossene Lösung

Net-ports have been set up within nodes (refer to Topic 3.2.2.2 Description of network nodes p. 22). The segmentation refers to the**< net-node-port>**s.

Release notes

6.16 [resp.obj] Assignment of responsibilities for objects

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	BMW H. Kram-mer	**passed**			[11.0] p. 59	[2.1.0] p. 56

Subject It should be possible to explicitly assign the responsibility for certain sub-trees.

Begründung The participants must know their partners.

Lösungsansatz 1

<Admin-data> will be propogated further, a **<team-member-ref>** will introduced in **< company-doc-info>**.

Beschlossene Lösung

References to objects will be introduced for the **<Team-Member>**s, similar to **<role>**.

<admin-data> will also be introduced for **<net-node-variant >**

Release notes

2.1.0[resp.obj] Assignment of responsibilities for objects p. 56

6.17 [seg.na] Segmentation is not always entered

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
enh	BMW. H. Kram-mer	**passed**	A1		[11.0] p. 59	

Subject Segmentation should also permit **<na>**. In principle, this should be executed such that differing scenarios can be covered, similar to msrsw.

Begründung It can be that the network shall be described in abstract terms, i.e without segmentation. This is currently not permitted by the DTD.

Beschlossene Lösung

<setmentation> and its twins will receive **< na>** etc.

6.18 [var-trans] Variants in transmission spec.

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
bug	Weichel	**passed**			[11.0] p. 59	[1.1.0] p. 58

Subject **<transmission-spec>** is not yet variant-dependent even though this is contended by Topic 3.4 Variants p. 35

Begründung Inconsistency

Beschlossene Lösung

Documentation corrected in Topic 3.4 Variants p. 35.

Release notes

1.1.0 bug: [var-trans] Variants in transmission spec. p. 59

6.19 [lnk-sw] How are signals allocated to SW variables

Kind	proposed by	State	Prio	days	Occurred in	Geplant für
	Rauleder 18.1.97	**rejected**			[11.0] p. 59	

Subject How do the links come to *MSRSW.DTD* entities. Is only the signal-signal allocation to be considered here?

Begründung We have no possibility of executing this arrangement.

Lösungsansatz 1

<net-sig-var-assocs > will be introduced in parallel to **<locs>**. The arrangement of net-signal to the variable can be described here at a suitable point.

pro
- Resolves the given problem

con
- This information is more a part of the software and should be described here.

Beschlossene Lösung

To be handled with *MSRSW.DTD*.

App. A Glossary

Sampling point Determines as a network parameter the point in time when the bit is sampled.

Address mode The address mode for a message can be ”Standard” or ”Extended”.

ASAP The ASAP interfaces have been agreed by the ”Study Group for the Standardization of Application Systems (ASAP)”. Members of this study group are the German autombile manufacturers and companies in the supplier industry.

Basic CAN CAN Chip; possesses only one sender and one reception memory.

Baud rate

Operating mode Transmission parameter for a message that describes for instance, whether remote operation is present or not.

Bit stuffing A technique that is used for resynchronization in bit-oriented network protocols. A change in the flank is caused by the inclusion of an additional complementary bit according to a defined number of bits of the same level. The additional bit is removed again by the receiver.

BRT Abbreviation for baud rate prescaler, defined as the number of BTL cycles.

BTL cycles Abbreviation for Bit Timing Logic cycle; a sampling frame is formed via the baud rate prescaler from the oscillator frequency of the chip. This auxiliary cycle (BTL cycle) serves to determine the time segments for the bit timing.

Byte order Network parameter that defines the sequence for the higher-value and the lower-value byte within a word. Since the processor manufacturers ”Intel” and ”Motorola” have made contrary definitions for the byte order, the values ”Intel”, ”Motorola forwards” and ”Motorola backwards” are possible for this parameter.

