Automating Testing with Telelogic Doors @ NXP

Dr. Bernd GRAHLMANN – (www.grahlmann.net / bernd@grahlmann.net)
Requirements Engineering / Management / Development & Telelogic DOORS Expert / Trainer / Consultant

Joint work with MIFARE Plus Team (NXP Semiconductors, Caen & Gratkorn & Hamburg)
Business Line Identification, Business Unit Automotive and Identification

Telelogic User Conference 2008 – France (November 26, 2008)

Abstract:

This presentation gives an overview of the introduction of DOORS in the business line 'Identification' at NXP. Important aspects of the main choices (such as scope diagrams for system decomposition, traceability schemas, a scope based and re-use driven DOORS database structure, the usage of requirements management cockpits and sophisticated DOORS module templates) are explained.

An approach which automates system testing within the flagship project (the 'MIFARE Plus' proximity smart card system) is presented in more detail. Engineers enter test procedures for test cases in pseudo code in a DOORS attribute within a test specification module; a DXL attribute automatically generates (skeletons of) test benches in C#; when run those test benches produce test results in such a form that they can be 'imported' easily in DOORS; a powerful visualization of the traceability allows (among others) a sophisticated analysis of the test results from the requirements.

Agenda

- NXP Semiconductors, Business Line Identification, MIFARE Plus Project
- Aspects of the main choices of the DOORS implementation:
  - scope diagrams for system decomposition
  - traceability schemas
  - scope based and re-use driven DOORS database structure
  - usage of requirements management cockpits
  - sophisticated DOORS module templates
  - RM & DOORS architects dedicated to NXP projects
- Automating system level testing in MIFARE Plus:
  - Problem
  - Test Cases in DOORS with ‘pseudo code’ for test procedures
  - DXL attribute generating (skeletons for) C# test benches
  - Test benches runs resulting in ‘importable’ CSV files
  - Additional option to feed back test bench updates into DOORS
- Summary
- Questions / Discussion

NXP Semiconductors

- Established in 2006 (formerly a division of Philips)
- Builds on a heritage of 50+ years of experience in semiconductors
- Headquarters: Eindhoven, the Netherlands
- Provides engineers and designers with semiconductors, system solutions and software that deliver better sensory experiences
- Net sales of $6.32 billion in 2007 *)
- Key focus areas:
  - Home
  - Automotive & Identification
  - Multimarket Semiconductors
- Owner of NXP Software: a fully independent software solutions company

*) These figures include the Mobile & Personal business which is largely part of the ST-NXP Wireless JV since August 2, 2008

Identification

Lead in focus areas

- #1 in **RFID tagging** solutions
  - Supplying >50% of HF tagging market
  - Breakthrough in UHF Cost effectiveness
- #1 in **eGovernment** solutions
  - Supplying 80% of ePassports projects worldwide
- #1 in contactless **banking** solutions
  - Supplied > 500 million banking cards in 35 countries
- #1 in **NFC** solutions
  - Creator of NFC technology together with Sony
  - NXP products used in about 150 NFC trials worldwide
- #1 in **Public transportation**
  - MIFARE is used in >70% transport infrastructure

MIFARE Plus

Products	**Mainstream contactless smartcard IC**
End User	**Citizens (Consumers, public servants & employees)**
Supply Chain	**Card and Reader Makers; Solution Developers;** and **End User Solution Providers**
Application Needs	**increased security** with **fast and easy** solution development
Values	**seamless upgrade** to existing infrastructure and end user services with **low cost upgrade** procedures
Alternatives	**CPU based smartcard ICs**
Firsts	(1) first **advanced encryption standard [AES]** (2) first **relay attack detection** and (3) first system level **Common Criteria EAL 4+**
Proof Points	**pre-issuance** of cards **prior** to infrastructure upgrade **migration demonstrations** at **Cartes 2008**
Support	NXP MIFARE Classic **Migration Team**

Goal of RM&DOORS improvement @ NXP (BL-ID)

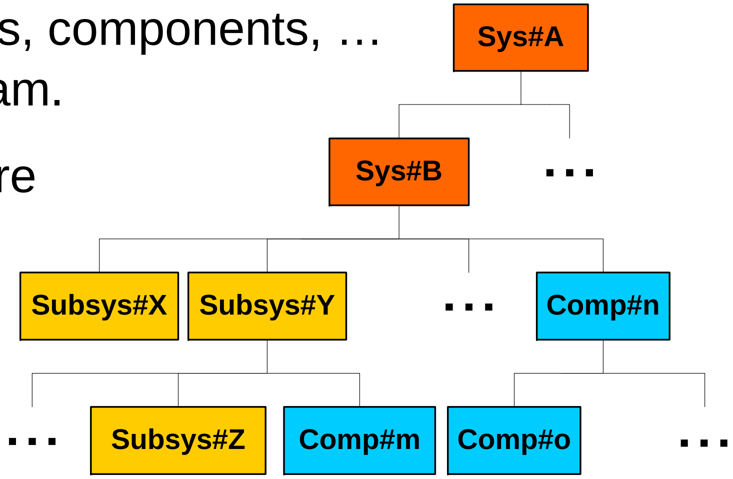
The goal of the RM&DOORS improvement has been **NOT just** to have a solution to **document requirements** !!!

