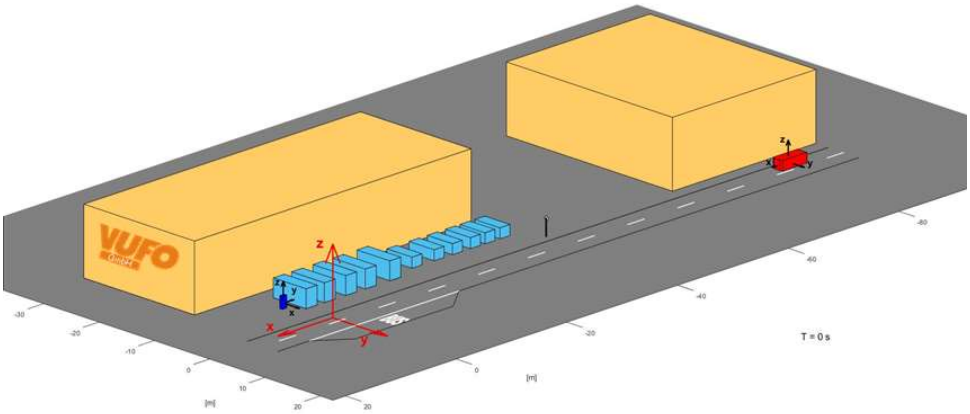
Pre-Crash-Matrix (PCM)

Format Specification v5.0

Effective 31.01.2019



Contact	VUFO (Institute for Traffic Accident Research at Dresden University of Technology)
Mail	pcm@vufo.de
Address	Semperstraße 2a 01069 Dresden - Germany

Table of Contents

Legal Framework

GENERAL

All trade names, trademarks and/or company logos referenced in this document belong to their respective companies.

VUFO GmbH (VUFO) is the owner of all intellectual property rights in this document including but not limited to, copyrights to this document.

The PCM format as well as this document will be improved by VUFO in the future. Changes to the PCM format can be made by VUFO at any time with no official notice. It is planned to incorporate changes in this document and publish new editions periodically. No part of this document may be reproduced, in whole or in part, in any way, including, but not limited to, photocopy, photograph, or other record, without the prior written permission of VUFO.

WARRANTY

The PCM format specifications are provided by VUFO “as is”, without any warranty, either express or implied, including (but not limited to) any warranties or conditions of title, non-infringement, merchantability or fitness for a particular purpose. All warranties are expressly and specifically disclaimed.

DISCLAIMER OF LIABILITY

VUFO is not responsible for any direct, indirect, incidental, special, exemplary, consequential damages (including without limitation of lost profits) arising from the use of this document, the provided database and example cases.

The document may contain inaccuracies or typographical errors.

PUBLIC LICENSE STATEMENT

The PCM format is property of VUFO GmbH (VUFO) and protected under the copyright law of the Federal Republic of Germany. Users should name VUFO as owner of the PCM format in publications that refer to the PCM.

VUFO grants all users a non-exclusive and non-transferable right to use the format as follows:

I. Storing data in a PCM database created according to the format specifications
II. Accessing data stored in a PCM database
III. Exchanging data in the PCM format with other stakeholders

Any attempt to sub-license, transfer or assign all or any part of the PCM format specification is prohibited.

Preface – What is PCM?

WHAT IS PCM?

PCM is the abbreviation for Pre-Crash-Matrix, and is a specified format to describe the phase of a traffic accident prior to the collision (the so-called pre-crash phase). It describes participants, its dynamics as well as the environment.

Initially the VUFO (Institute for Traffic Accident Research at Dresden University of Technology) developed the format in 2011 to provide pre-crash data for accidents investigated within the German In-Depth Accident Study (GIDAS). In January 2019, the format was defined in a more general way. Since then, it is applicable for any accident or scenario database and publicly accessible.

As it does not describe further data regarding the accident / scenario (e.g. accident description, accident type, injury severities, et al.), specific analyses should always refer to the linked accident / scenario database.

PCM STRUCTURE

Figure 1 shows the PCM structure including all tables in their hierarchic definitions. Attached to some environment tables there are property tables (containing meta data) and library tables which refer to country-specific definitions. The completion as well as the extension on other country-specific libraries is possible and one of the intended objectives for the future PCM development process.

CREATION AND SPECIFIC PCM DATABASE

A PCM case can be created by simulation methods (e.g. accident reconstruction) as well as data recording (e.g. from Event Data Recorders (EDR), Naturalistic Driving Study (NDS), Field Operational Test (FOT), …). The resulting database contains a set of accidents / scenarios in PCM format.

The standard file format is a Microsoft Access database (mdb). As there exists the 2GB limit on a 32-bit operating system an alternative will be discussed for a future format release.

The name of the resulting PCM database should always refer to the linked database, e.g. “GIDAS-PCM”.

EXAMPLE CASE

An example case is available for download on our website https://www.vufo.de/.

FURTHER DEVELOPEMENT

PCM is a “living format” that will evolve in cooperation with their users and partners. Send suggestions for format extensions, new or missing objects (OBJTYPE), mistakes in the specification and especially further libraries (e.g. with your countries specific characteristics) directly to VUFO via pcm@vufo.de. In order to provide uniform definitions for all users the extensions can then be included in the next release.

Description of tables and variables

The following chapters describe the variables in the respective tables as well as their unit and data type. In general, if any information is unknown use the value 99999.