CAN Abbreviation for Controller Area Network

CAN High Speed Network link and network medium according to *[/ Standard: To be defined / State: DIS / URL: / Relevant Position: Entire document]*, amongst others, characterized by a data rate of between 125 Kbit/s and 1 Mbit/s.

CAN Low Speed Network link and network medium according to *[/ Standard: To be defined / State: DIS / URL: / Relevant Position: Entire document]*, amongst others, characterized by a data rate of up to 125 Kbit/s

CAN conformance 2.0A CAN with an identifier length of 11 bits (standard).

CAN conformance 2.0B CAN with an identifier length of 29 bits (Extended). All CAN controllers that can process CAN 2.0B telegrams are also capable of processing standard frame, i.e. to both send as well as to receive.

CAN conformance 2.0B passive CAN controllers that only process standard frames and that are passive in their response to CAN version 2.0B. These ignore extended telegrams according to version 2.0B.

CANdb CAN data basis

CRC Abbreviation for Cyclic Redundancy Check; included the error code over all previous possitions. The CRC checksum is used for fault detection yet not for fault corrections.

CSMA/CD Abbreviation for Carrier Sense Multiple Access with Collision Detect

DLC Abbreviation for Data Length Code; gives the length of the message in bytes.

Readiness to receive following power-down Transmission parameter for a message that describes the time following a *power- down message*, during which a reaction to incoming messages is possible even though sending is not possible. It is prevented by this that a component goes into the sleep mode when a message is en route to the component.

Readiness to receive following power-up Transmission parameter for a message that describes the point in time following power-up, as of when the message can be received.

Error value Value for a signal if a fault is detected and the actual information can no longer be transmitted.

Full CAN CAN chip that, unlike the basic CAN, includes several sending and receiving storage devices.

Initialization value Value of a signal before the first reception in the receiver.

Node A network node that includes any participant or branch of the network.

Latency time Transmission parameter for a message that describes the transmission time in the network. The latency time is the time by which the the sending process for a message can be delayed by messages having a higher priority. In technical terms, the latency time is the period of time between setting the TransmitRequestBit and receiving the AckMessage. The maximum latency time for this message can thus be specified in the message.

Multiplex-signal group Several signal groups can be grouped together in a multiplex-signal group that can be identified by a respective (group) value. A receiver of the message can filter out the signal group intended for the receiver from this value.

Network EMC design Description of the designs or set-up of the network with regard to electromagnetic compatibility.

net-signal group ...

net-signal class The net-signal class classifies the net-signals according to application signals and network management signals.

OSEK To be defined

Phase relation to other messages Transmission parameter for a message that describes whether there is a phase relation with other messages present.

Physical range Permissible range for the physical (external) values for a signal that are given on account of the conversion from the network-internal presentation.

Quartz frequency The frequency of the quartz or quartz oscillator connected to the CAN controller.

RTR Abbreviation for Remote Transmission Request; a flag for identification of a data request. A sender is requested to tranmsit a corresponding data telegram as the response.

Transmission conditions Transmission parameter for a message that describes the transmission conditions.

Readiness to transmit following power-up Transmission parameter for a message that describes the point in time following *power-up* as of which the message can be sent.

SJW Abbreviation for Synchronization Jump Width that is given in BTL cycles. Defines the maximum amount by which the CAN controller may extend or shorten both major segments of the bit time.

Sleep/wake-up mechanism A mechanism that puts the network into a " *dormant state*" with low power consumption (sleep) and re-activates this as required (wake-up); e.g. for a central-locking system.

SLIO Abbreviation for Serial Linked I/O.

Memory layout For a message, the memory layout is a (graphical) supplement to the description of signal lists and multiplex-signal groups that describes the signal distribution of a message in the memory.

Conversion Procedure, on how to convert the network-internal presentation (internal values) of a signal in the corresponding physical values (external values).

Prescaler Or baud-rate prescaler; for generation of the BTL-cycle frequency from the quartz frequency.

Cycle time (Also cycle tolerance time), transmission parameter for a message that describes the tolerance time for a message for a cyclic control system (cycle sending of the message).