NXP needed an efficient solution to:

- Handle **re-use** on all levels (stopping copying specs over and over)
- Establish **traceability** between top-level features, top-level customer requirements, top-level system requirements, low-level component requirements, architecture elements and test results;
- Ensure that components (including IPs) do what is needed on the next higher level
- **Track/analyze/ensure status** for:
  - satisfaction
  - qualification
  - realization and
  - testing
- **Review** requirements, architecture and test specifications
- …

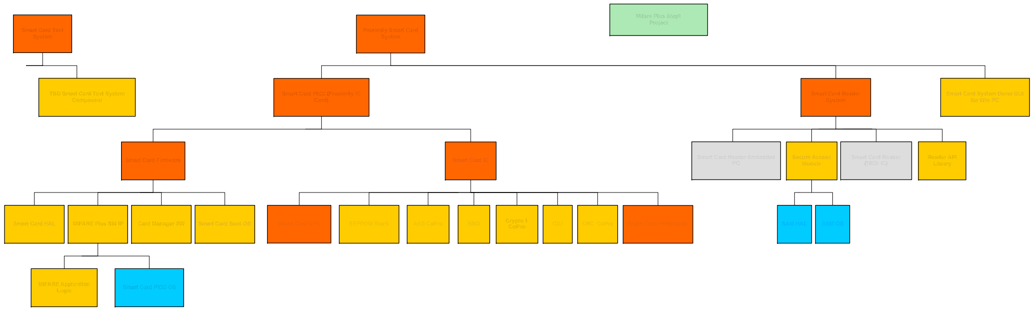
Scope Diagrams for System Decomposition

- The decomposition of a system into subsystems, components, … (called scopes) can be shown in a scope diagram.
- A box in the scope diagram means that there are requirements for this scope (as a black box) [but not necessarily that there will be a DOORS module with them]
- The scope diagram helps to come up with the traceability schema for a project.



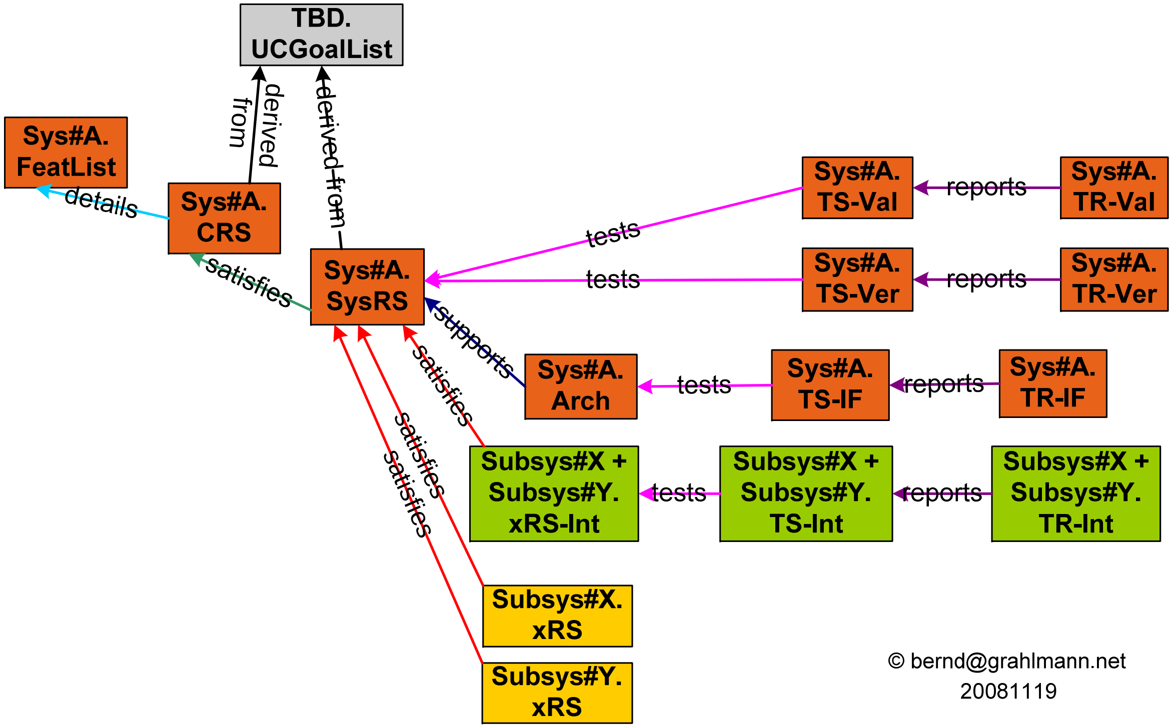
© bernd@grahlmann.net
20081119

- Scope diagrams should be **in synch** with architecture, work packages, configuration items, … => helping architects, project managers, configuration managers, …



Traceability schema

- The traceability schema for a project:
  - shows all specifications (i.e. DOORS modules) with their traceability
  - is derived from scope diagram using / applying a pattern
  - complexity, maturity, … influence decision whether or not there will be, e.g., a DOORS module for a requirements specification for a certain component



