# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

## IBM IoT Exchange 2019
## London, Paris & Berlin

# Nikolai Stein

## Solution Architect & DOORS/DNG Expert

- RE&RM Methods and Concepts
- Data-Exchange with ReqIF and OSLC
- DNG-Interfaces and Automation
- Data Migration
- Global Configuration Management in DNG
- DOORS Server Security

- Since 2002 in automotive domain in the area of requirements engineering

# REQUISIS GmbH

- Based in Germany
- Focus on tool based Requirements Management
- Many tools and addons for DOORS 9
  - Document Generation, Data Exchange, Data Synchronization, Work Offline, DNG Migration
  - DOORS Security Proxy
- Growing List of Tools for DNG
  - Data exchange automation
  - DOORS9-DNG-Synchronization
- Tools for ReqIF
- Many happy customers
  - Most customers are in the automotive domain

# Agenda

- Short Introduction
- Migration Basics
- Data Analysis
- Migration Preparation
- Performing the Migration

- Topic is complex
- Time is limited, so topics are kept short.

If you would like to get the presentation or have detailed questions, email me!

nikolai.stein@requisis.com

# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

## Migration Basics

# Important Differences DOORS and DNG (relevant for migration)

| | DNG | DOORS |
|---|---|---|
| Structure | Project Areas -> Components -> Folders -> Modules / Collections | Projects & Folders -> Modules |
| Artifact types | Multiple artifact types different attribute-sets | Only one type of objects per Module „User-Made“ Object Type by setting attribute value All objects share attribute-set |
| Attribute / Type Definition | Component-Wide | Module-Wide |
| Text Formatting | Only in Primary Text | In all Text and String attributes |
| Binary data | Embedded Artifacts Saved binary directly Only in Primary Text | Embedded OLE-Objects Saved as OLE in RTF In all Text Attributes |
| Links | Different link types | Link types by using different named link modules |

# Types of Migration – By Size

- **Big Bang Migration**
  - Migrate the whole database at the same time
- **Project-Wise Migration**
  - Migrate a project when the project is ready to shift to DNG
  - Migrate related projects only if necessary (e.g. as a Shadow-Migration)
- **Module-Wise Migration**
  - Migrate single Modules when the content is required
- **Shadow-Migration**
  - Migrate data that is supposed to stay read-only in DNG

# Types of Migration – By Type

| Type | Steps | Consequences | Technology |
|---|---|---|---|
| Migrate as-it-is | • Just migrate | • Duplicated Types and attributes | ReqIF<br>Migiz (better Results, but one-time only) |
| Align data types before migration | • Adjust the data types that they are equal<br>• Ensure enumeration attributes share the same values<br>• Add RDF-URIs to all attributes and types | • Modified Data in DOORS<br>• Mostly clean data types in DNG<br>• Risk of duplication in some cases<br>• Data in DNG more or less 1:1 from DOORS | |
| Align data types after migration | • Adjust the data types in DNG after Migration (nearly impossible!) | • Nearly impossible to reach a clean data type model in DNG<br>• Data in DNG more or less 1:1 from DOORS | |
| Map, Transform & Migrate | • Create data model in DNG first<br>• Map data types and enum values from DOORS to DNG<br>• Select attributes to be merged | • Clean data type model<br>• Depending on Tool: Data ready to use in DNG | 3rd Party Tools |

# Focus of this session

- Project-Wise-Migration with Shadow-Migration of related content
- Usage of “Map, Transform & Migrate” with a rule-based mapping

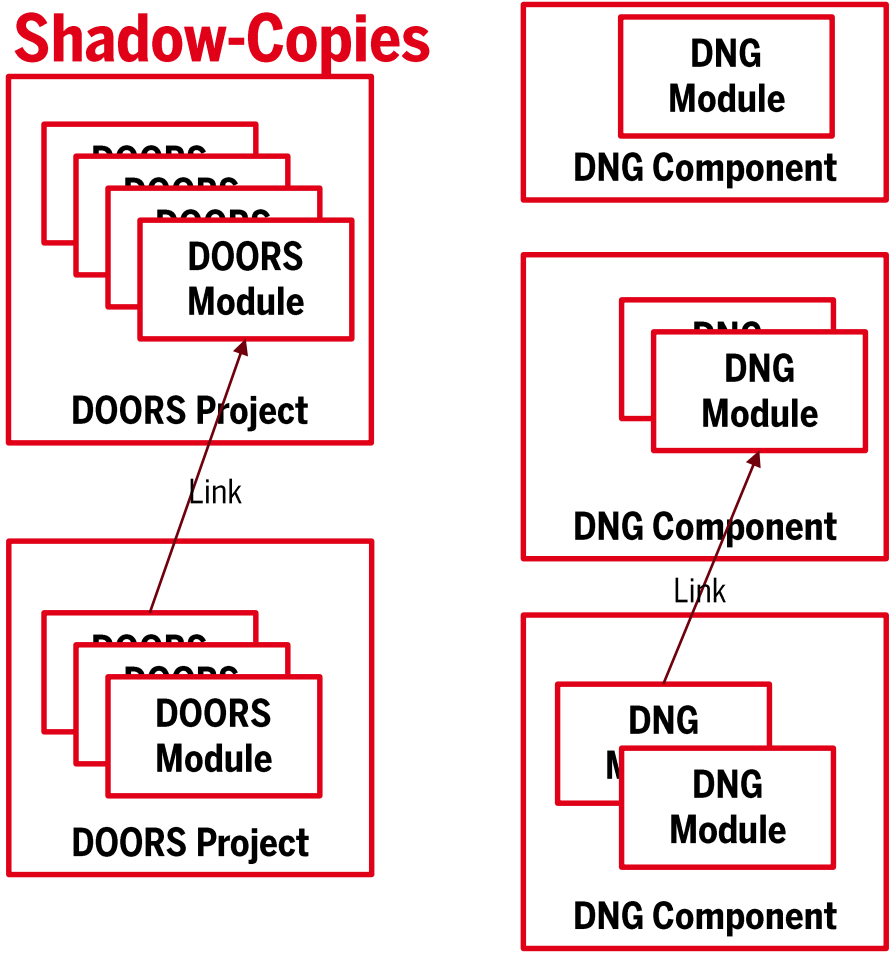
- Reasons:
  - Our Customers have big databases and many users (1000+) -> “Big Bang Migration” not possible
  - Pragmatic approach with best results in data quality and user experience
  - We think this is the most sustainable and cleanest migration approach

- Created tools to support this approach:
  - DOORS Attribute Analyzer
  - requisis_MiX

- Prerequisites:
  - Existing Data Types and DNG Modules in DNG
  - Mapping Rules to automate mapping

# Project-Wise Migration with Shadow-Copies

- Create DNG Components based on a Template
- Create DNG Modules to define the migration target
- Map modules and data types
- Migrate data into the DNG Modules
- Migrate Links
- Switch Shadow-Copies and migrated content read only
- Update Shadow-Copies regularly