The following abbreviations are used:

COS – Coordinate system (3-dimensional, cartesian)
COG – Center Of Gravity
GPS – Global Positioning System (according to WGS 84)

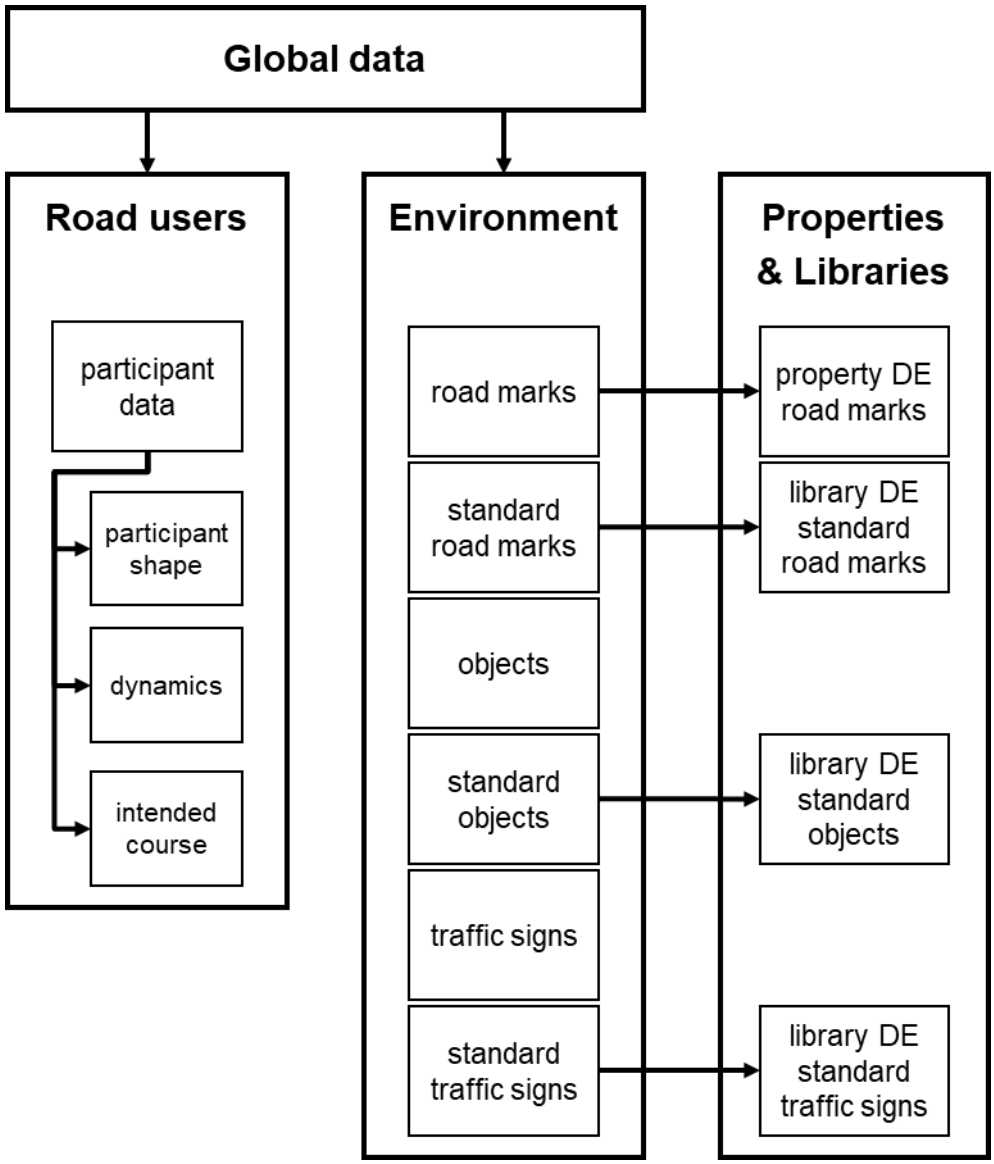


Figure 1: Hierarchical structure of PCM

Global data

Table: global_data

Description: The table global_data provides general information about the accident / scenario.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
DATETIME	Date and time expressed according to ISO 8601:2004 This represents the global zero for all TIME values. Unknown digits may be omitted or expressed by “9” Example: 2019-01-31T11:11:11.111+00:00	[]	Short text
PARTICIP	Number of involved participants	[]	Long int.
SOLVER	Used solver for vehicle dynamics 1 – IPG CarMaker 2 – DSD PC-Crash 3 – AnalyzerPro 4 – Virtual Crash 5 – IPG TruckMaker 6 – IPG MotorcycleMaker 7 – Mechanical Simulation™ CarSim® 8 – Mechanical Simulation™ TruckSim® 9 – Mechanical Simulation™ BikeSim® 10 – Tass International PreScan 11 – dSPACE Automotive Simulation Models (ASM) 12 – TESIS DYNA4 13 – Pro Impact 88888 – other	[]	Long int.
GPSLAT	GPS latitude of global COS according to WGS 84 (decimal, e.g. 51.034186)	[°]	Double
GPSLON	GPS longitude of global COS according to WGS 84 (decimal, e.g. 13.744801)	[°]	Double
GPSELE	Elevation of global COS according to WGS84 reference system (ellipsoidical height)	[m]	Double

Road users

Table: participant_data

Description: The table participant_data contains relevant variables to parametrise participants. This data can be used to model the geometry and further attributes. The variable PARTID is the participant identifier. Consider the table participant_shape to model the participant geometry in detail. The variable PARTTYPE describes the type of road user.

It is recommended to use a local reference COS according to ISO 8855 with the COG as origin.