TBD. UCGoalList

Sys#A. FeatList

Sys#A. CRS

Sys#A. SysRS

Sys#A. Arch

Sys#A. TS-Val

Sys#A. TR-Val

Sys#A. TS-Ver

Sys#A. TR-Ver

Sys#A. TS-IF

Sys#A. TR-IF

Subsys#X + Subsys#Y. xRS-Int

Subsys#X + Subsys#Y. TS-Int

Subsys#X + Subsys#Y. TR-Int

Subsys#X. xRS

Subsys#Y. xRS

derived from

derived from

details

satisfies

supports

satisfies

satisfies

satisfies

tests

tests

tests

tests

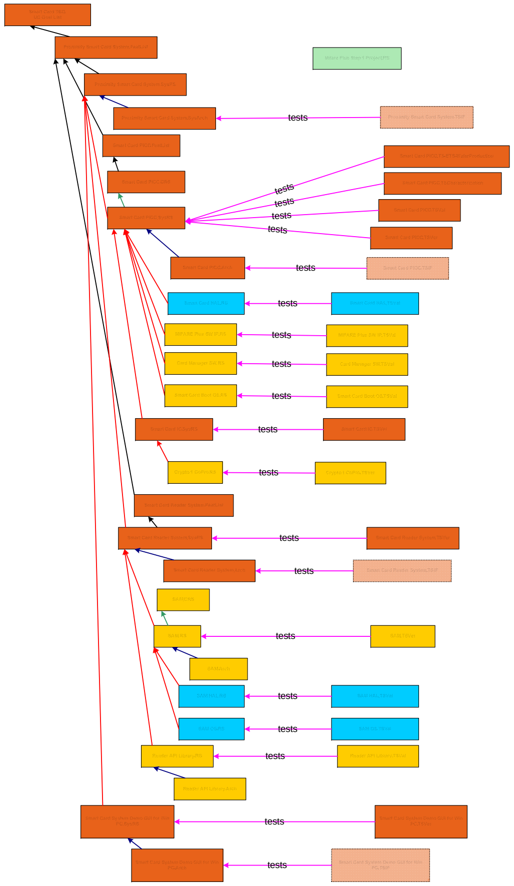
reports

reports

reports

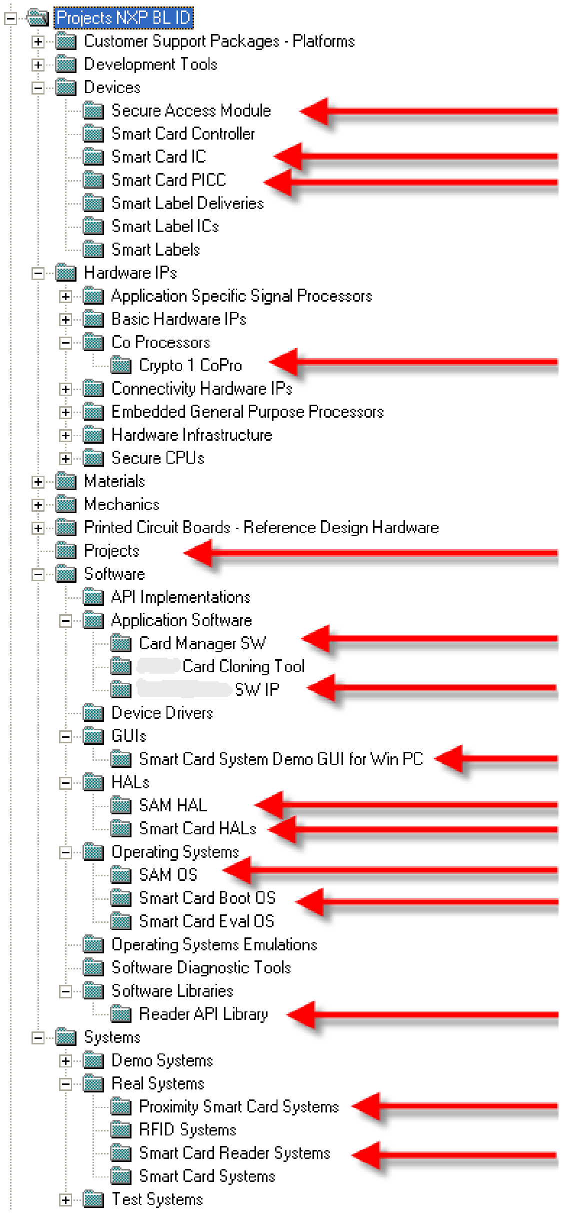
reports

© bernd@grahlmann.net
20081119



Scope based and Re-use driven DOORS Database Structure

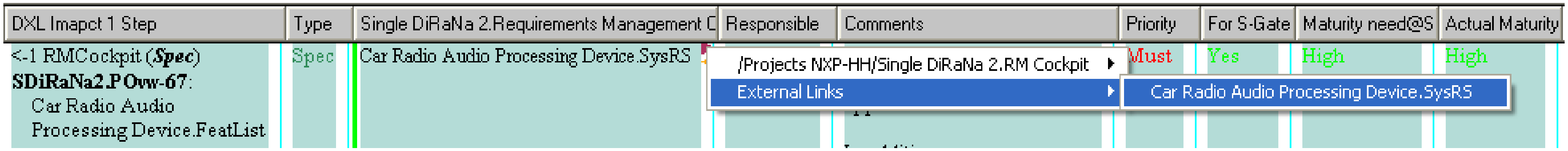
- Problem:
  - Plenty of different systems, subsystems, components, … (i.e. scopes) are built
  - A scope (e.g., a certain component) can be used in multiple NXP projects / products
  - For each scope there can be requirements, test, architecture, … specifications (i.e. DOORS modules).
  - DOORS does not offer ‘symbolic links’ ☹
- Solution:
  - (Not project centric, but) scope based (i.e., One DOORS project per scope – even if, e.g., a component is used in multiple NXP projects)
  - E.g., all specifications for all ‘Smart Card Boot OS’ are under ‘Projects->Software->Operating Systems->Smart Card Boot OS’
  - Minimizes duplications, redundancy, …



```
Projects NXP BL ID
  Customer Support Packages - Platforms
  Development Tools
  Devices
    Secure Access Module
    Smart Card Controller
    Smart Card IC
    Smart Card PICC
    Smart Label Deliveries
    Smart Label ICs
    Smart Labels
  Hardware IPs
    Application Specific Signal Processors
    Basic Hardware IPs
    Co Processors
      Crypto 1 CoPro
    Connectivity Hardware IPs
    Embedded General Purpose Processors
    Hardware Infrastructure
    Secure CPUs
  Materials
  Mechanics
  Printed Circuit Boards - Reference Design Hardware
  Projects
  Software
    API Implementations
    Application Software
      Card Manager SW
      [redacted] Card Cloning Tool
      [redacted] SW IP
    Device Drivers
    GUIs
      Smart Card System Demo GUI for Win PC
    HALs
      SAM HAL
      Smart Card HALs
    Operating Systems
      SAM OS
      Smart Card Boot OS
      Smart Card Eval OS