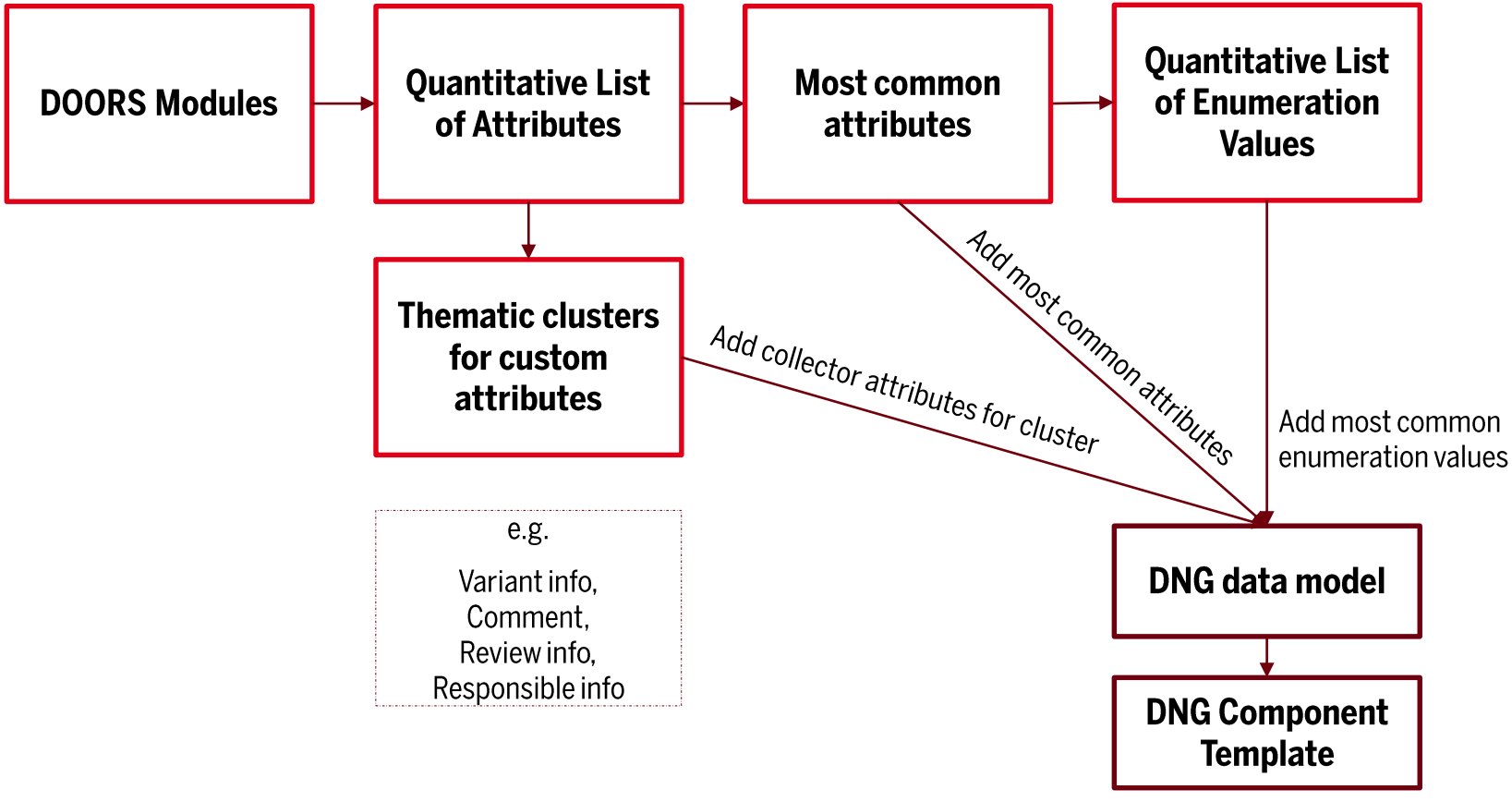



# Workflow
## for Map, Transform & Migrate

# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

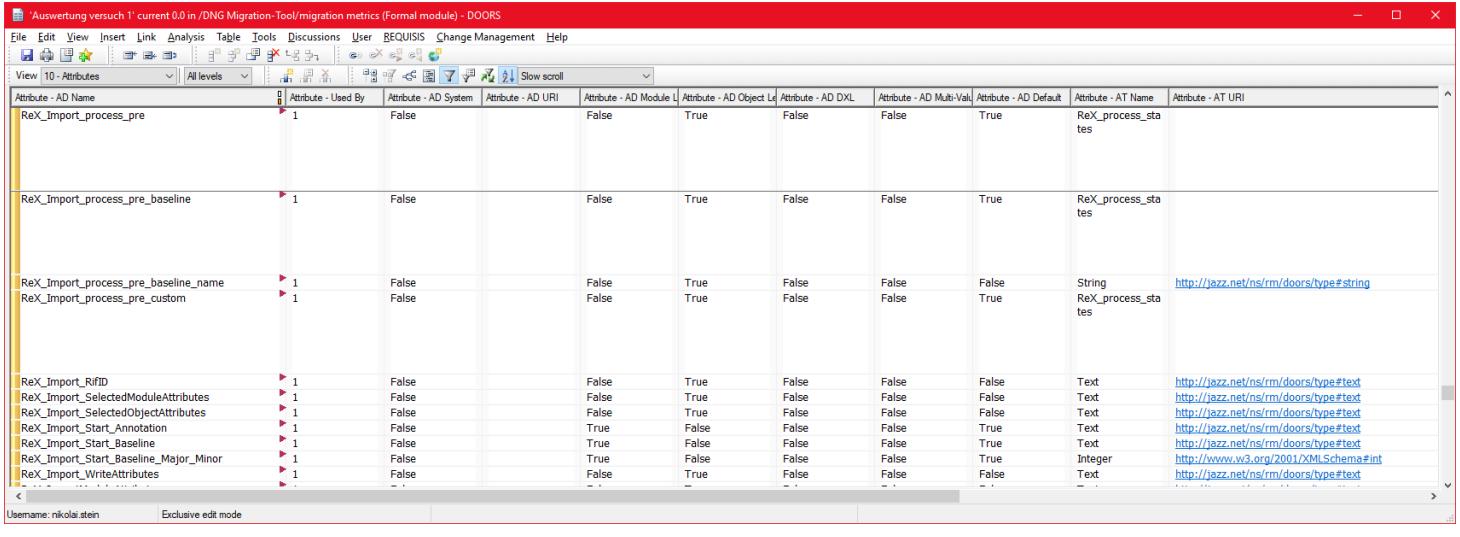
## Data Analysis

# Data analysis for defining the DNG data type model



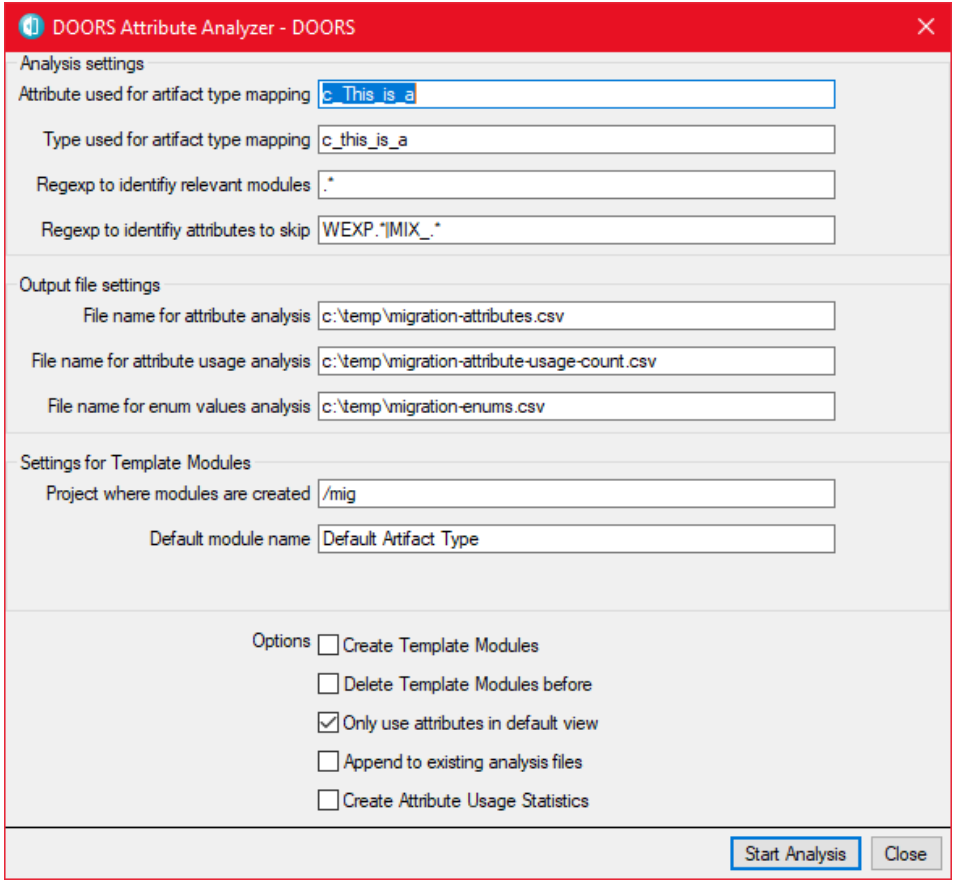
# Tools for Data Analysis: DOORS Built-In Migration Metrics

- Select projects and modules
- Report is created as a DOORS Module
- Counts Objects, Pictures, OLEs, Tables…
- Analyzes Attributes
  - Each equal attribute is listed and counted with links to the modules and a list of enum values if applicable