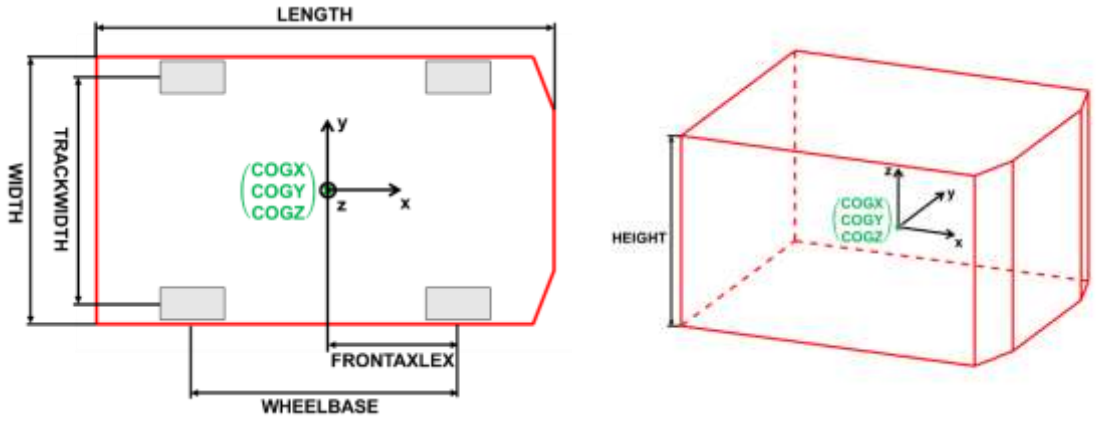


Figure 2: participant definitions

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
PARTID	Participant identifier per CASEID	[]	Long int.
PARTTYPE	Participant type: 0 – passenger car 1 – pedestrian 2 – motorbike 3 – bicycle 4 – truck 5 – bus 6 – tram/train 7 – trailer 8 – camper 9 – agricultural vehicle 10 – construction vehicle	[]	Long int.

	11 – emergency vehicle 12 – large animal 13 – small animal 88888 – other		
LENGTH	Length	[m]	Double
WIDTH	Width	[m]	Double
HEIGHT	Height	[m]	Double
TRACKWIDTH	Track width	[m]	Double
WHEELBASE	Wheelbase	[m]	Double
FRONTAXLEX	Position of front axle in x-direction in reference to participant local coordinate system (participant shape)	[m]	Double
WEIGHT	Weight	[kg]	Double
COGX	x-coordinate of COG in reference to participant local coordinate system (participant shape)	[m]	Double
COGY	y-coordinate of COG in reference to participant local coordinate system (participant shape)	[m]	Double
COGZ	z-coordinate of COG in reference to participant local coordinate system (participant shape)	[m]	Double
IXX	Moment of inertia around x-axis in reference to COG	[kgm²]	Double
IYY	Moment of inertia around y-axis in reference to COG	[kgm²]	Double
IZZ	Moment of inertia around z-axis in reference to COG	[kgm²]	Double

Table: participant_shape

Description: The table participant_shape defines the geometrical shape for each participant by surfaces. Each surface contour is defined by points. Each point is defined by its position $(x \quad y \quad z)^T$ in reference to the participants local COS.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
PARTID	Participant identifier per CASEID	[]	Long int.
SURFID	Surface identifier per PARTID	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
X	Local x-coordinate	[m]	Double
Y	Local y-coordinate	[m]	Double
Z	Local z-coordinate	[m]	Double

Examples for participant shapes:

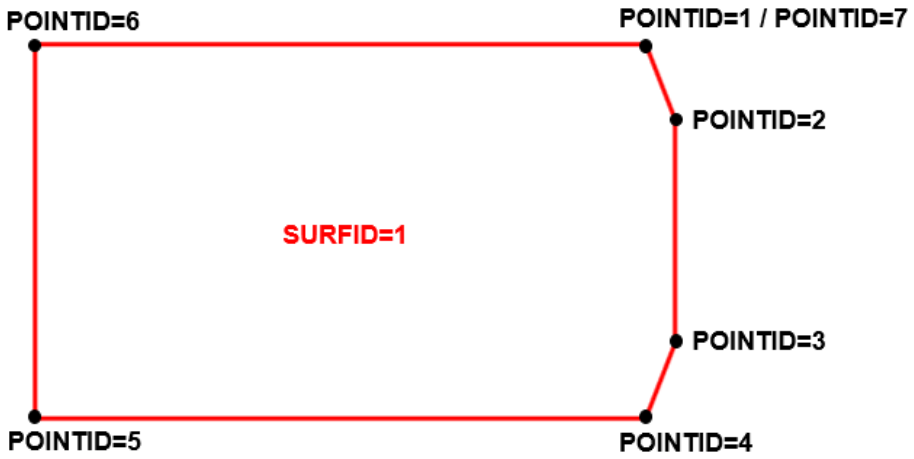


Figure 3: Example for 2D vehicle shape definition of a passenger car

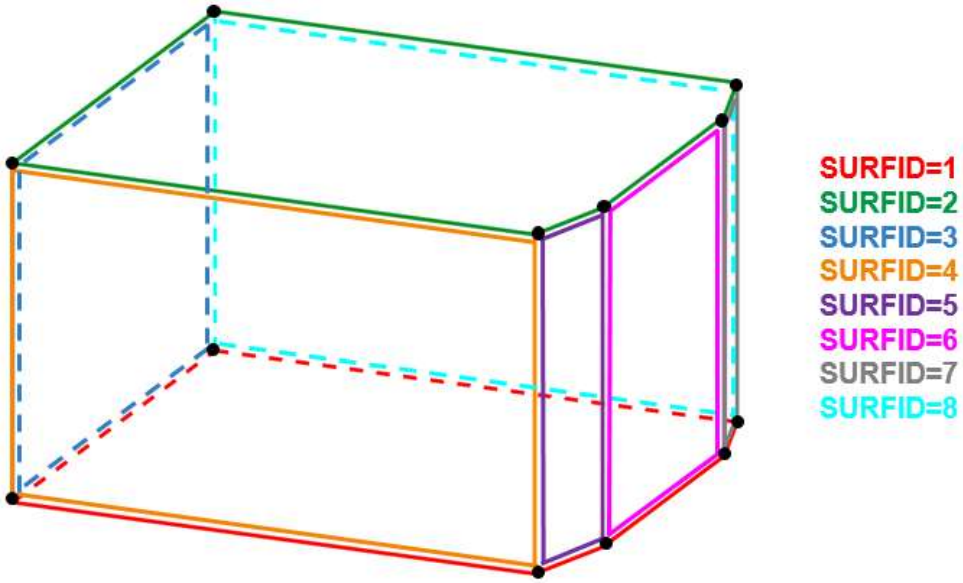


Figure 4: Example for 3D vehicle shape definition of a passenger car

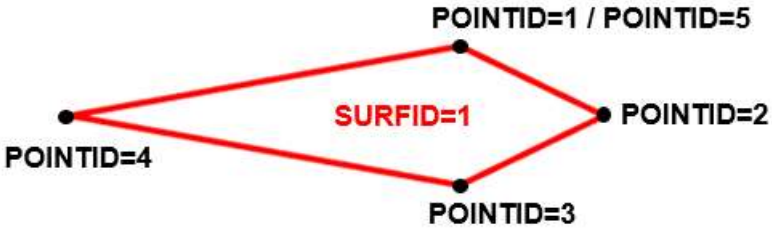


Figure 5: Example for 2D vehicle shape definition of a motorcycle/ bicycle

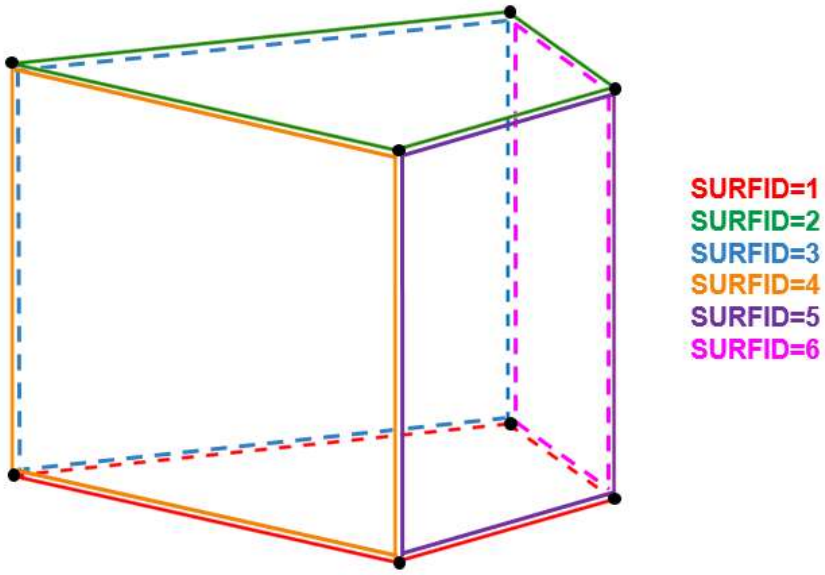


Figure 6: Example for 3D vehicle shape definition of a motorcycle/ bicycle

Figure 7: Example for 2D pedestrian shape definition

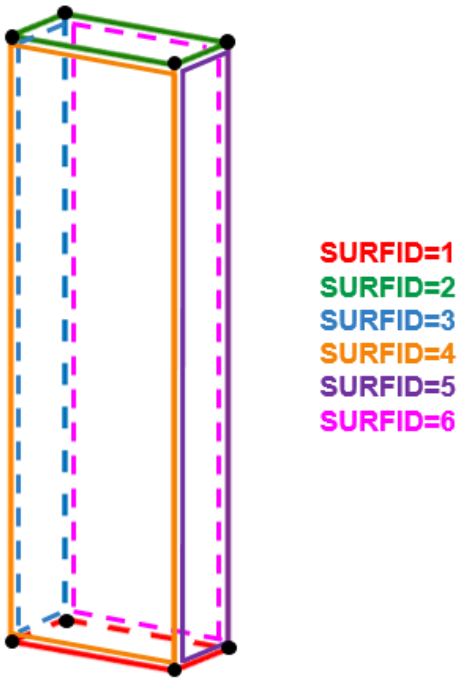


Figure 8: Example for 3D pedestrian shape definition

Table: dynamics

Description: The table dynamics defines the global position of participants $(x \quad y \quad z \quad \Phi \quad \Theta \quad \Psi)^T$ according to the global COS as well as velocity and acceleration of the participants COG according to the local COS at each time step of the simulation (see Figure 9).