    Operating Systems Emulations
    Software Diagnostic Tools
    Software Libraries
      Reader API Library
  Systems
    Demo Systems
    Real Systems
      Proximity Smart Card Systems
      RFID Systems
      Smart Card Reader Systems
      Smart Card Systems
    Test Systems
```

Usage of Requirements Management Cockpits

- One Requirements Management Cockpit per NXP project / product:
  - Being Requirements Management Plan
  - Showing Scope Diagram (i.e. System Decomposition)
  - Showing Traceability Schema
  - Showing DOORS DB structure indicating corresponding DOORS projects
  - Listing all systems, subsystems, components (i.e. scopes)
  - (For each scope) Listing all requirements, test, architecture, … specifications (i.e. DOORS modules):
    - Giving name, responsible(s), status, TBDs, priorities, maturity levels needed at gates, traceability information (via DOORS links), …
    - Allowing to open module via DOORS external links
    - Ready to accommodate automatically generated metrics, …

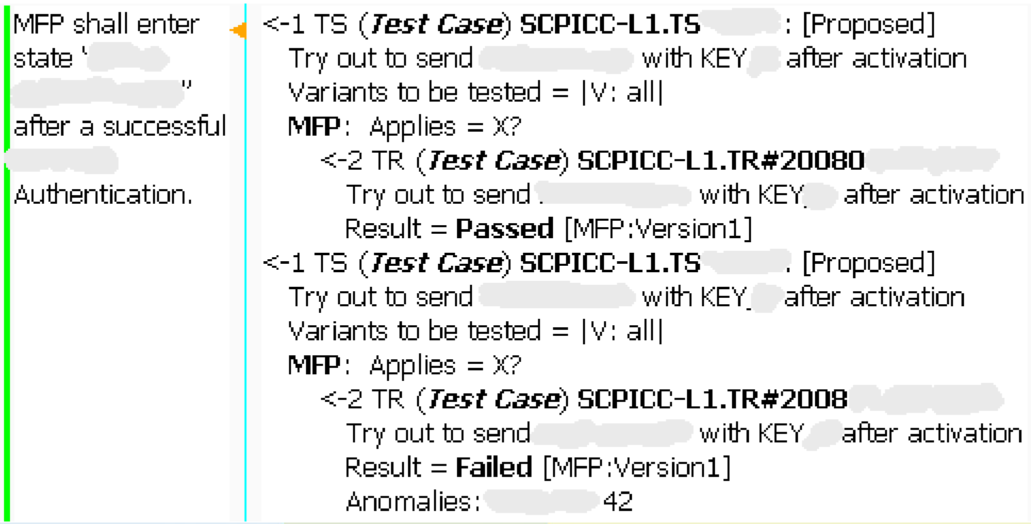


DXL Imapct 1 Step	Type	Single DiRaNa 2.Requirements Management C	Responsible	Comments	Priority	For S-Gate	Maturity need@S	Actual Maturity
<-1 RMCockpit (***Spec***) **SDiRaNa2.POvw-67**: Car Radio Audio Processing Device.FeatList	Spec	Car Radio Audio Processing Device.SysRS			Must	Yes	High	High

/Projects NXP-HH/Single DiRaNa 2.RM Cockpit
External Links
Car Radio Audio Processing Device.SysRS

Sophisticated DOORS Module Templates

- Attributes ‘Workforce’ to come up with ‘standard’ set of attributes (incl. types, usage information, …) for the different types of specifications.
- DOORS module templates contain:
  - Module level attribute documenting the template, its attributes, views, …
  - Attributes and types
  - Module level attributes driving print-out (e.g. via WEXP or DocExpress)
  - Views for the main tasks (showing each time the ‘relevant’ attributes)
  - DXL attributes automatically visualizing traceability (1-step or all steps – optionally filtered on opened and currently displayed objects) in a sophisticated and powerful way (automatically showing the relevant attributes in a ‘nicely’ formatted way)