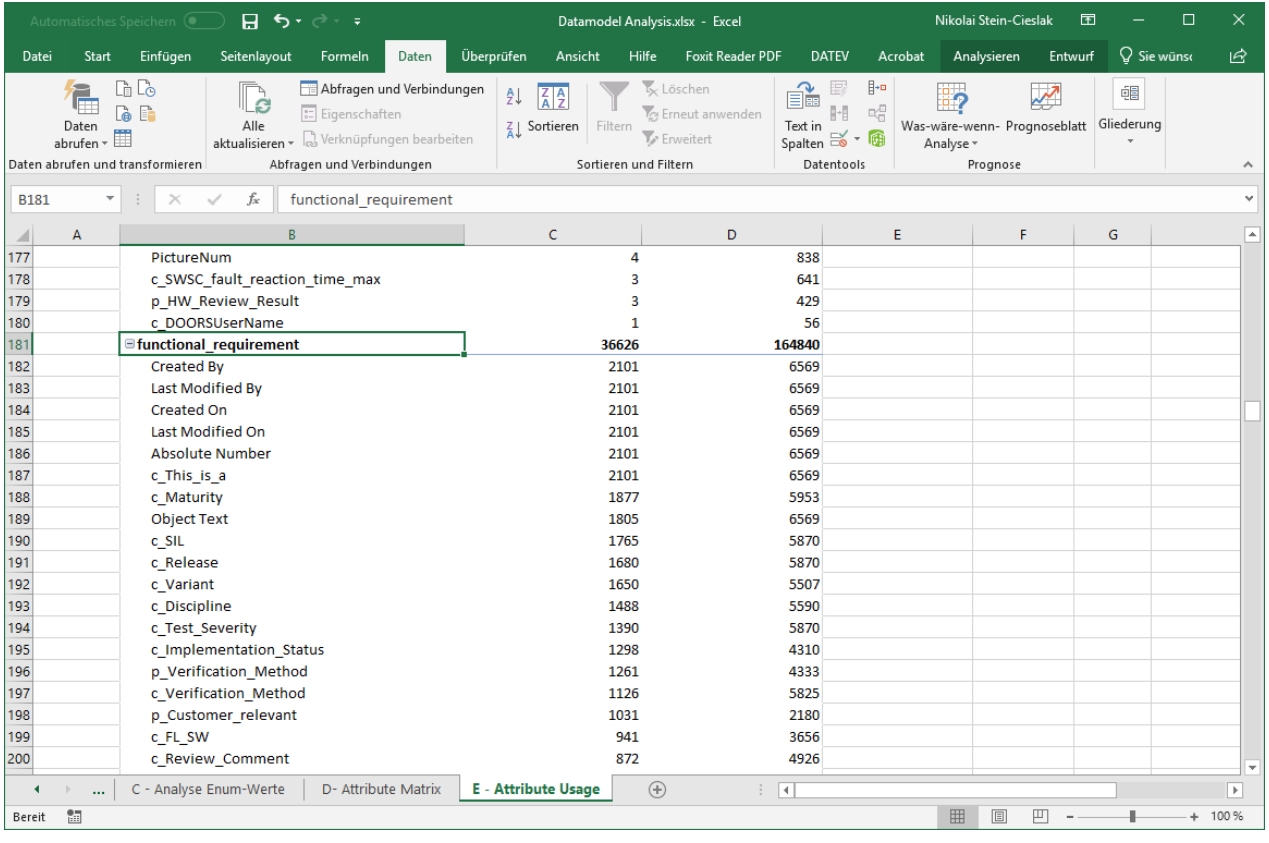
# Tools for Data Analysis: DOORS Attribute Analyzer

- Analyzes and Counts
  - Existing Attributes and their types
  - Per “Object Type“: attributes with values
  - Enum attributes with possible values
- Data Extraction to be used in Excel as Pivot tables
  - Sorted Lists
  - Double-Click on all numbers opens a list of corresponding modules
- Merged data types stored as Attribute and Type Definitions in a DOORS-Module



# Tools for Data Analysis: DOORS Attribute Analyzer

- Attribute Usage per “Object Type”
- To derive artifact types
- Double-Click on numbers shows list of modules

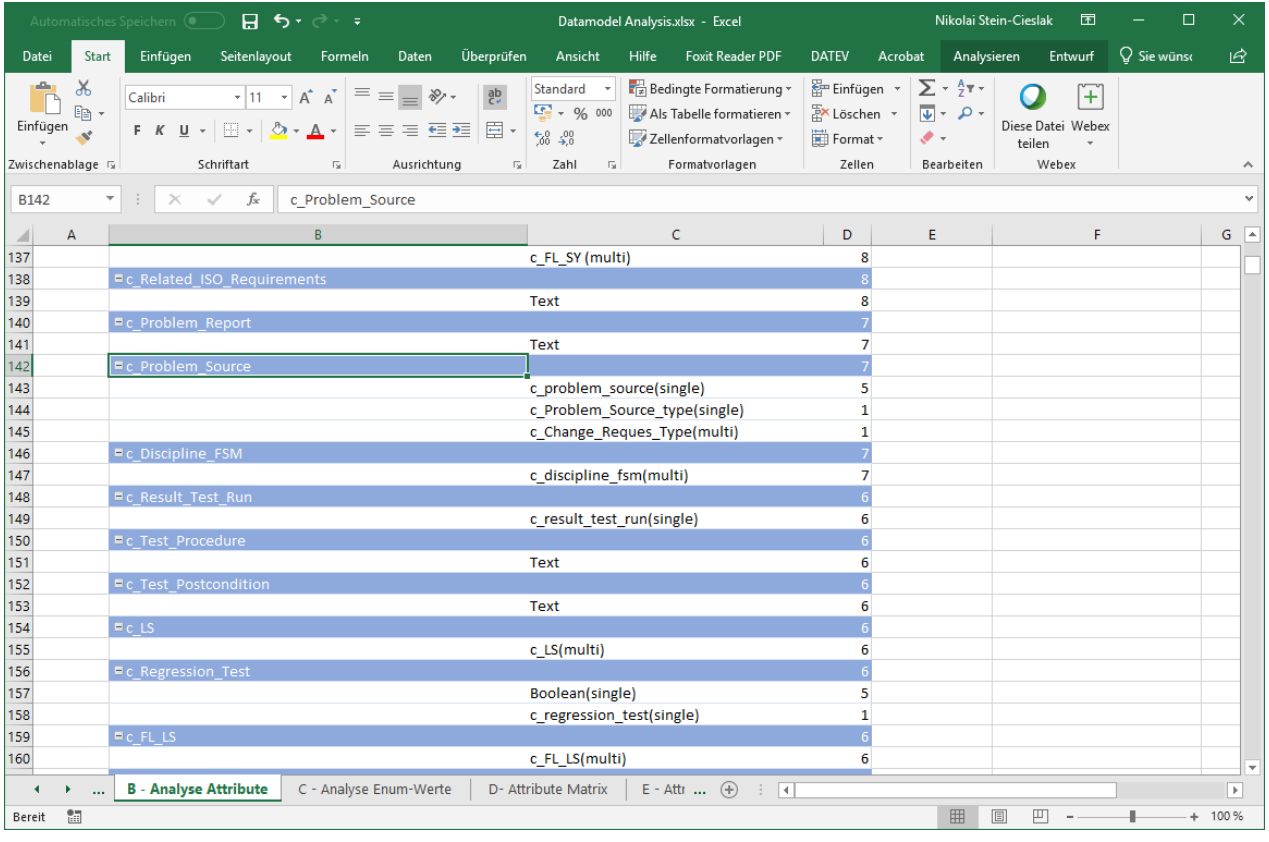



| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 177 | | PictureNum | 4 | 838 | | | |
| 178 | | c_SWSC_fault_reaction_time_max | 3 | 641 | | | |
| 179 | | p_HW_Review_Result | 3 | 429 | | | |
| 180 | | c_DOORSUserName | 1 | 56 | | | |
| 181 | | **functional_requirement** | **36626** | **164840** | | | |
| 182 | | Created By | 2101 | 6569 | | | |
| 183 | | Last Modified By | 2101 | 6569 | | | |
| 184 | | Created On | 2101 | 6569 | | | |
| 185 | | Last Modified On | 2101 | 6569 | | | |
| 186 | | Absolute Number | 2101 | 6569 | | | |
| 187 | | c_This_is_a | 2101 | 6569 | | | |
| 188 | | c_Maturity | 1877 | 5953 | | | |
| 189 | | Object Text | 1805 | 6569 | | | |
| 190 | | c_SIL | 1765 | 5870 | | | |
| 191 | | c_Release | 1680 | 5870 | | | |
| 192 | | c_Variant | 1650 | 5507 | | | |
| 193 | | c_Discipline | 1488 | 5590 | | | |
| 194 | | c_Test_Severity | 1390 | 5870 | | | |
| 195 | | c_Implementation_Status | 1298 | 4310 | | | |
| 196 | | p_Verification_Method | 1261 | 4333 | | | |
| 197 | | c_Verification_Method | 1126 | 5825 | | | |
| 198 | | p_Customer_relevant | 1031 | 2180 | | | |
| 199 | | c_FL_SW | 941 | 3656 | | | |
| 200 | | c_Review_Comment | 872 | 4926 | | | |