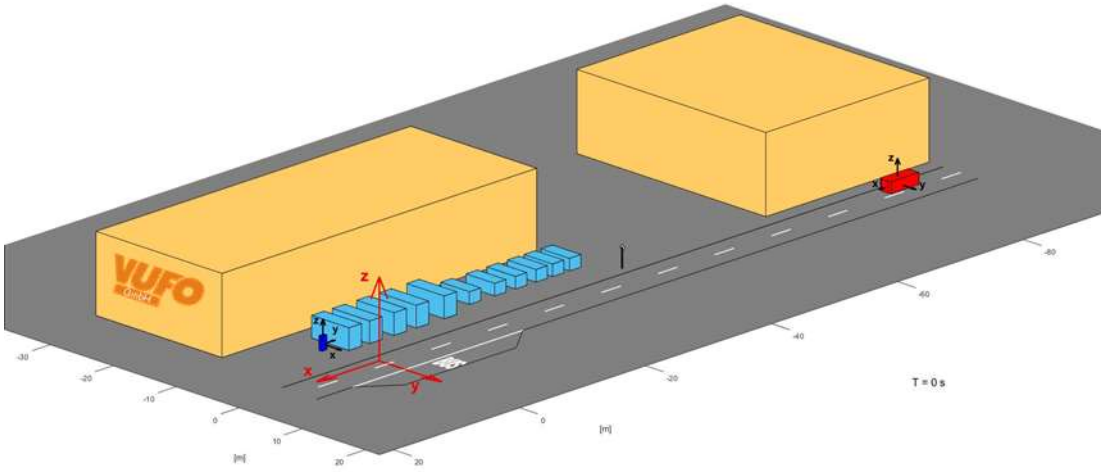


Figure 9: Global (red) and local (black) coordinate systems

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
PARTID	Participant identifier per CASEID	[]	Long int.
VARIATIONID	Variation identifier per PARTID 0 – original accident / scenario ≥ 1 – simulation variation, like additional braking or other	[]	Long int.
TIME	Time step	[s]	Double
POSX	Global x-position of COG	[m]	Double
POSY	Global y-position of COG	[m]	Double
POSZ	Global z-position of COG	[m]	Double
POSPHI	Global roll angle Φ of COG (cardan angles)	[rad]	Double
POSTHETA	Global pitch angle Θ of COG (cardan angles)	[rad]	Double
POSPSI	Global yaw angle Ψ of COG (cardan angles)	[rad]	Double
VX	Velocity of COG in local x-direction	[m/s]	Double
VY	Velocity of COG in local y-direction	[m/s]	Double

VZ	Velocity of COG in local z-direction	[m/s]	Double
AX	Acceleration of COG in local x-direction	[m/s²]	Double
AY	Acceleration of COG in local y-direction	[m/s²]	Double
AZ	Acceleration of COG in local z-direction	[m/s²]	Double
MUE	Coefficient of friction	[]	Double
REC	Reconstruction / data recording availability for the given time step 1 – Values based on reconstruction / data recording 0 – Values based on extension of reconstruction / data recording	[]	Long int.

Table: intended_course

Description: The table intended_course defines the course the participant initially intended to follow. The POINTID determines the points with their global position $(x \quad y \quad z)^T$.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
PARTID	Participant identifier per CASEID	[]	Long int.
VARIATIONID	Variation identifier per PARTID 0 – original accident / scenario ≥ 1 – variation	[]	Long int.
POINTID	Point identifier per VARIATIONID	[]	Long int.
POSX	Global x-coordinate	[m]	Double
POSY	Global y-coordinate	[m]	Double
POSZ	Global z-coordinate	[m]	Double

Environment

Several tables and their link to a respective library or property describe the accident / scenario's environment.

Table: road_marks

Description: The table road_marks is part of the environment description. It defines the relevant boundaries and markings of the road as objects, each with an OBJID as identifier and consisting of points with their position $(x \quad y \quad z)^T$ in reference to global COS. These are no 3D objects, but they can be positioned in 3D.

The variable OBJTYPE defines the kind of road boundary or marking. The list of possible OBJTYPEs and their definition is given in the table property_DE_road_marks (example for Germany).

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
POINTID	Point identifier per OBJID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
X	Global x-coordinate	[m]	Double
Y	Global y-coordinate	[m]	Double
Z	Global z-coordinate	[m]	Double

Table: standard_road_marks

Description: The table standard_road_marks is part of the environment description. It defines frequently existing road markings (e.g. turning arrow, bicycle path, “BUS” symbol) each with an OBJID as identifier. The advantage of standard road marks is that they are defined only once and can be used for multiple cases, which saves storage space in the database.

The variable OBJTYPE defines the kind of road mark. The list of possible OBJTYPEs and their definition is given in the table library_DE_standard_road_marks (example for Germany).

These are no 3D objects, but can be positioned and scaled in 3D to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
REFX	Global x-coordinate of reference point	[m]	Double
REFY	Global y-coordinate of reference point	[m]	Double
REFZ	Global z-coordinate of reference point	[m]	Double
REFROTX	Rotation angle around x-axis at reference point (cardan angles)	[rad]	Double
REFROTY	Rotation angle around y-axis at reference point (cardan angles)	[rad]	Double
REFROTZ	Rotation angle around z-axis at reference point (cardan angles)	[rad]	Double
SCALEX	Scaling factor in x-direction at reference point	[]	Double
SCALEY	Scaling factor in y-direction at reference point	[]	Double
SCALEZ	Scaling factor in z-direction at reference point	[]	Double

Table: objects

Description: The table objects is part of the environment description. It defines the geometrical shape of relevant stationary objects by surfaces. Each surface contour is defined by points. Each point is defined by its position $(x \quad y \quad z)^T$ in reference to global COS.

The variable OBJTYPE defines the kind of object. The list of possible OBJTYPEs is given in the table below.

Comments: In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
SURFID	Surface identifier per OBJID	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
X	Global x-coordinate	[m]	Double
Y	Global y-coordinate	[m]	Double
Z	Global z-coordinate	[m]	Double

List of OBJTYPEs:

OBJTYPE	Description
520	Traffic barrier (steel)
521	Traffic barrier (concrete)
522	Traffic barrier (not specified)
523	Traffic pole for guidance, warning and barriers
524	House wall, bridge pier
525	Game fence, wooden fence
526	Bush, hedge
527	Railing
528	Group of trees
529	Railway gate
530	Traffic cone

531	Bus Stop
532	Taxi rank
533	Speed bump
534	Bridge
535	Tunnel
536	Garbage bin / waste basket
537	Earth wall
538	Embankment
549	Other areal objects
599	Objects – not specified

Table: standard_objects

Description: The table standard_objects is part of the environment description. It defines frequently existing stationary objects (e.g. trees, parking vehicles), each with an OBJID as identifier. The advantage of standard objects is that they are defined only once and can be used for multiple cases, which saves storage space in the database.