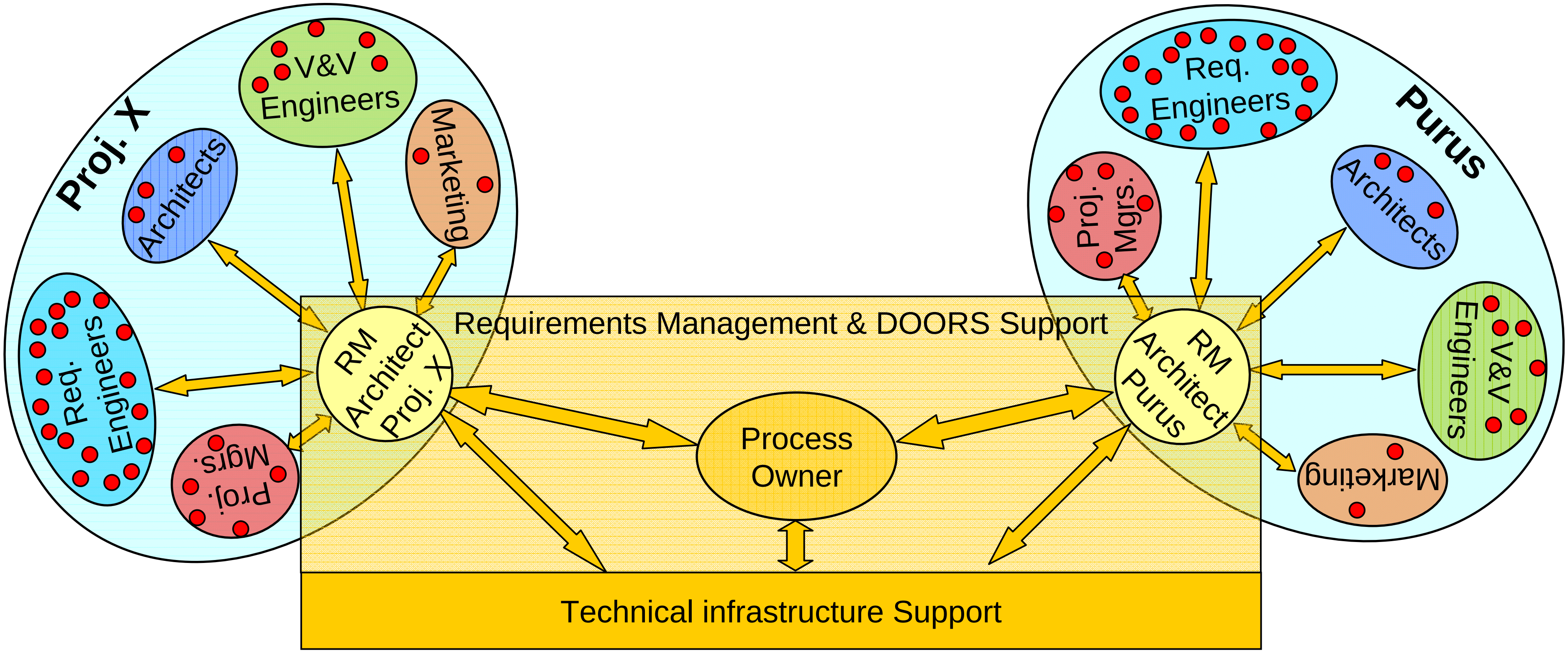


MFP shall enter state '[redacted] [redacted]" after a successful [redacted] Authentication.

```
<-1 TS (Test Case) SCPICC-L1.TS[redacted]: [Proposed]
  Try out to send [redacted] with KEY[redacted] after activation
  Variants to be tested = |V: all|
  MFP: Applies = X?
    <-2 TR (Test Case) SCPICC-L1.TR#20080[redacted]
      Try out to send [redacted] with KEY[redacted] after activation
      Result = Passed [MFP:Version1]
<-1 TS (Test Case) SCPICC-L1.TS[redacted]. [Proposed]
  Try out to send [redacted] with KEY[redacted] after activation
  Variants to be tested = |V: all|
  MFP: Applies = X?
    <-2 TR (Test Case) SCPICC-L1.TR#2008[redacted]
      Try out to send [redacted] with KEY[redacted] after activation
      Result = Failed [MFP:Version1]
      Anomalies: [redacted]42
```

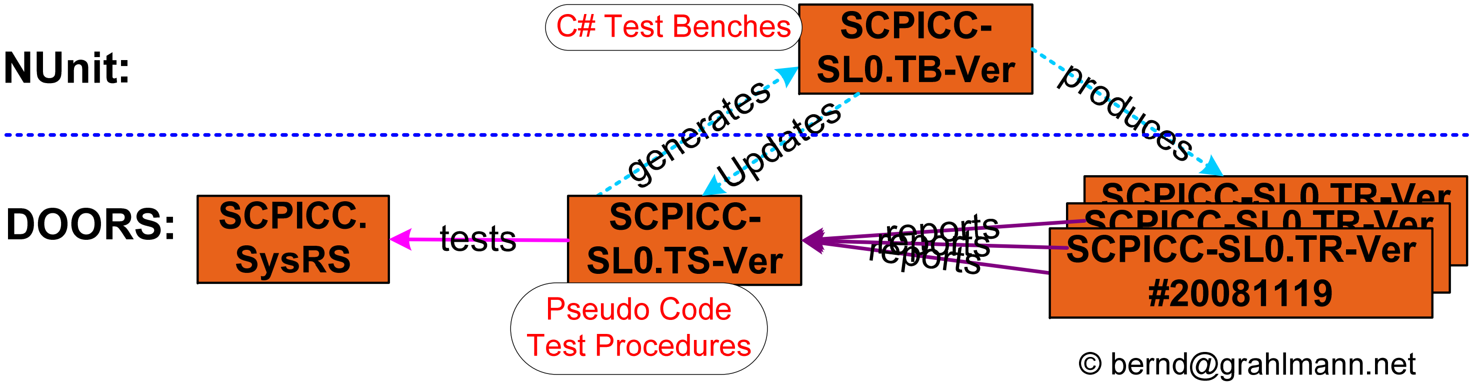
RM & DOORS architects dedicated to NXP projects

1. Technical Infrastructure Support
2. Project (Office) / Process Owner
3. RM&DOORS Architect (for set-up, coaching, checks, metrics, expert parts, ...)