# Tools for Data Analysis: DOORS Attribute Analyzer

- Types per Attribute
- Separated for single and multi enum attributes
- To derive attributes and types

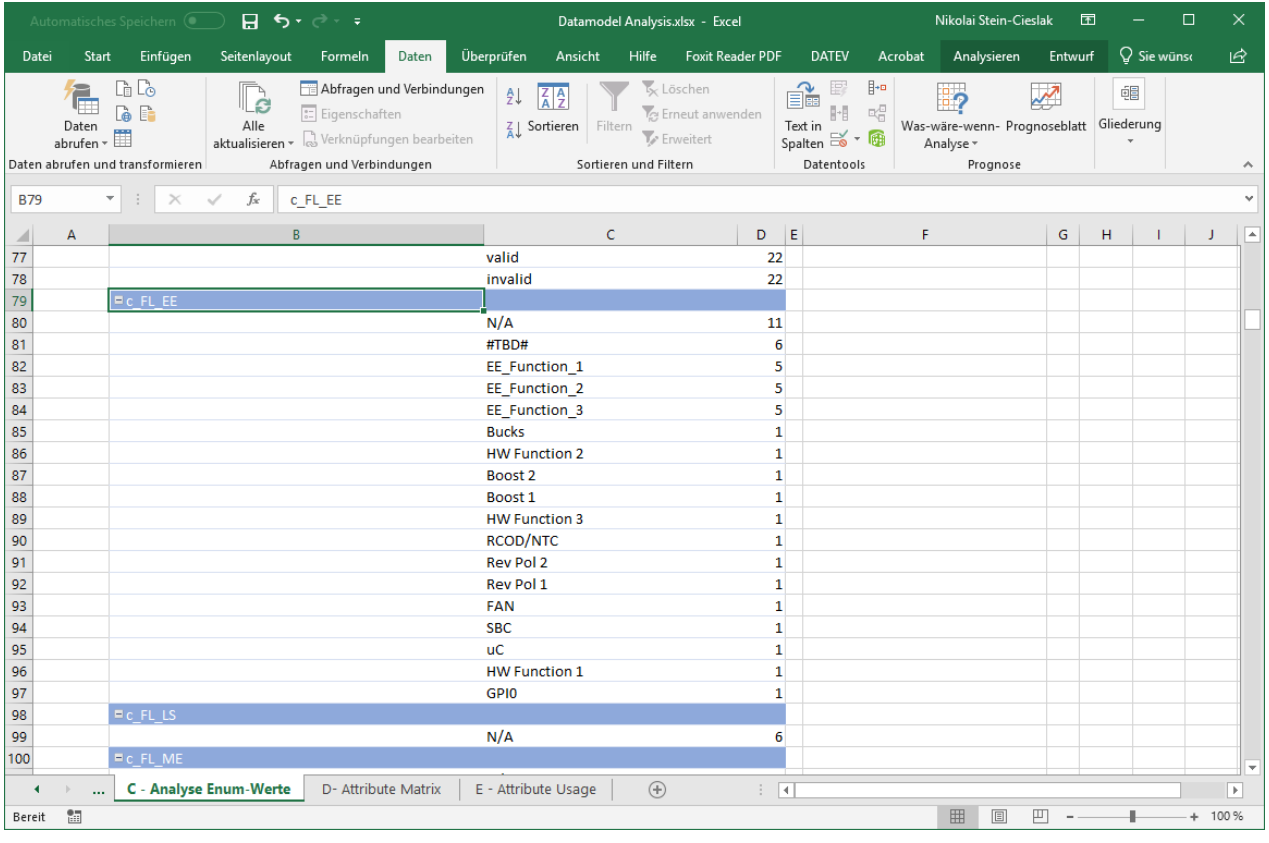
- Double-Click on numbers shows list of modules




| | B | C | D |
|---|---|---|---|
| 137 | | c_FL_SY (multi) | 8 |
| 138 | c_Related_ISO_Requirements | | 8 |
| 139 | | Text | 8 |
| 140 | c_Problem_Report | | 7 |
| 141 | | Text | 7 |
| 142 | c_Problem_Source | | 7 |
| 143 | | c_problem_source(single) | 5 |
| 144 | | c_Problem_Source_type(single) | 1 |
| 145 | | c_Change_Reques_Type(multi) | 1 |
| 146 | c_Discipline_FSM | | 7 |
| 147 | | c_discipline_fsm(multi) | 7 |
| 148 | c_Result_Test_Run | | 6 |
| 149 | | c_result_test_run(single) | 6 |
| 150 | c_Test_Procedure | | 6 |
| 151 | | Text | 6 |
| 152 | c_Test_Postcondition | | 6 |
| 153 | | Text | 6 |
| 154 | c_LS | | 6 |
| 155 | | c_LS(multi) | 6 |
| 156 | c_Regression_Test | | 6 |
| 157 | | Boolean(single) | 5 |
| 158 | | c_regression_test(single) | 1 |
| 159 | c_FL_LS | | 6 |
| 160 | | c_FL_LS(multi) | 6 |

# Tools for Data Analysis: DOORS Attribute Analyzer

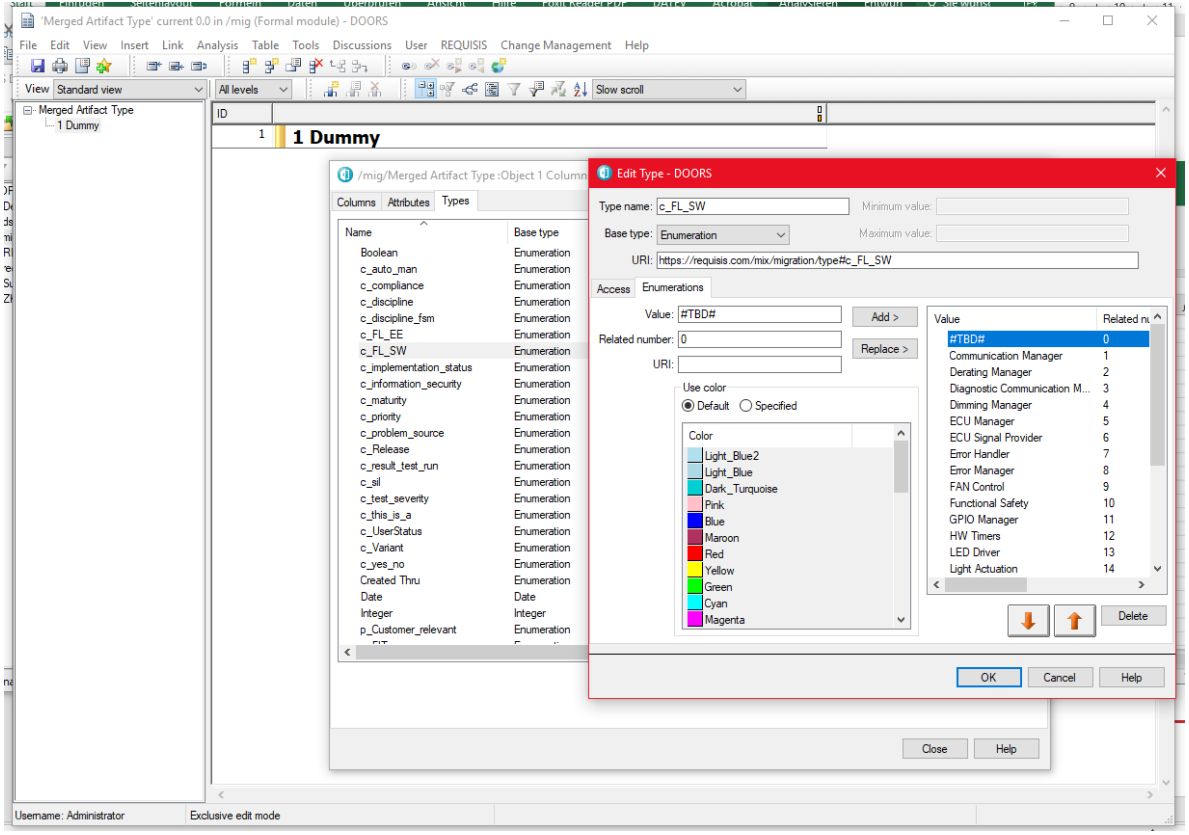
- Enum-Values per Enum-Type
- To derive Enum-Values
- Double-Click on numbers shows list of modules




| | B | C | D |
|---|---|---|---|
| 77 | | valid | 22 |
| 78 | | invalid | 22 |
| 79 | c_FL_EE | | |
| 80 | | N/A | 11 |
| 81 | | #TBD# | 6 |
| 82 | | EE_Function_1 | 5 |
| 83 | | EE_Function_2 | 5 |
| 84 | | EE_Function_3 | 5 |
| 85 | | Bucks | 1 |
| 86 | | HW Function 2 | 1 |
| 87 | | Boost 2 | 1 |
| 88 | | Boost 1 | 1 |
| 89 | | HW Function 3 | 1 |
| 90 | | RCOD/NTC | 1 |
| 91 | | Rev Pol 2 | 1 |
| 92 | | Rev Pol 1 | 1 |
| 93 | | FAN | 1 |
| 94 | | SBC | 1 |
| 95 | | uC | 1 |
| 96 | | HW Function 1 | 1 |
| 97 | | GPI0 | 1 |
| 98 | c_FL_LS | | |
| 99 | | N/A | 6 |
| 100 | c_FL_ME | | |

