

# MonkeyProof – Data Dictionary Manager 2.0

... understand, manage and trust your data.

## Introduction



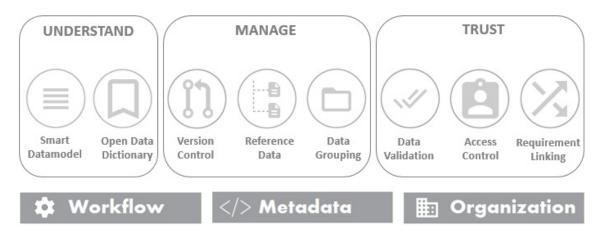
The file-based Data Dictionary Manager (DDM) is a central data store hosting the configurations of your software projects. DDM carries your data through the development cycle, from model design to software development and deployment to your targets. Its main objectives are to let you understand, manage and trust your data.

C-\MPS\pMPS_Products\DDM\monkeyproo	f_ddm\trunk\Imple	ementation	n\sampleDictionary2.dd - DDM	-		in space	-	-			_			and an	- 0 - X
Eile Edit View Insert Modify Version	n <u>C</u> ontrol Layo	ut <u>H</u> elp	,												
i 🗈 🗈 - 📲 🔍 🧉														Properties	5 P
sampleDictionary1 🔀 sampleDictionary2	3 sampleDiction	ary3 🔀												CPURAM_Cust: CPU_SENSING_MODE	
CIBBBB P+-			-   🚯 - 🗃 -   🏢 🛧 - 🕹 -											Description	
			1 1					-			-	-		001: Rotary XY (2-poles magnet) 010: Rotary YZ (2-poles magnet)	^
	Z <sup>y</sup> Type	2	Source		Addr	R	BitCount 6	BitOffset	Default		⊮ <sup>¥</sup> Spec 6	of fpDouble	🖉 fpinteger 🖉	011: Rotary XZ (2-poles magnet)	
Data Items     CPU SENSING MODE	CPURAM Cu		C:\MPS\pMPS_Products\DDM\	C	0x0002	2 8			0x0	_				100: Linear w/stray field robustness	E
Name	CPURAM_CU		C:\MPS\pMPS_Products\DDM\ C:\MPS\pMPS Products\DDM\		0x0002			2	0x0					101: Linear Hall Zx 110: Linear stroke (in BAB: with extended range)	-
CPU DSP NB CONV	CPURAM Cu		C:\MPS\pMPS_Products\DDM\		0x1010			3	0x2						•
CPU DSP REVPOL	CPURAM Cu		C:\MPS\pMPS Products\DDM\		0x0001			5	0x0					Requirements	
CPU DSP FILTER	CPURAM Cu		C:\MPS\pMPS Products\DDM\		0x1010	) 2		6	0x0						+-
CPU_T_ADC_SEQ	CPURAM_Tir		C:\MPS\pMPS_Products\DDM\			C 2	1	0	704	2	SYS	10	40		
CPU_AFE_T_SETTLING_0	CPURAM_Tir	ning	C:\MPS\pMPS_Products\DDM\			D 8	5	0	5	1	HSI	4	8	665	
CPU_AFE_T_SETTLING_1	CPURAM_Tir		C:\MPS\pMPS_Products\DDM\					0	1	3	HSI	20	160		
CPU_AFE_T_SETTLING_0_cop			C:\MPS\pMPS_Products\DDM\					0	5	3	HSI	16	128		
CPU_T_ADC_SEQ_copy	CPURAM_Tir	ning	C:\MPS\pMPS_Products\DDM\	m CRITICAL_TIMING   MPS	0x10B	с з		0	704	2	SYS	12	48		
Referenced Dictionaries     testDisplayDoubleLogicalArra			testDisplayDoubleLogicalArray.											Labels	
testbisplayboubleLogicalArra	iy		testolsplayboublecogicalArray.	10										CRITICAL_TIMING	- + -
														Customer	
														DSP	
														USP	
														Comments	
														Value 7 to be confirmed	
														value / to be commed	
														Checks	Run
														% Union check = all(eg(#Addr#, #children.Addr#))	
														<ul> <li>Adresses of children should be the same as the p</li> </ul>	parent's
Eind: CPU		h Case	9 matches											= all(eg(#BitCount#, #children.BitCount#))	
														% BitCount should be the same	
+											Ite	ems: 11 of 11	Columns: 12 of 1	2	