Problem - Automating system level testing in MIFARE Plus

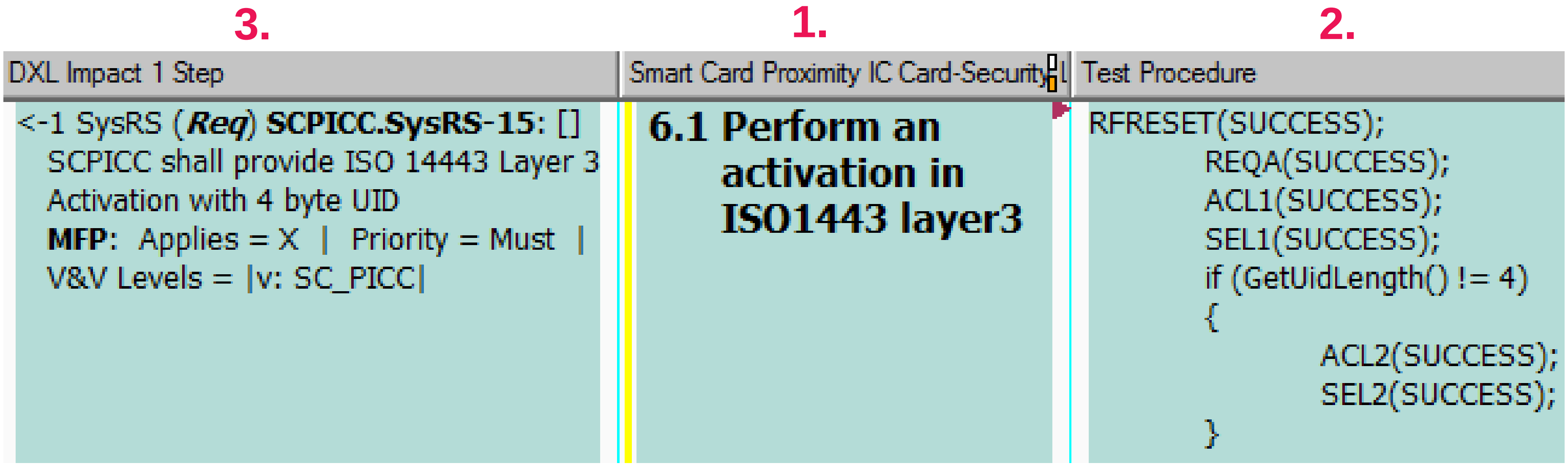
- On system level, rigorous extensive testing (> 4000 test cases) is necessary to ensure correct ‘functioning’ of the Smart Card PICC (in particular, wrt. Security);
- Traceability is needed from test results via test cases in test specification(s) to requirements (in DOORS);
- Powerful analysis & metrics (verification coverage, …) is a must;
- In test specifications you want to deal with lightweight pseudo code test procedures;
- The full C# test benches are ‘done’ in NUnit (from NUnit.org);



© bernd@grahlmann.net
20081120

Test Cases in DOORS with ‘pseudo code’ for test procedures

- In test specifications you want to (in particular):
  1. give a tangible test case name
  2. deal with lightweight pseudo code test procedures (rather than full fledge C# code);
  3. see traceability to requirements