# Tools for Data Analysis: DOORS Attribute Analyzer

- Merged Attribute and Type Definitions in DOORS
- Can be used to export types and attributes to DNG using ReqIF
- Attributes and Types are created in DNG during import using ReqIF
- Some minimal cleanup in DNG needed (e.g. remove created Module Artifact Type)



# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

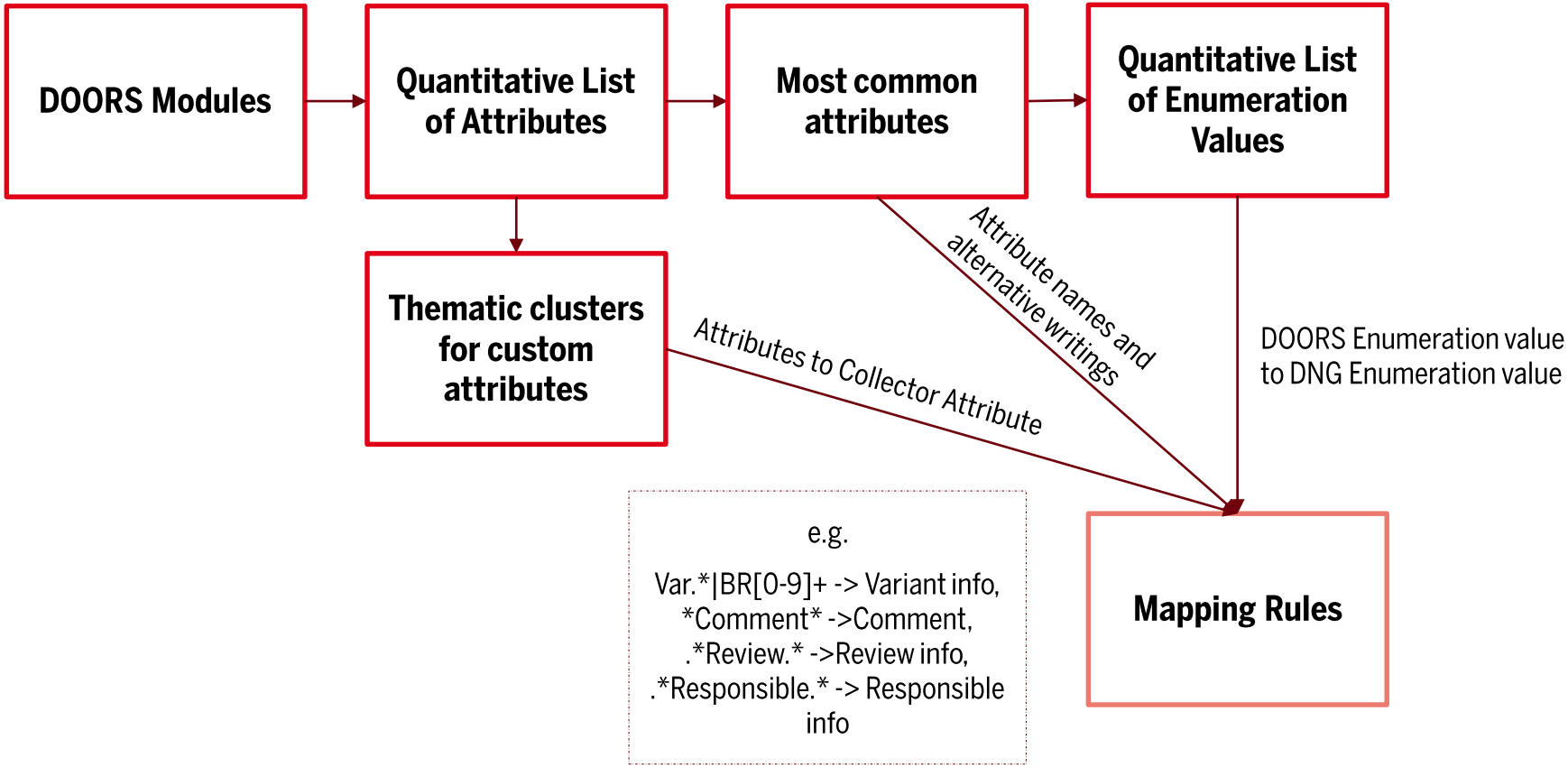
## Create Mapping Rules

# Goals for creating mapping rules

- Reduce manual effort to a minimum
- Reduce mis-configuration
- Increase Quality by always having the same mapping

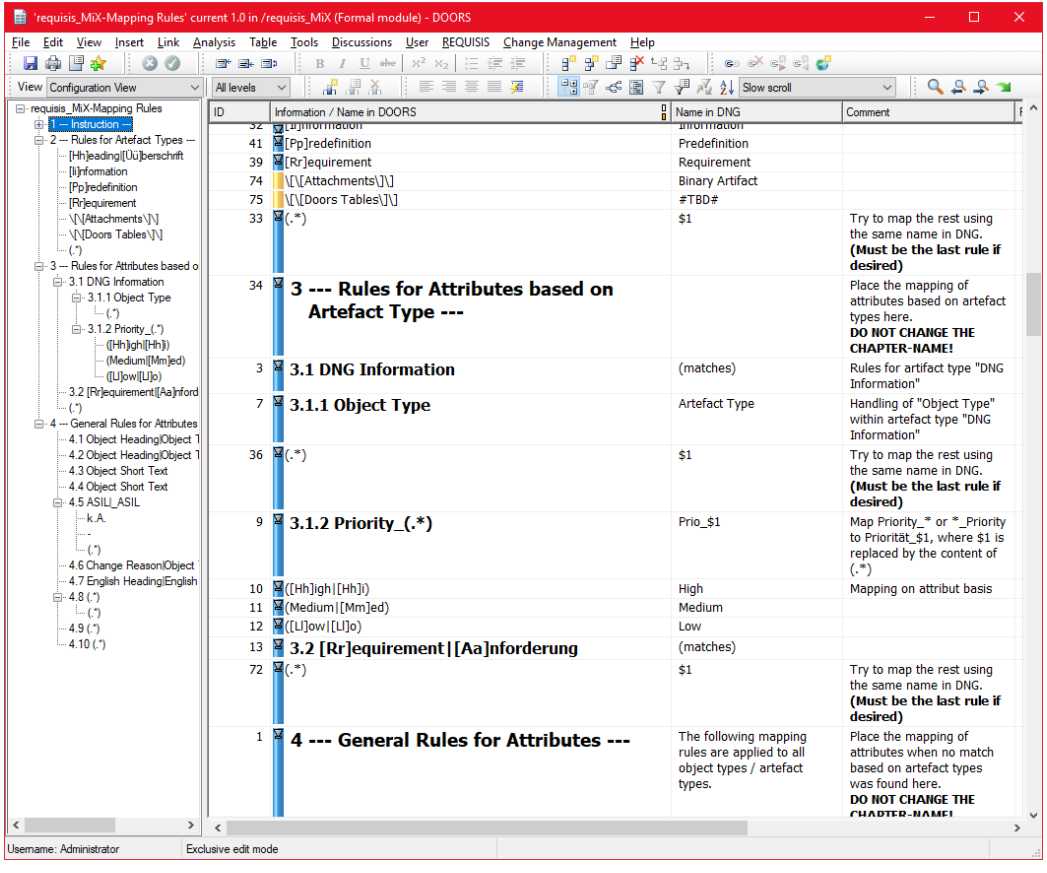
- Only manual effort to
  - Deal with custom attributes
    - that were to rare to be considered during the data analysis
  - Check findings of the pre-check
    - to decide if further steps are needed to prevent a data-loss