# **Key features**

- Standalone, cross-platform (Windows, Linux)
- File based dictionary, XML format
- Migration support services available
- Works with multiple dictionaries in parallel
- Supports inclusion of reference dictionaries
- Integrated SVN and Git version control
- Extensive dictionary preload validation
- Customizable and reusable dictionary view

- Export dictionaries to any text format
- Create Word, Excel, PDF reports
- Supports command-line operation
- Flexible data model, highly customizable dictionaries
- Role based access control
- Supports fixed-point data types
- Supports graphical inspection of array values
- Allows linking of data items to your requirements



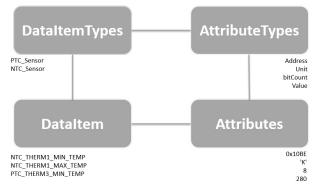
Data Dictionary Manager: Core Objectives, Key Capabilities and Business Environment



### **Smart Datamodel**

The datamodel lets you tailor your custom data definitions and facilitates migration of your existing (legacy) dictionaries.

You can separate configurations from your blueprints, enabling reuse across models and software projects. The user interface allows you to graphically inspect array data and to create personal and reusable views.



#### Figure 1: DDM Data Model overview

To govern consistency, checks can be implemented at dictionary and data item level. When required, you can create dependencies between data item attributes. MATLAB/Octave syntax formulas and regular expressions are supported as value definitions. Fixed-point datatypes are supported, as are merge or union data structures.



Figure 2: Sample checks, attribute linking & using formulas

# **Open Data Dictionary**

Easy workflow integration is ensured by the DDM API and the open nature of the XML based dictionaries, in conjunction with template-based export facilities. In addition, migration services are available and you can create PDF (and other) reports of your dictionaries.

# **Version Control**

Integrated Git and Subversion facilities let you safely track, commit and check your work in two of the most widely used version control tools, from within DDM.

#### **Reference Data**

DDM lets you include reference data into your dictionaries. Referenced dictionaries are datasets that (pre)exist in your organization, but are inherently important to your project. You can open referenced dictionaries alongside your project, and work on them simultaneously.

#### **Data Grouping**

In DDM, you can label and group your data. Labeling is for example used to define a set of parameters that are linked to a specific setting or feature in your software. It helps you organizing your data items and to ensure proper overview of dictionary contents.

- NTC_THERM1_MAX_TEMP_NTC_Sensor NTC_Sensing 0x10BD K 300	
	8
DTC TUEDAD MUNITEME DTC Comments DTC Comments 0 1010	8
PTC_THERM3_MIN_TEMP PTC_Sensor PTC_Sensing 0x1010 K 0	8
PTC_THERM3_MAX_TEMP_PTC_Sensor PTC_Sensing 0x1011 K 280	8
O     Referenced Dictionaries	

Figure 3: Organizing / labeling data items

# **Data Validation**

DDM features extensive validation functionality: any user modification is checked. Loading dictionaries automatically invokes Schema Definition checks (XSD) followed by more rigorous content validations. In addition: DDM will detect and validate modifications of the dictionary XML files. On top of that you can your own checks.

#### **Access Control**

By organizing data items into groups, you can set ownership and permissions. This is of particular importance for collaborative development work where teams and/or individuals need to be assigned read-only or read/write access for development, review, or useonly purposes.

# **Requirement Linking**

You can link your requirements to individual data items, keeping specifications traceable.

# System Requirements

DDM is a standalone application, for Windows and Linux.

MonkeyProof Solutions BV PO Box 4611 – 4803 EP Breda – The Netherlands info@monkeyproofsolutions.nl www.monkeyproofsolutions.nl