App. B Bibliography

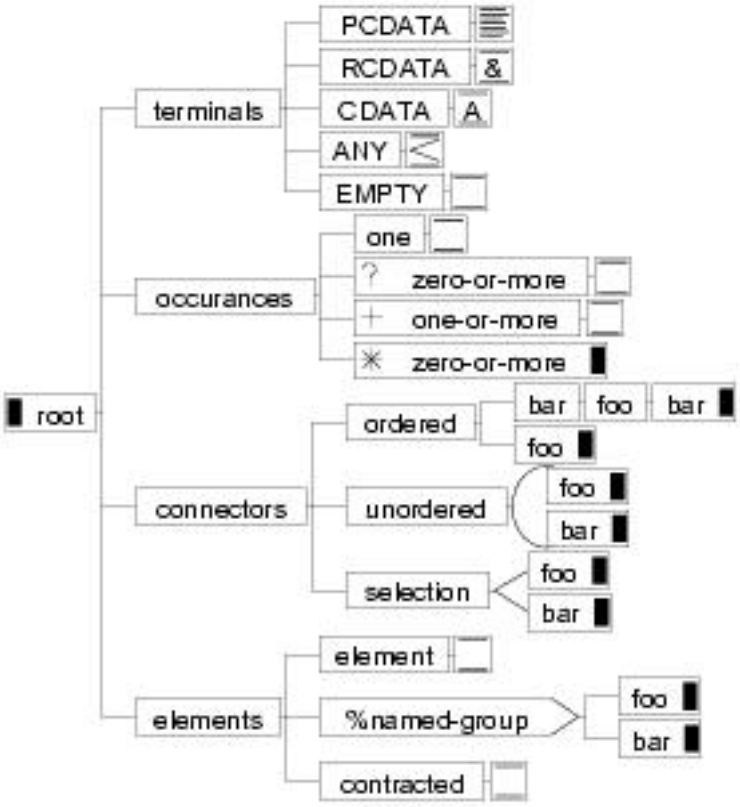
[External Document: CAN-ER-Modell / Date: 1988 / Publisher: Robert Bosch GmbH / URL: / Relevant Position:]

[External Document: Analysis of the CAN database, documents of the Bosch company, / Document number: / State: / Date: 17.04.1989 / URL: / Relevant Position:]

[External Document: CAN-Contoller Area Network, Principles and Practice / Date: 1984 / Publisher: Hüthing Publishers / URL: / Relevant Position:]

[External Document: CAN Controller Area Network, Principles, Protocols, Modules, Applications / Date: 1994 / Publisher: Carl Hanser Publishers Munich Vienna, Munich / URL: / Relevant Position:]

App. C Legend of structural grafiphics



legend.bmp

Figure 24: Meaning of symbols in the structural graphics

App. D Alternatives considered for linking control unit and network

App. D.1 Linking via individual signals

In case a), the individual signals of the network connection (e.g. *Low*, *High*, *Shield*) are specified and applied to the corresponding ports for the connection specification.

For b), only one net-signal is applied and applied to all ports.

App. D.2 Use of nomenclature conventions

The names for the individual lines for the network are handled as normal signals. The arrangement of the control-unit signals to a network is by means of a nomenclature convention, in which the designation for the network appears again in the signal name as the prefix. Signals for a network CANx could take the names CANx_High, CANx_Low, and CANx_Shield.

This variant does not suffice since in case c) in Figure 2 Linking network and control unit p. 15, it is not clear which lead-through is applied to the network.

These nomenclature conventions can also be realized in that signals in **< signal>** receive an additional element **< net-name>** to allocate the individual signals to the network.

App. D.3 Introduction of a net-signal element

A new type of signal definition is introduced in addition to the existing signal description. This can be made within the scope of the signal specification **< signal-spec>** or within a signal group **< signal-group>**.