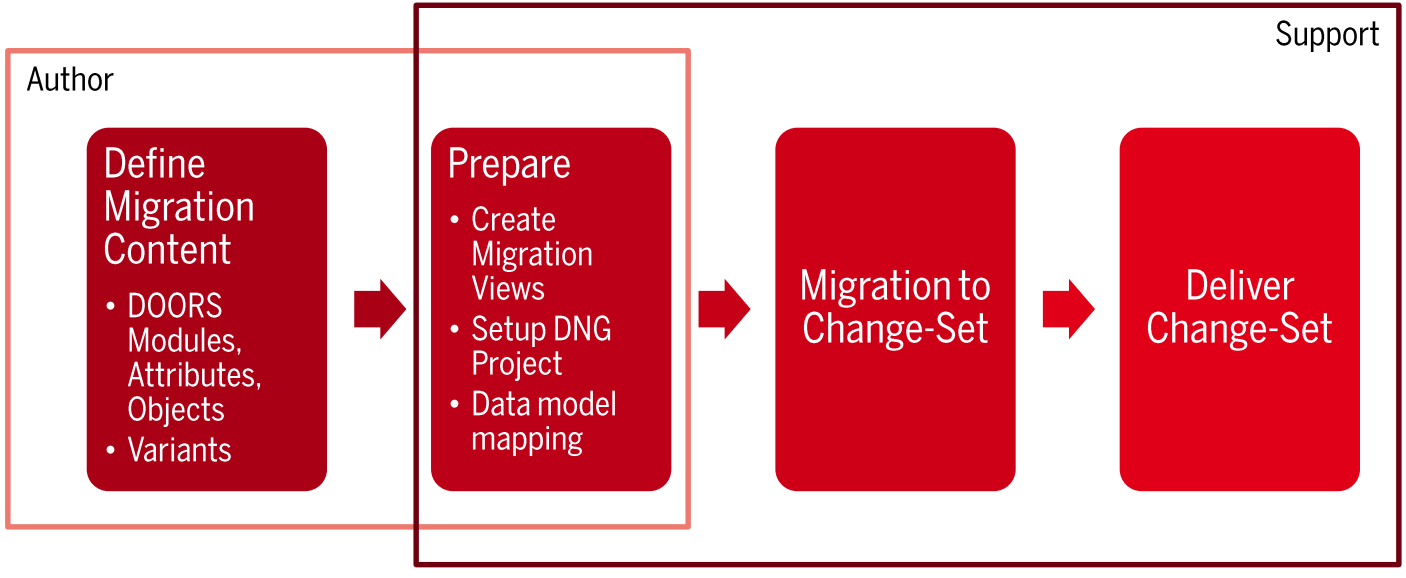
# Data analysis for defining the mapping rules



# Mapping Rules in requisis_MiX

- Mapping of Artifact Types
- Mapping of Attributes
  - specific for Artifact Type
  - independent of Artifact Type
- Mapping of Enum-Values for each Attribute-Rule
- Mapping uses regular expression
- Catch-All-Rules and substitution
- First hit wins






# Migration Workflow

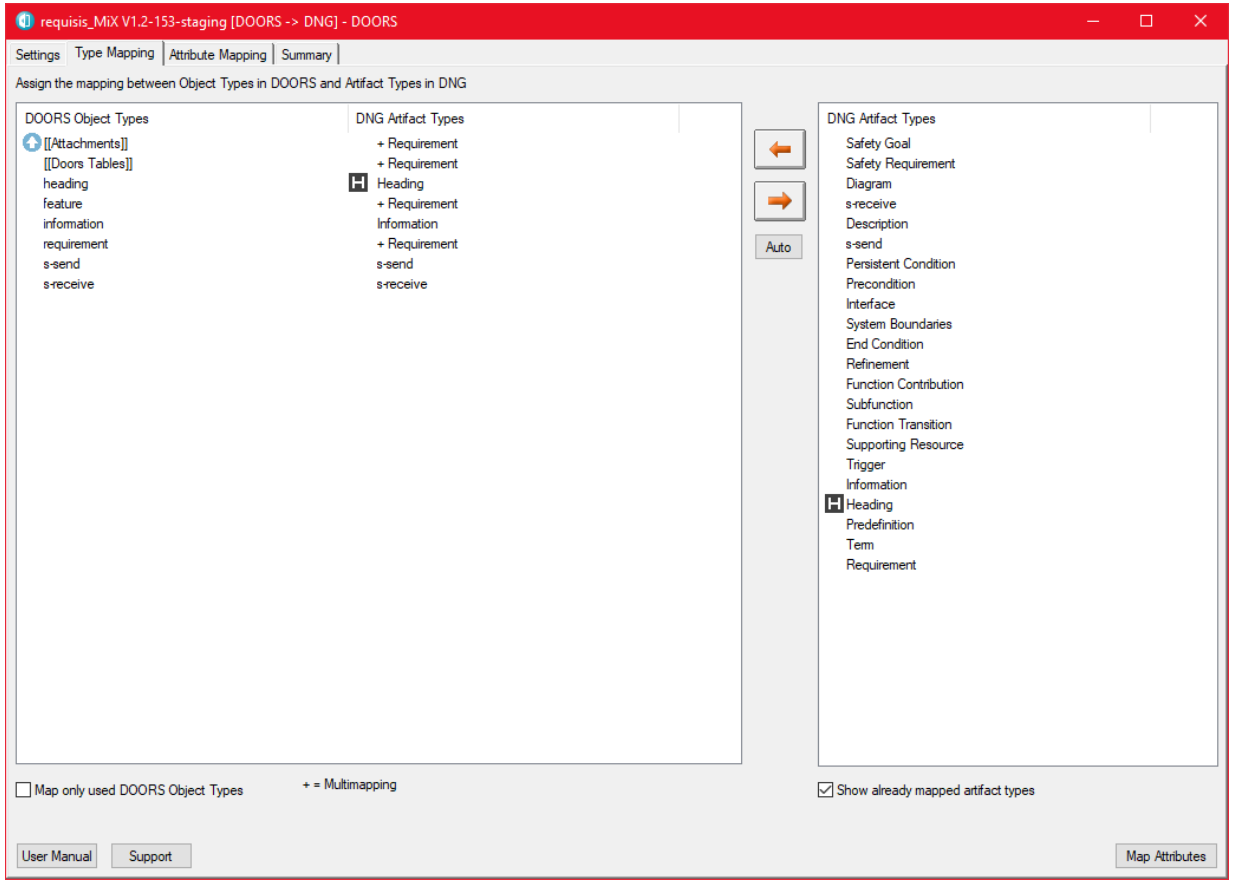
## How to conduct a migration

# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

## Prepare Modules and Mapping

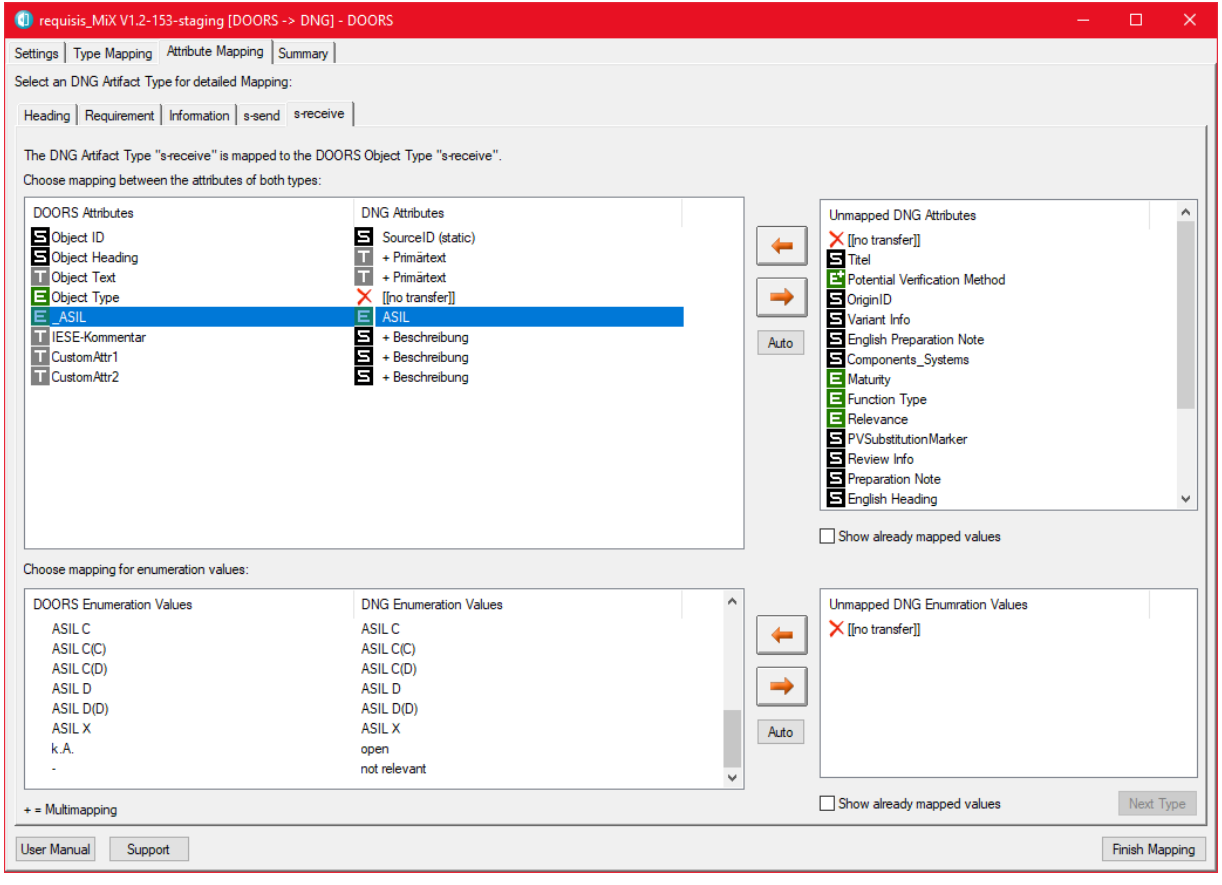
# requisis_MiX: Mapping of Artifact Types