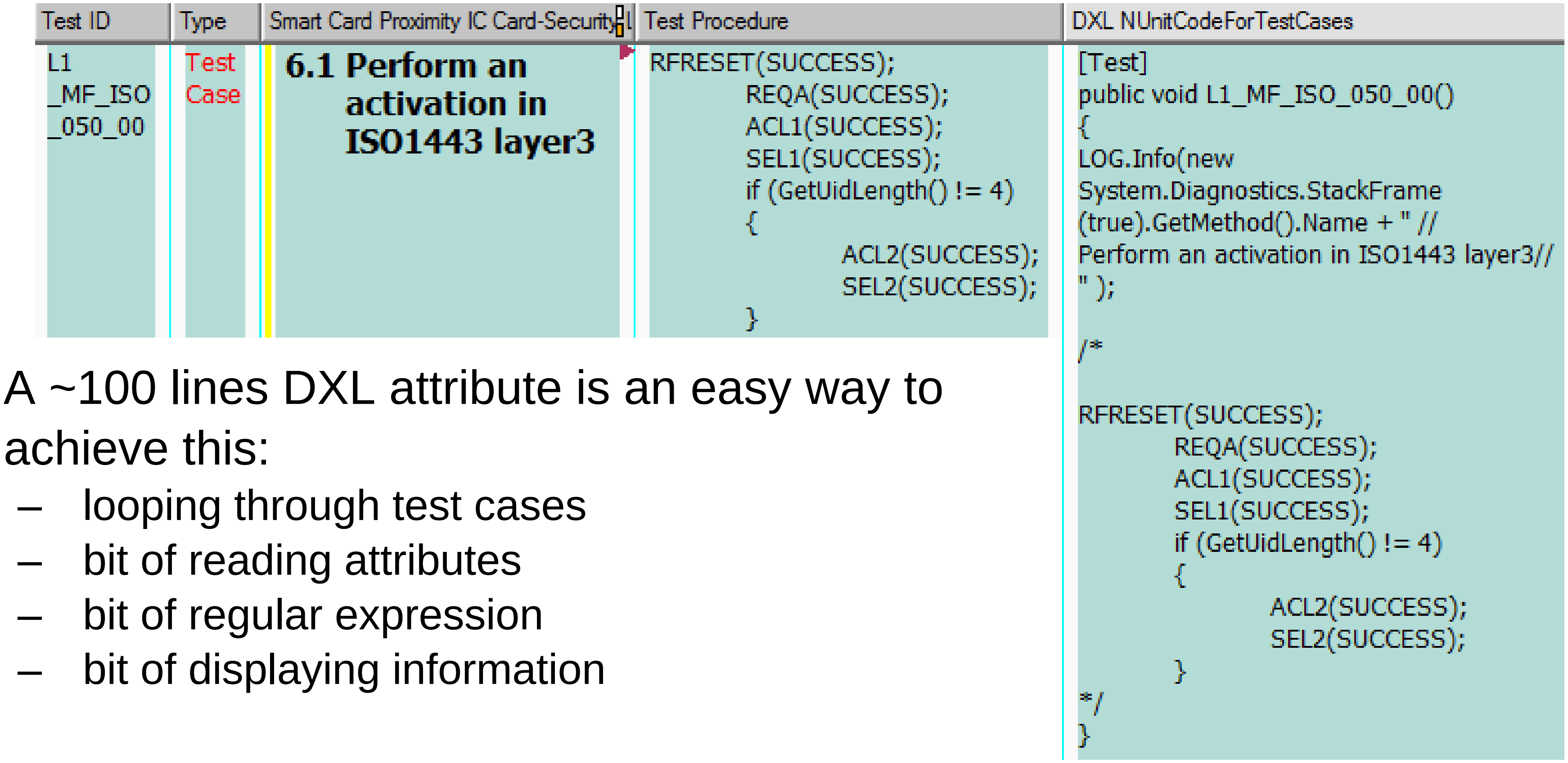


DXL Impact 1 Step	Smart Card Proximity IC Card-Security L	Test Procedure

<-1 SysRS (***Req***) **SCPICC.SysRS-15**: []
SCPICC shall provide ISO 14443 Layer 3 Activation with 4 byte UID
MFP: Applies = X | Priority = Must |
V&V Levels = |v: SC_PICC|
	6.1 Perform an activation in ISO1443 layer3	RFRESET(SUCCESS);
		REQA(SUCCESS);
		ACL1(SUCCESS);
		SEL1(SUCCESS);
		if (GetUidLength() != 4)
		{
		ACL2(SUCCESS);
		SEL2(SUCCESS);
		}

DXL attribute generating (skeletons for) C# test benches

- From the light weight and more tangible ‘pseudo code’ test procedures (and some other attributes) you want to get (at least skeletons of) the full fledge C# Code test benches in an automatic (and error free) way:



Test ID	Type	Smart Card Proximity IC Card-Security	Test Procedure	DXL NUnitCodeForTestCases

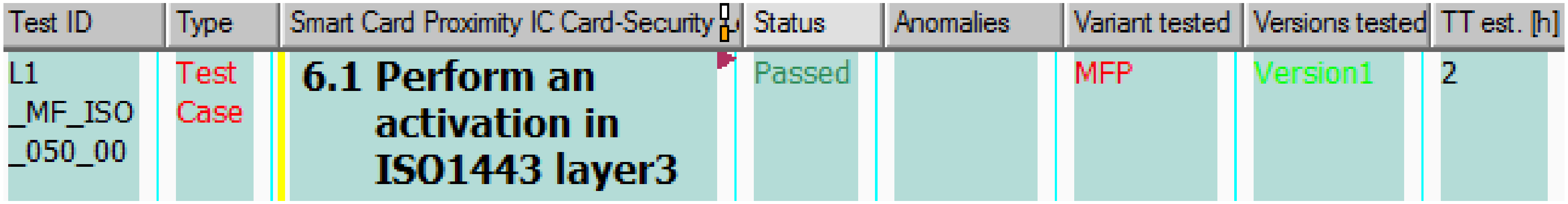
L1_MF_ISO_050_00	Test Case	**6.1 Perform an activation in ISO1443 layer3**	RFRESET(SUCCESS);
			REQA(SUCCESS);
			ACL1(SUCCESS);
			SEL1(SUCCESS);
			if (GetUidLength() != 4)
			{
			ACL2(SUCCESS);
			SEL2(SUCCESS);
			}
				[Test]
				public void L1_MF_ISO_050_00()
				{
				LOG.Info(new System.Diagnostics.StackFrame (true).GetMethod().Name + " // Perform an activation in ISO1443 layer3// ");
				
				/*
				
				RFRESET(SUCCESS);
				REQA(SUCCESS);
				ACL1(SUCCESS);
				SEL1(SUCCESS);
				if (GetUidLength() != 4)
				{
				ACL2(SUCCESS);
				SEL2(SUCCESS);
				}
				*/
				}

- A ~100 lines DXL attribute is an easy way to achieve this:
  - looping through test cases
  - bit of reading attributes
  - bit of regular expression
  - bit of displaying information