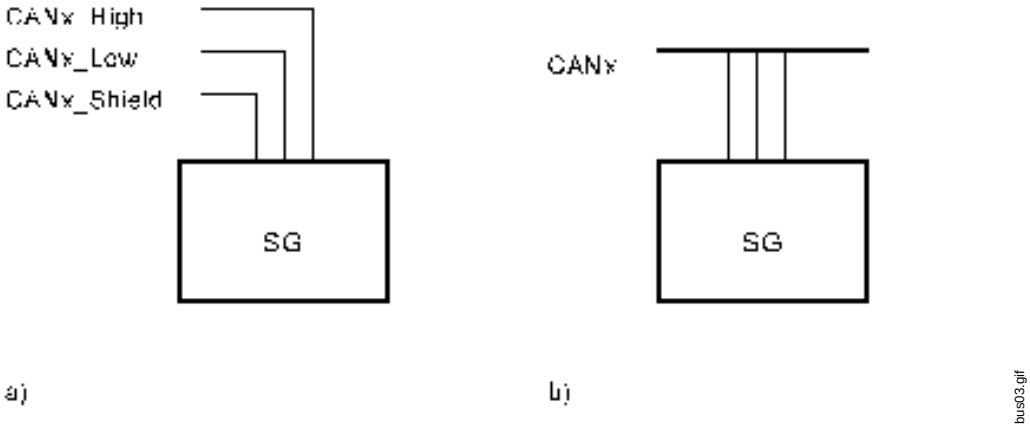


Figure 25: Arrangement of net-lines to the control unit (SG)

```
a)
<signal-group>
   <signals>
   <net-lines>
      <long-name>
      <short-name>
      <variant-def-refs>

b)
<signal-spec>
   <signal-groups>
   <net-lines>
```

a) **<signals>** and **<net-lines>** are equivalent to the extent that means are already available for the **<net-lines >** for designation of the network. That **<signals>** are included in **< net-lines>** is attributable to the convention whereby elements occurring several times should have an envelope element. From the purely logical viewpoint, there is a large number of **<signal >**s occuring as for **<signals >** as well. Three **<signal-group>**s for example must be applied for one network.

```
<signal-group>
  <signals>
  <net-lines>
     <long-name>           Designation of the network
     <short-name>          Designation of the network
     <signals>
  <signal-spec-variants>
```

b) This approach is equivalent to a). Three **<signal-group >**s are to be aplied here as well. The hierarchy is however somewhat flatter.

```
<signal-group>
   <signals>
   <net-lines>
       <net-line>
          <long-name>      Signal designation
          <short-name>     Signal designation
          <net-name>        should consequently be <net-ref>

                           is basically an indicator from the
                           MSRDOC-entity in the MSRNET entity.
          <variant-def-refs>
```

c) This variant describes the network closed. Three **< signal-group>** elements are applied for one network here as well, though these are grouped as one. This variant has the disadvantage that signals are defined at different points. If then all signals of one component shall be searched, then the search must be carried at several locations. This is in contrast to the advantage that one network can be described completely as such.

```
<signal-spec>
   ...
   <busses>
      <bus>
         <long-name>       Bus designation
         <short-name>
            <signal-groups>
```

This variant does not address the problem in figure b) either Figure 2 Linking network and control unit p. 15. Additional elements that modify the signal name would be needed to cover this.

App. D.4 Referencing network elements for signal definition

Network signals are treated as signals. The identification as a network signal is by means of a reference to a new network element.

```
<architecture>
   <buses>
      <bus>
         1
         2    <-
                |
<signal> -> <bus-ref>
```

Networks are introduced within the architecture. Alternatively, each signal can be assigned exactly to one network. This is a further development of Topic App. D.3 Introduction of a net-signal element p. 75 in that the **<net-name>** here is standardized.

```
<architecture>
  <buses>
     <bus>
        <long-name>
        <short-name>
        <desc>
        <add-info>

<signal>
   <bus-ref>
```

Does not resolve the problematical nature of Figure 2 Linking network and control unit p. 15 case b).