- Artifact Types are read directly from DNG
- Mapping prefilled by mapping rules
- n:1 Mapping is possible
- Just drag and drop or use arrows to map



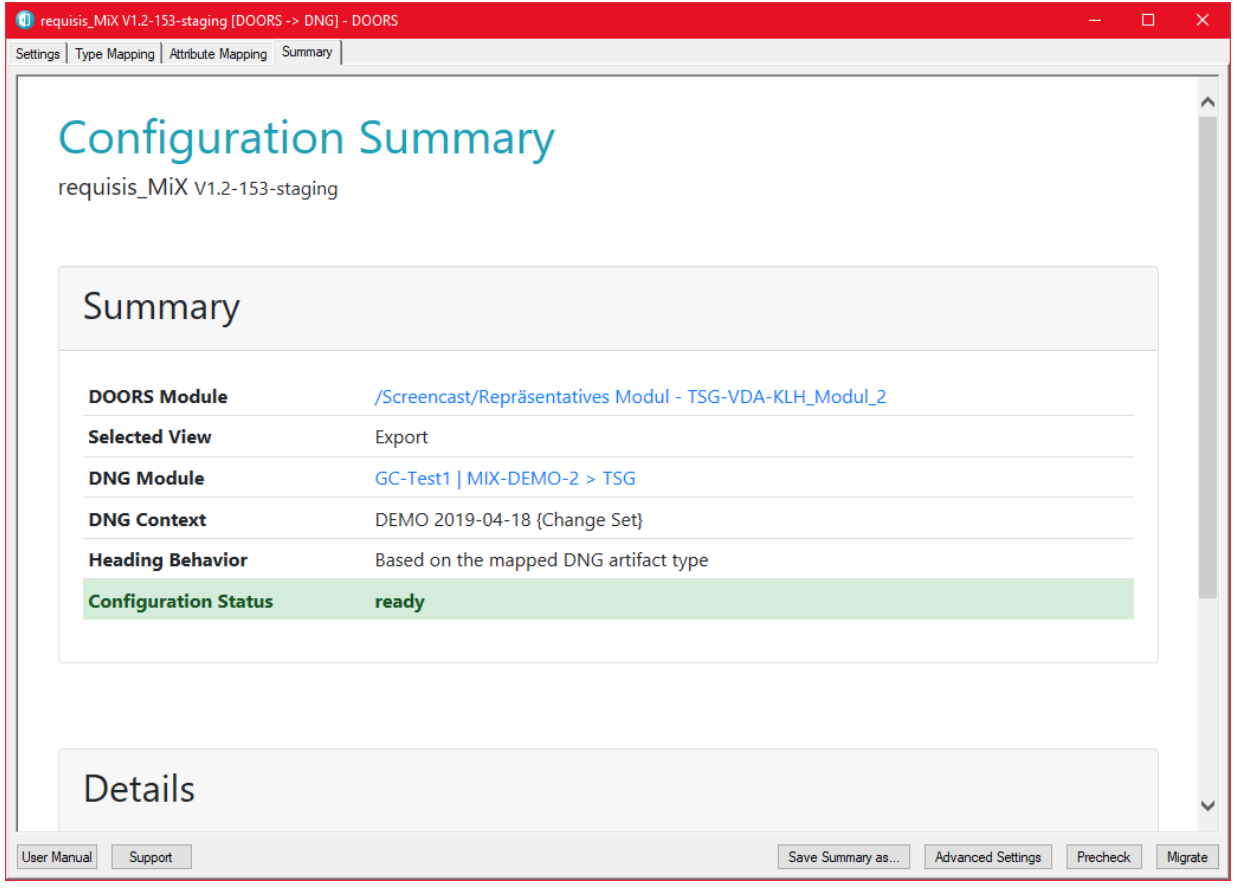
# requisis_MIX: Mapping of Attributes for each Artifact Type

- Attribute are read directly from DNG
- Mapping prefilled by mapping rules
- Mappings per Artifact type
- n:1 Mapping is possible
- Also mapping of Enumeration Values
- Just drag and drop or use arrows to map

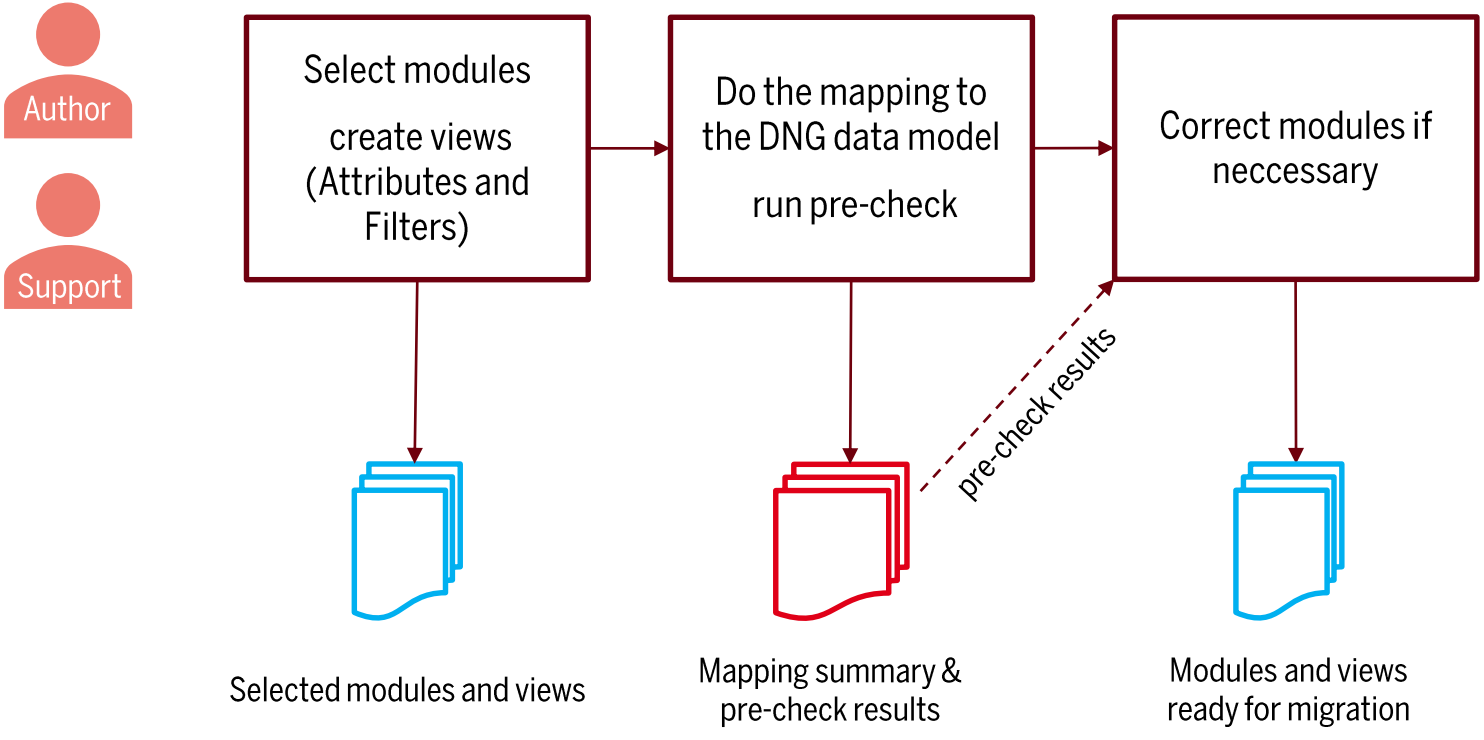


# requisis_MIX: Configuration Summary

- All relevant settings
- Including chosen Mappings
- Basic configuration check
  - Do all objects have a selected “object type”
  - Does the view contains a filter without “show ancestors”
- Can be saved to disk