Test benches runs resulting in ‘importable’ CSV files I

- The DXL generated (skeletons of) C# Code test benches contain all the necessary information such that test benches:
  - have the reference to the test case
  - write ‘importable’ CSV files with the test results when run (allowing simple standard import into DOORS test result modules)

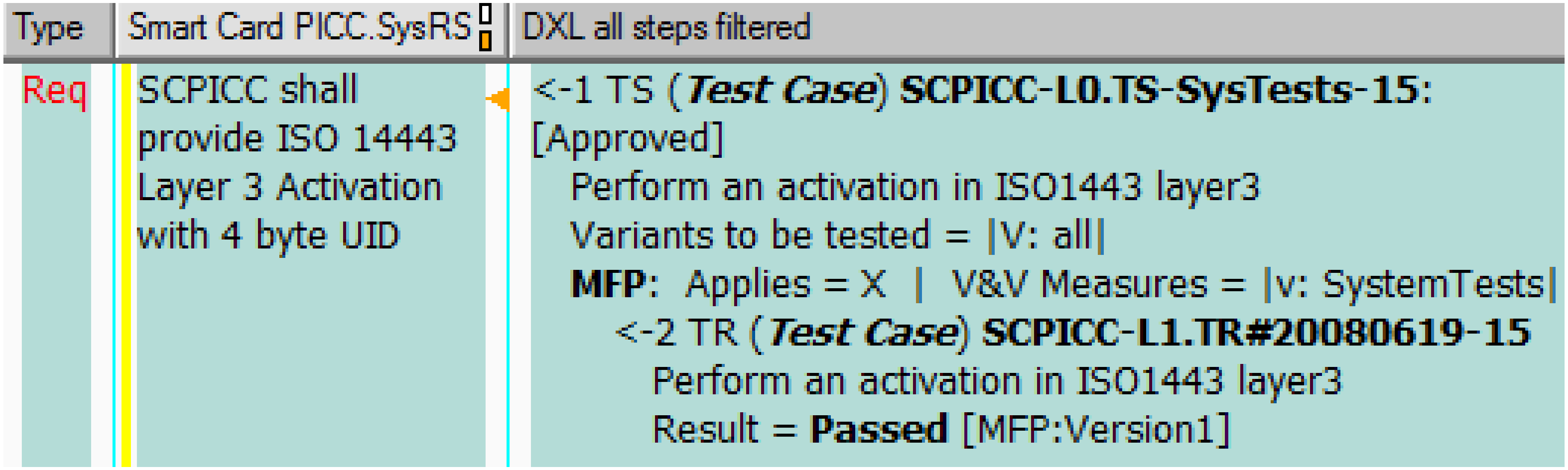
```
a:TestID,a:OverallTestRunStatus,a:TestTimeActual, …
L1_MF_ISO_050_00,Passed,2, …
…
```



Test ID	Type	Smart Card Proximity IC Card-Security	Status	Anomalies	Variant tested	Versions tested	TT est. [h]
L1_MF_ISO_050_00	Test Case	**6.1 Perform an activation in ISO1443 layer3**	Passed		MFP	Version1	2

Test benches runs resulting in ‘importable’ CSV files II

- Yielding full traceability in DOORS:
  Requirement(s) <- Test Case (in Test Specification) <- Test Result
- Allowing to filter on (particular) failed tests, …
- Allowing to establish metrics, …



Type	Smart Card PICC.SysRS	DXL all steps filtered
Req	SCPICC shall provide ISO 14443 Layer 3 Activation with 4 byte UID	<-1 TS (*Test Case*) **SCPICC-L0.TS-SysTests-15:** [Approved] Perform an activation in ISO1443 layer3 Variants to be tested = |V: all| **MFP:** Applies = X | V&V Measures = |v: SystemTests| <-2 TR (*Test Case*) **SCPICC-L1.TR#20080619-15** Perform an activation in ISO1443 layer3 Result = **Passed** [MFP:Version1]

Option to feed back test bench updates into DOORS

- Additional new option to:
  - Update C# test benches in NUnit
  - Add new C# test benches in NUnit
  - Generate ‘importable /update-able’ (to DOORS) TSV files from NUnit for those changed / new test benches

Summary

- The chosen approach (with *scope diagrams for system decomposition*, *traceability schemas*, *scope based and re-use driven DOORS database structure*, usage of *requirements management cockpits*, sophisticated *DOORS module templates*, *RM & DOORS architects* dedicated to NXP projects, …) allowed a successful DOORS deployment in various projects at NXP (and in particular in their Business Line Identification) solving the various requirements management and/or validation & verification related challenges ☺
- This good approach together with the (well used) 'Power' of DOORS and DXL allowed to go one step further automating testing with Telelogic DOORS in the MIFARE Plus project @ NXP (saving a lot of effort and increasing quality) ☺

➔*Good success* ☺

Many thanks to the MIFARE Plus Team !!!

and Dirk Lützelberger, Adrian Haw and Dolf Riedel from NXP Hamburg.

Discussion / Questions

- Please pose your questions, give feedback, …

*Do not hesitate to contact me if you have further questions,
feedback, are looking for a
trainer and/or consultant on
requirements engineering / management / development
and, of course, Telelogic DOORS ;-)*

Dr. Bernd GRAHLMANN
+33 682866803
Bernd@Grahlmann.net
www.grahlmann.net

Thank You !!!