App. E Defining ranges of values

App. E.1 Node type

TBD

App. E.2 Topology type

TBD

App. E.3 Net-signal class

TBD

App. F Zusätzliche Anmerkungen zum Dokument-stand 2 am 23.7.98

Status dieses Dokument: cd

Documentadministration

Overview of Changes

Total	Documentpart	Nr.	Change	Reason	Related to
30.05.96		12	Conversion to SGML, editorial revisions	Quality enhancement	Document
			Conversion to SGML	Study group decision	Document
			Incorporation of results from meeting 13.3.96	Study group decision	Content
			Deleted title from figures and tables	These are created by the formatter.	Document

Total	Documentpart	Nr.	Change	Reason	Related to
05.06.96		11	Fundamental revison of the structure	Coverage of different network topologies and network types - Standardization with MSR SW	Content
			Fundamental revisions to coupling of NET and control unit. Inserted chapter structure	Clarity, and coverage of 2b	Content
			Bus renamed as net.	Description not only of bus topologies.	Document
			Bus entity to net port	Because there are no bus entities	Content
			Notes on network management revised (refer to Topic 3.3.1 General network management p. 28)	Update	Content
			Block mode no longer has separate structure (refer to Topic 3.3.5 Block transmission modes p. 29)	The messages in the block transmission modes can also be described by normal messages.	Content
			SW compu method included	Standardization	Content
			Optimum introduction of physical properties for the net-signals.	Support of block mode and miscellaneous signals	Content
			Glossary updated	Update	Content
			Transmission/reception readiness optional	These data arise primarily only for network management signals	Content
			Explicit inclusion of multiplexer.	Greater transparency for the user	Content
			Offsets are principally with reference to message start (refer to Topic 3.3.7 Messages p. 31), Multiplexer signal group interlaced.	Greater flexibility	Content
			Chapter on variants introduced (refer to Topic 3.4 Variants p. 35)	Preparation for handling variants.	Document
12.06.96		10	BUS on NET changed	Correction to report	Content

Total	Documentpart	Nr.	Change	Reason	Related to
22.8.96		9	Converted to dtd16.6	Quality enhancement	Document
			Topic 3.2.2.1 Network lines p. 22: Terminal resistance replaced by terminating protective circuit	Networks can have more complex terminating protective circuits	Content
			Topic 3.2.2.1 Network lines p. 22: Structure of nodes replaced by node port	Coverage of case 2b)	Content
			Topic 3.2.2.1 Network lines p. 22: Node number replaced by node name (short-name)	Short designation for node is not only numerical	Content
			Topic 3.2.1 Connection components p. 21: Definition of connecting components now made within a MSRNET entity	Network descriptions thus ”self-contained”	Content
			Topic 3.3.6 Network signals p. 29: Removed reference to physical signal	Avoidance of document-overriding references. Such a reference would be semantically unclear.	Content
			Conversion formulae to be defined within MSRNET	MSRNET to be self-contained - conversion formulae can be included as entities.	Content

Total	Documentpart	Nr.	Change	Reason	Related to
09.10.96		8	Editorial and structural improvements (handling faults, terminal resistance)	Feedback from company-internal reviews	Document
			Editorial revisions, corrected typing errors	Improvement of readability	Document
			Company 2.1 Requirements p. 14 in Topic 2 Arrangement of network and control units p. 14 Inserted	Improvements in the structure	Document
			Introduction of network spec. in Topic 2.2 Description of network connections in MSRDOC p. 15 detailed justification	Greater clarification	Document
			DTD's renamed as Figure 1 MSRDOC.DTD and MSRNET.DTD p. 12	Correction of errors	Document
			Introduced Topic 3.3.3 Error handling p. 29	Ensure that handling errors will be explicitely processed and not lost in the general description.	Content
			Terminal resistance replaced in structure as well by interface circuit in Topic 3.2.2.1 Network lines p. 22	Rectification of errors	Content