The variable OBJTYPE defines the kind of object. The list of possible OBJTYPEs and their definition is given in the table library_DE_standard_objects (example for Germany).

Each object can be positioned and scaled in 3D to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
REFX	Global x-Coordinate of reference point	[m]	Double
REFY	Global y-Coordinate of reference point	[m]	Double
REFZ	Global z-Coordinate of reference point	[m]	Double
REFROTX	Global rotation angle around x-axis at reference point (cardan angles)	[rad]	Double
REFROTY	Global rotation angle around y-axis at reference point (cardan angles)	[rad]	Double
REFROTZ	Global rotation angle around z-axis at reference point (cardan angles)	[rad]	Double
SCALEX	Scaling factor in x-direction at reference point	[]	Double
SCALEY	Scaling factor in y-direction at reference point	[]	Double
SCALEZ	Scaling factor in z-direction at reference point	[]	Double

Table: traffic_signs

Description: The table traffic_signs is part of the environment description. It defines the geometrical shape of arbitrary traffic signs by surfaces (be aware of standard traffic signs). Each surface contour is defined by points. Each point is defined by its position $(x \quad y \quad z)^T$ in reference to global COS.

The variable OBJTYPE defines the kind of object. There is currently no list of possible OBJTYPEs beyond the library_DE_standard_traffic_signs.

Comments: In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
SURFID	Surface identifier per OBJID	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
X	Global x-coordinate	[m]	Double
Y	Global y-coordinate	[m]	Double
Z	Global z-coordinate	[m]	Double

Table: standard_traffic_signs

Description: The table standard_traffic_signs is part of the environment description. It defines frequently existing traffic signs, each with an OBJID as identifier. The advantage of standard traffic signs is that they are defined only once and can be used for multiple cases, which saves storage space in the database.

The variable OBJTYPE defines the kind of object. The list of possible OBJTYPEs and their definition is given in the table library_DE_standard_traffic_signs (example for Germany).

Each object can be positioned and scaled in 3D to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

Variable	Description	Unit	Type
CASEID	Unique case identifier	[]	Short text
OBJID	Object identifier per CASEID	[]	Long int.
OBJTYPE	Type of object	[]	Long int.
REFX	Global x-coordinate of reference point	[m]	Double
REFY	Global y-coordinate of reference point	[m]	Double
REFZ	Global z-coordinate of reference point	[m]	Double
REFROTX	Rotation angle around x-axis at reference point (cardan angles)	[rad]	Double
REFROTY	Rotation angle around y-axis at reference point (cardan angles)	[rad]	Double
REFROTZ	Rotation angle around z-axis at reference point (cardan angles)	[rad]	Double
SCALEX	Scaling factor in x-direction at reference point	[]	Double
SCALEY	Scaling factor in y-direction at reference point	[]	Double
SCALEZ	Scaling factor in z-direction at reference point	[]	Double

Properties and Libraries

Table: property_DE_road_marks

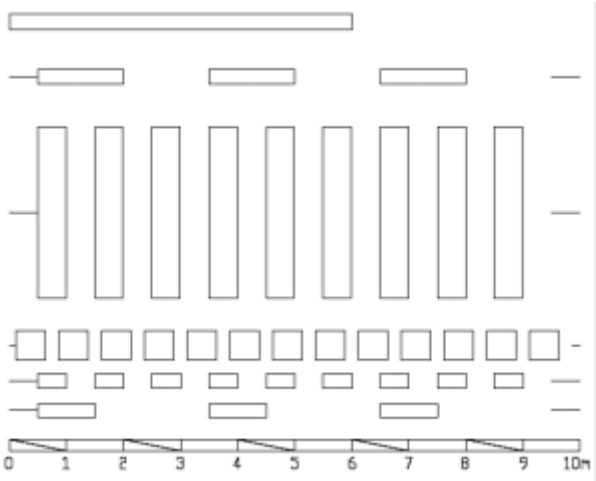
Description: The table property_DE_road_marks is referred to by table road_marks through OBJTYPE. It defines the width, length and gap of a line and can be used to visualize the various road boundaries and markings.

Comments: Countries-specific road boundaries and markings may be required. Additional properties can be added for this purpose. The format of the library must remain the same. The name of the library should be changed according to the ISO3166 Alpha-2 code.

Example:
Germany: "property_DE_road_marks"
China: "property_CN_road_marks"
…

In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
OBJTYPE	Type of object	[]	Long int.
DESCRIPTION	Description object	[]	Short text
WIDTH	Width of the line	[m]	Double
LENGTH	Length of line	[m]	Double
GAP	Length of gap	[m]	Double



242	Motorway – interrupted road edge
223	Extra urban – interrupted road edge
111	lateral, interrupted for pedestrians (crosswalk)
109	Lateral, interrupted for turning events
106	Interrupted for cyclists
105	Interrupted for cyclists

Figure 10: Examples for road mark types

OBJTYPE	DESCRIPTION	WIDTH	LENGTH	GAP
General				
101	Roadside (e.g. curb)	-	-	-
102	Continuous to indicate stopping restriction (e.g. zigzag pattern)	0.12	-	-
103	Continuous for barred areas	0.5	-	-
104	Continuous for cyclists	0.25	-	-
105	Interrupted for cyclists	0.25	1	2
106	Interrupted for cyclists	0.25	0.5	0.5
107	Lateral, interrupted for cyclists	0.25	0.5	0.2
108	Lateral, continuous for stopping traffic (e.g. at stop signs, traffic lights, railway crossing signs)	0.5	-	-
109	Lateral, interrupted for turning events	0.5	0.5	0.25
110	Lateral, interrupted for pedestrians (e.g. at traffic lights)	0.12	0.5	0.2
111	Lateral, interrupted for pedestrians (crosswalk)	3	0.5	0.5
112	Railway tracks	-	-	-
114	Parking lot	-	-	-
115	Interrupted for cyclists	0.12	1	1
116	Lane Marking without white marking (e.g. with stones for parking spaces or bus stops)	-	-	-
196	Former marks continuous	-	-	-
197	Former marks of the road edge	-	-	-
198	Former marks interrupted short	-	-	-
199	Former marks interrupted long	-	-	-
Urban				
201	Continuous road edge	0.25	-	-
202	Interrupted road edge (no junction)	0.25	1	0.5
203	Interrupted road edge (junction area)	0.25	3	3
204	Interrupted road edge (close to junction area)	0.25	1.5	1.5
205	Continuous lane marking	0.12	-	-
206	Guiding marks (no junction)	0.12	3	6

207	Guiding marks (in general)	0.12	3	1.5
208	Guiding marks (junction area)	0.12	3	3
209	Interrupted road edge (junction area)	0.12	1.5	1.5
Extra urban				
221	Continuous road edge	0.25	-	-
222	Interrupted road edge (junction area)	0.25	3	3
223	Interrupted road edge (close to junction area)	0.25	1.5	1.5
224	Continuous lane marking	0.12	-	-
225	Guiding marks (no junction)	0.12	4	8
226	Guiding marks (in general)	0.12	4	2
227	Guiding marks (junction area)	0.12	3	3
Motorway				
241	Continuous road edge	0.3	-	-
242	Interrupted road edge (junction area)	0.3	6	6
243	Continuous lane marking	0.15	-	-
244	Guiding marks (no junction)	0.15	6	12
245	Guiding marks (in general)	0.15	6	3
246	Guiding marks (connecting ramp, adjunct lane)	0.15	6	6
Construction site				
261	Continuous road edge	0.25	-	-
262	Interrupted road edge (close to junction) – urban	0.25	1.5	1.5
263	Interrupted road edge (junction area) – extra urban	0.25	3	3
264	Interrupted road edge (junction area) – motorway	0.3	6	6
265	Guiding marks (no junction) – urban	0.12	3	6
266	Guiding marks (junction area) – urban	0.12	3	3
267	Guiding marks (no junction) – extra urban	0.12	4	8
268	Guiding marks (no junction) – motorway	0.15	6	12
269	Interrupted marks for bicycle / pedestrian passage	0.25	0.5	0.2
270	Lateral, continuous for stopping traffic (e.g. at stop signs, traffic lights, railway crossing signs)	0.5	-	-

271	not specified	-	-	-
Roadside				
401	Pavement	-	-	-
402	Bicycle path	-	-	-
405	Combined bicycle pedestrian path	-	-	-

Table: library_DE_standard_road_marks

Description: The table library_DE_standard_road_marks defines the general shape of standard road marks (e.g. turning arrow, bicycle path, “BUS” symbol) in Germany and is referred to by standard_road_marks through OBJTYPE.

The variable OBJTYPE defines the kind of road mark. The list of possible OBJTYPEs is given in the table below (for Germany). Each OBJTYPE consists of one or several surfaces, each consisting of points with their position $(x \quad y \quad z)^T$ in reference to local COS (always SURFID = 1, POINTID = 1).

These are no 3D objects, but can be positioned and scaled in 3D to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

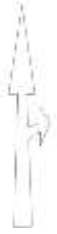
Comments: Countries-specific standard road marks may be required. Additional libraries can be added for this purpose. The format of the library must remain the same. The name of the library should be changed according to the ISO3166 Alpha-2 code,

Example:
Germany: "library_DE_standard_road_marks"
China: "library_CN_standard_road_marks"
…

In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
OBJTYPE	Type of object	[]	Long int.
SURFID	Surface identifier per OBJTYPE	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
X	Local x-coordinate	[m]	Double
Y	Local y-coordinate	[m]	Double
Z	Local z-coordinate	[m]	Double

List of OBJTYPE:

OBJTYPE	Description	Pictogram
301	Arrow to turn left	
302	Arrow to turn right	
303	Arrow to turn left and right	
304	Arrow to go straight	
305	Arrow to go straight and to turn left and right	
306	Arrow to go straight and turn left	
307	Arrow to go straight and turn right	

308	Preliminary arrow to turn left	
309	Preliminary arrow to turn right	
310	Pictogram velocity 20	
311	Pictogram velocity 30	
312	Pictogram velocity 50	
313	Pictogram bikeway 1	
314	Pictogram bikeway 2	
315	Pictogram wheelchair	
316	Pictogram STOP	

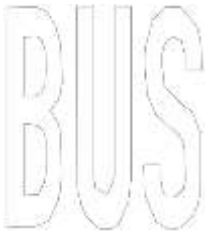
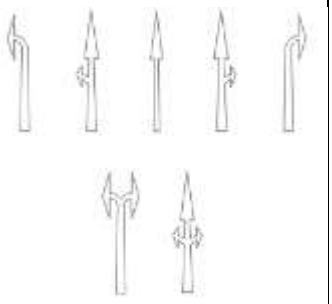
317	Pictogram BUS 1	
318	Pictogram BUS 2	
319	Pictogram BUS 3	
320	Bicycle arrow to turn left	
321	Bicycle arrow to turn right	
322	Bicycle arrow to turn left and right	
323	Bicycle arrow to go straight	
324	Bicycle arrow to go straight and to turn left and right	
325	Bicycle arrow to go straight and turn left	
326	Bicycle arrow to go straight and turn right	
327	Arrow to turn around	

Table: library_DE_standard_objects

Description: The table library_DE_standard_objects defines the general shape of frequently existing stationary objects in Germany (e.g. trees, parking vehicles) and is referred to by standard_objects through OBJTYPE.