Total	Documentpart	Nr.	Change	Reason	Related to
21.04.97		7	Editorial revisions	Enhancement of readability	Document
			Update of graphics	Graphics were not clear	Content
			Deleted the chapter ”Alternative network description concepts”	Chapter was not comprehensible	Content
			Improvement in the presentation, refinements in the descriptions of netports, net-interfaces, block modes, net-signals, messages	Addition of further aspects of networks for completion, clarification of details	Content
			Introduced element names	Synchronization with DTD development	Content
			Remarks on gateway functionality	Clarification	Content
			Referencing of netports extended to include reference to part	Networks are linked with parts, not with part-types	Content
			Detailed description of network interface	Description of all aspects of the network	Content
			Introduce description of block modes	Description of all aspects of the network	Content
			net-signal and net-signal groups Topic 3.3.6 Network signals p. 29 Detailed	Clarification	Content
			Systemized structure of messages	Simplification and clarification	Content
05.11.97		6	Included requirements from discussions with BMW		Content
18.11.97		5	Included results of study group AG-NET 18.11.97	Requirements for implementation 1.1.0	Content
18.2.98		4	Editorial revsions to the report	Documentation for 1.1.1	Content
			Inclusion of changes in changes for 1.1.1	Detailed discussions showed the necessity	Content
			Summary of messages for documentation purposes.	Clarification	Content
10.7.98		3	English translation	Publish the final document	Content
			moved to new document strategy	adoption of MSR-TR-DOV	Content
			inserted msr application profile	completeness	Content
23.7.98		2	updated description of msr application profile		Content

Total	Documentpart	Nr.	Change	Reason	Related to
2002-02-07		1	Create index, technical terms and reference. Convert to MSRREP V210 XML.		Content

Versions Overview

Document Part	Date	Editor			
		Company	Version	State	Remarks
From page 9	2002-02-07	Dipl.-Ing. R. Reimer			
	Changes 1	MEDOC	3	RD	
	23.7.98	Dipl.-Ing. B. Weichel			
	Changes 2	MEDOC	2	cd	
	10.7.98	Dipl.-Ing. B. Weichel			
	Changes 3	MEDOC	1.1.1	CD	
	18.2.98	Dipl.-Ing. B. Weichel			
	Changes 4	MEDOC	1.1.1	Concluded	
	18.11.97	Dipl.-Ing. B. Weichel			
	Changes 5	MEDOC	11.0f	Concluded	
	05.11.97	Dipl.-Ing. B. Weichel			
	Changes 6	MEDOC	11.0b	Outdated	
	21.04.97	Dipl.-Ing. B. Weichel			
	Changes 7	MEDOC	11.0a	Outdated	
	09.10.96	Dipl.-Ing. B. Weichel			
	Changes 8	MEDOC	10.3	Complete	
	22.8.96	Dipl.-Ing. B. Weichel			
	Changes 9	MEDOC	10.2	Concluded	
	12.06.96	Dipl.-Ing. B. Weichel			
	Changes 10	MEDOC	10.1	Obselete	
	05.06.96	Dipl.-Ing. B. Weichel			
	Changes 11	MEDOC	10.0	Obselete	
	30.05.96	Dipl.-Ing. B. Weichel			
	Changes 12	MEDOC	9.1	Following meeting 13.3.96	

References

Standards

Designation:	**[ISO639::1988((E/F))]:** Code for the representation of names of languages	
URL:		
Relevant Position:	Part1	55

Designation:	**[ISO 8879]:** Information Processing - Text and Office Information Systems (Standard Generalized Markup Language)	
State:	standard	
Date:	1986	
URL:		
Relevant Position:	entire document	45

Designation:	**[ISO 11519-1]:** To be defined	
State:	DIS	
URL:		
Relevant Position:	Entire document	70