The variable OBJTYPE defines the kind of road mark. The list of possible OBJTYPEs is given in the table below (for Germany). Each OBJTYPE consists of one or several surfaces, each consisting of points with their position $(x \quad y \quad z)^T$ in reference to local COS (always SURFID = 1, POINTID = 1).

They can be positioned and scaled in 3D in reference to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

Comments: Countries-specific standard road marks may be required. Additional libraries can be added for this purpose. The format of the library must remain the same. The name of the library should be changed according to the ISO3166 Alpha-2 code,

Example:
Germany: "library_DE_standard_objects"
China: "library_CN_standard_objects"
…

In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
OBJTYPE	Type of object	[]	Long int.
SURFID	Surface identifier per OBJTYPE	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
X	Local x-coordinate	[m]	Double
Y	Local y-coordinate	[m]	Double
Z	Local z-coordinate	[m]	Double

List of OBJTYPE:

OBJTYPE	Description
501	Tree trunk (w/o crown)
502	Pole of wood, concrete or steel
519	Other round objects
550	Standing motorcycle
551	Standing passenger car - small
552	Standing passenger car - medium
553	Standing passenger car - large
554	Standing transporter
555	Standing trailer
556	Standing bus
557	Standing truck
558	Standing truck trailer
559	Standing tram Dresden
560	Standing tram H
561	Standing tram Har
562	Standing bicycle

will be available soon …

Table: library_DE_standard_traffic_signs

Description: The table library_DE_standard_traffic_signs defines the general shape of frequently existing traffic signs in Germany and is referred to by standard_traffic_signs through OBJTYPE.

The variable OBJTYPE defines the kind of road mark. The list of possible OBJTYPEs is given in the table below (for Germany). Each OBJTYPE consists of one or several surfaces, each consisting of points with their position $(x \quad y \quad z)^T$ in reference to local COS (always SURFID = 1, POINTID = 1).

They can be positioned and scaled in 3D in reference to global COS through its reference point and the scaling factors. See also Appendix A: Advices for positioning objects.

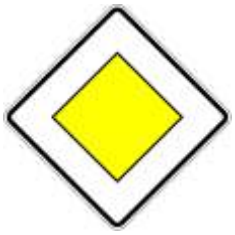
Comments: Countries-specific standard road marks may be required. Additional libraries can be added for this purpose. The format of the library must remain the same. The name of the library should be changed according to the ISO3166 Alpha-2 code,

Example:
Germany: "library_DE_standard_traffic_signs"
China: "library_CN_standard_traffic_signs"
…

In order to provide a uniform table for all users, it is recommended to forward extensions to pcm@vufo.de. The extensions can then be included in a new release.

Variable	Description	Unit	Type
OBJTYPE	Type of object	[]	Long int.
SURFID	Surface identifier per OBJTYPE	[]	Long int.
POINTID	Point identifier per SURFID	[]	Long int.
X	Local x-coordinate	[m]	Double
Y	Local y-coordinate	[m]	Double
Z	Local z-coordinate	[m]	Double

List of OBJTYPE:

OBJTYPE	Description	Pictogram
306	Priority road	

Appendix A: Advices for positioning objects

SCALING

- Take x-, y-, and z-coordinates from library for concerning OBJID (e.g. library_DE_standard_objects) and multiply by scaling factor 'SCALEX', 'SCALEY', 'SCALEZ':

$$x_{s_{ijk}} = SCALEX_i \cdot x_{ijk} ,$$

$$y_{s_{ijk}} = SCALEY_i \cdot y_{ijk} ,$$

$$z_{s_{ijk}} = SCALEY_i \cdot z_{ijk} ,$$

for all $i \in OBJID,\ j \in SURFNO,\ k \in POINTNO$

ROTATION

- Define matrix R_x for rotation around the x-axis by 'REFROTX':

$$R_x(\alpha_i) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha_i & -\sin\alpha_i \\ 0 & \sin\alpha_i & \cos\alpha_i \end{pmatrix} \quad \text{, where } \alpha_i = \mathrm{REFROTX}_i$$

- Define matrix R_y for rotation around the y-axis by 'REFROTY':

$$R_y(\beta_i) = \begin{pmatrix} \cos\beta_i & 0 & \sin\beta_i \\ 0 & 1 & 0 \\ -\sin\beta_i & 0 & \cos\beta_i \end{pmatrix} \quad \text{, where } \beta_i = \mathrm{REFROTY}_i$$

- Define matrix R_z for rotation around the z-axis by 'REFROTZ':

$$R_z(\gamma_i) = \begin{pmatrix} \cos\gamma_i & -\sin\gamma_i & 0 \\ \sin\gamma_i & \cos\gamma_i & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad \text{, where } \gamma_i = \mathrm{REFROTZ}_i$$

The rotation is composed of the rotation matrices multiplied by the x-, y-, z-coordinates.

$$\left(x_{r_{ijk}}, y_{r_{ijk}}, z_{r_{ijk}}\right)^T = R_z(\gamma_i) \cdot R_y(\beta_i) \cdot R_x(\alpha_i) \cdot \left(x_{s_{ijk}}, y_{s_{ijk}}, z_{s_{ijk}}\right)^T$$

for all $i \in OBJID,\ j \in SURFNO,\ k \in POINTNO$

TRANSLATION

- Translate x-, y-, z-coordinates by vector ('REFX','REFY','REFZ'):

$$\left(x_{t_{ijk}}, y_{t_{ijk}}, z_{t_{ijk}}\right) = \left(x_{r_{ijk}}, y_{r_{ijk}}, z_{r_{ijk}}\right) + \left(x_{ref_i}, y_{ref_i}, z_{ref_i}\right) ,$$

for all $i \in OBJID,\ j \in SURFNO,\ k \in POINTNO$