Designation:	**[ISO 11898]:** To be defined	
State:	DIS	
URL:		
Relevant Position:	Entire document	70

External Documents

Designation:	EMPTY title of external document	
URL:		
Relevant Position:		45

Designation:	Analysis of the CAN database, documents of the Bosch company,

Document number:		
State:		
Date:	17.04.1989	
URL:		
Relevant Position:	all	73

Designation:	CALS table spec	
URL:		
Relevant Position:	all	48

Designation:	CAN Controller Area Network, Principles, Protocols, Modules, Applications	
Date:	1994	
Publisher:	Carl Hanser Publishers Munich Vienna, Munich	
URL:		
Relevant Position:	Entire book	73

Designation:	CAN-Contoller Area Network, Principles and Practice	
Date:	1984	
Publisher:	Hüthing Publishers	
URL:		
Relevant Position:		73

Designation:	CAN-ER-Modell	
Date:	1988	
Publisher:	Robert Bosch GmbH	
URL:		
Relevant Position:	Hand-written graphics and SQL structure definitions	73

Designation:	Fundamental structures for software
State:	1.1.0
Date:	16.12.97
Publisher:	MSR AG-MEDOC

URL:		
Relevant Position:	Chapter Contents model software	30

Designation:	ISO Norm CAN High-Speed	
URL:		
Relevant Position:	all	25

Designation:	ISO Norm CAN Low-Speed	
URL:		
Relevant Position:	all	25

External FILES

Designation:	MOTRONIC wiring diagram	45, 47
URL:	motronic.asc	
Format:	concept-ascii-format	
Tools:	concept	
Version:	2.7	

Index

Technical Terms

Code

C

content-related 54

crpctmt.wmf 42

D

doc-related 54

E

ENTITY 42

extended 59, 65

M

mm crp.sgm 46

S

scale=”0.5” 42

SDATA 42

standard 59, 65

Control elements

O

OK 46

Material

P

plastic 46

W

wood 46

Organisations

Symbols

ASAP 28

ISO TC22 SC3 WG1 29

OSEK 28

M

MEDOC Networks 11

OTHER

Symbols

Application signal 30

CAN_LOW 22

Low 75

net-signal group 28

Network management signal 30

power- down message 71

power-down message 34

Standard-Format 32, 64

C

CALS table 48, 48, 48

CAN_HIGH 22

CAN_SHIELD 22

E

Extended-Format 32, 64

Extended-Format 32, 64

H

High 75

I

ISO 46

N

ncoi-2 36, 36

ncoi-3 36, 36

S

Shield 75

T

thing 47

Parameter

C

Cycle time 20

K

KFZW 46

L

Latency 20

Products

Symbols

CAN-controller 26

control unit 15, 15

MSRNET.DTD 27, 61

MSRSW.DTD 61

SGML Attributes

SGML Elements

Symbols

T

U

V

X

State

Symbols

A

I

M

P

R

S

X

Tools

Symbols

A

C

E

L

M

P

R

S

T

Variables

M

N

R

T

Configuration Parameters

Company (—company)
MEDOC

Language (—lang)
English

Treatment of content for Xrefs (—xrefcontent)
Xref classes are shown

Specifying 'See' for XRefs
'See' is to be inserted for xrefs

Treatment of filenames in graphics (—figname)
Filenames for graphics are shown

Treatment of width and height attributes of graphics (—figdimension)
Width and height of graphics is not considered

Titlepage Graphic (—graphic)
No title graphic specified

Logo Graphic (—head-logo)
msrreportlogo.eps

Fixtext File (—fixtext)
C:\Programme\Medoc\Metapage\mmapps\msrrep\lib\msrrep_ft.xml

Output of Local Administrative Data (—admindata)
Local administrative data is output

Filename
C:\Docs\MEDOC\MSRNET\docs\en\sp\msrnet-sp-en_V210.xml

MetaMorphosis-Version
3.2

Form Version
2.0 (MetaPage)

Date
11/02/2002 13:17:24