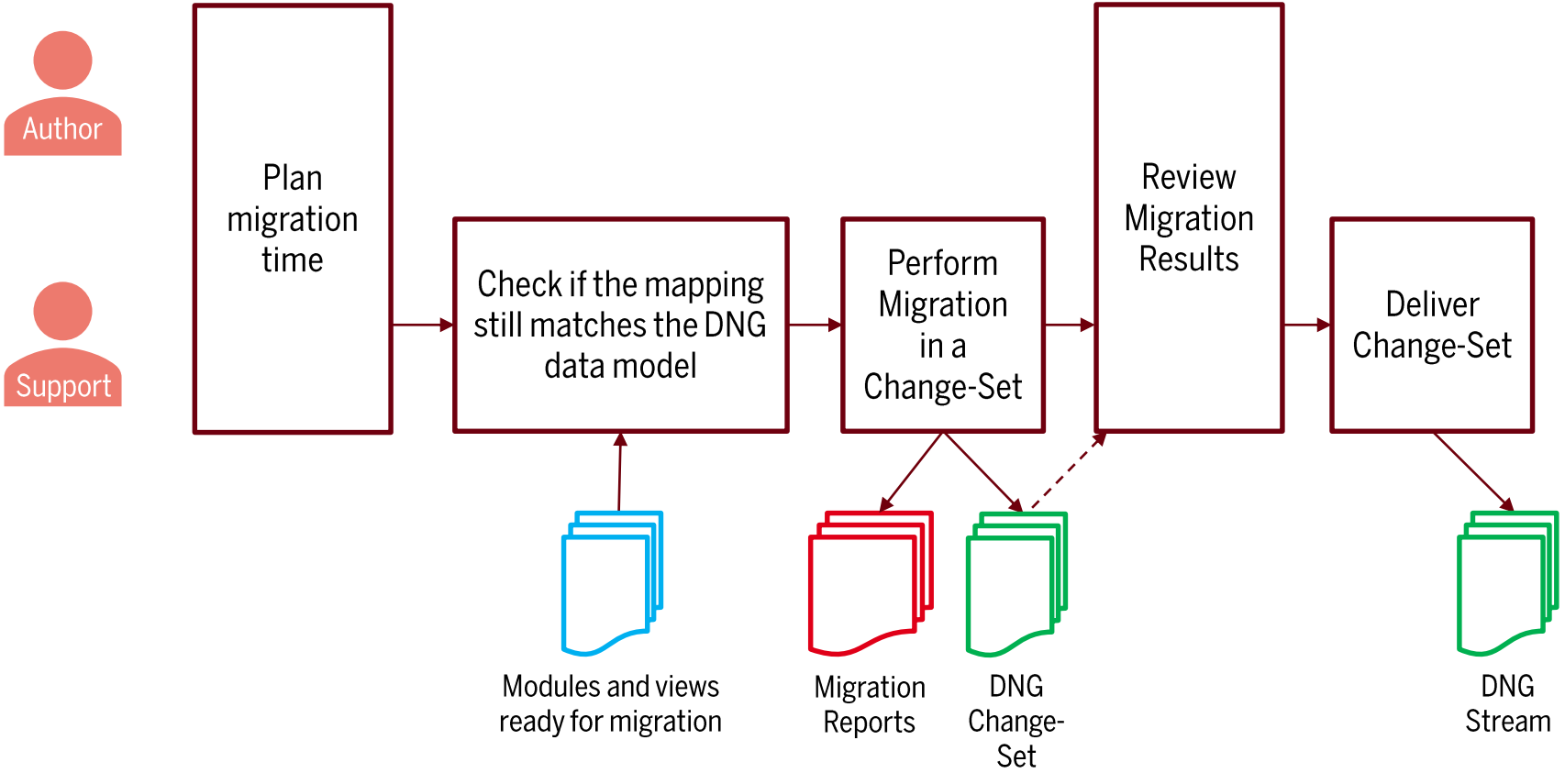
# Migration Preparation with requisis_MiX



# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

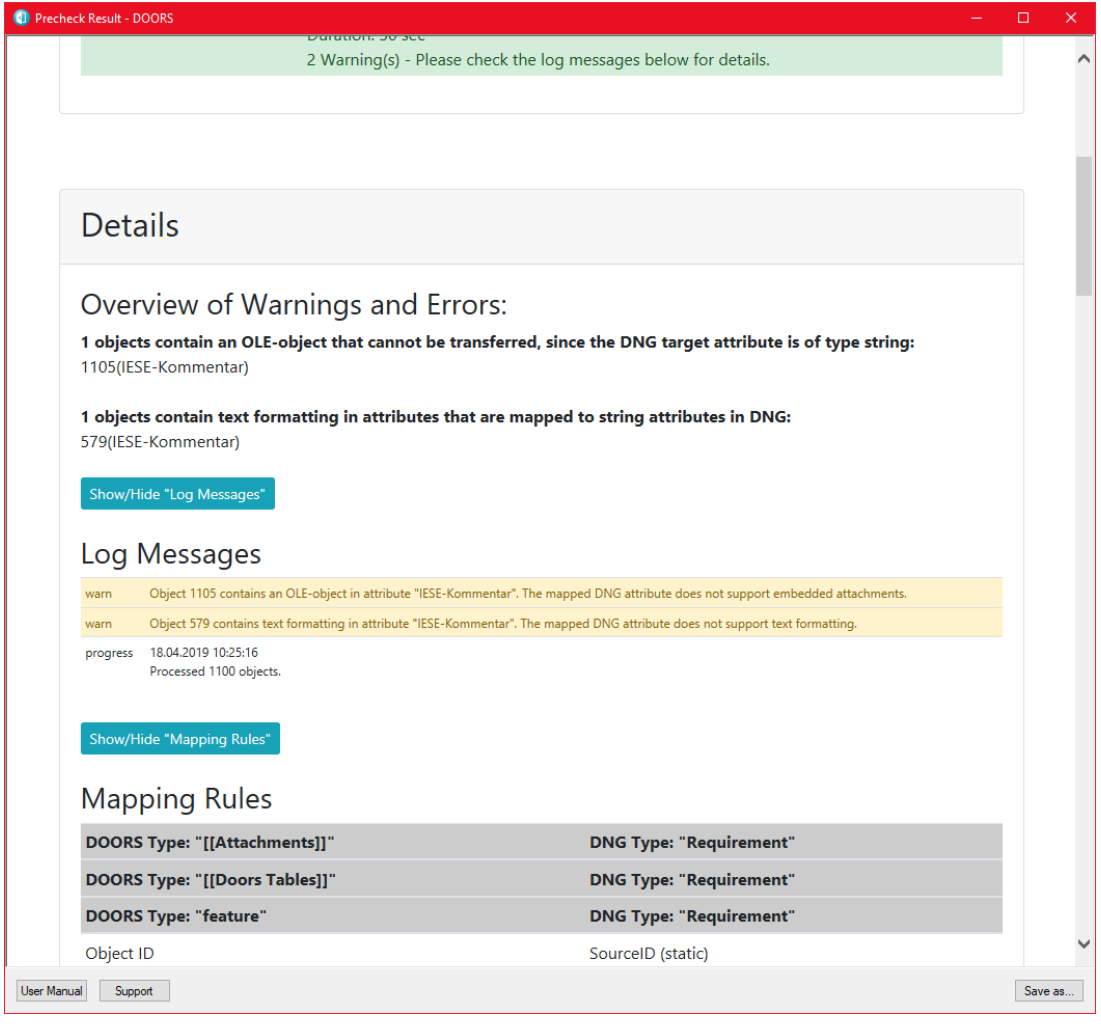
## Perform the Migration

# Example Workflow for Migration Execution



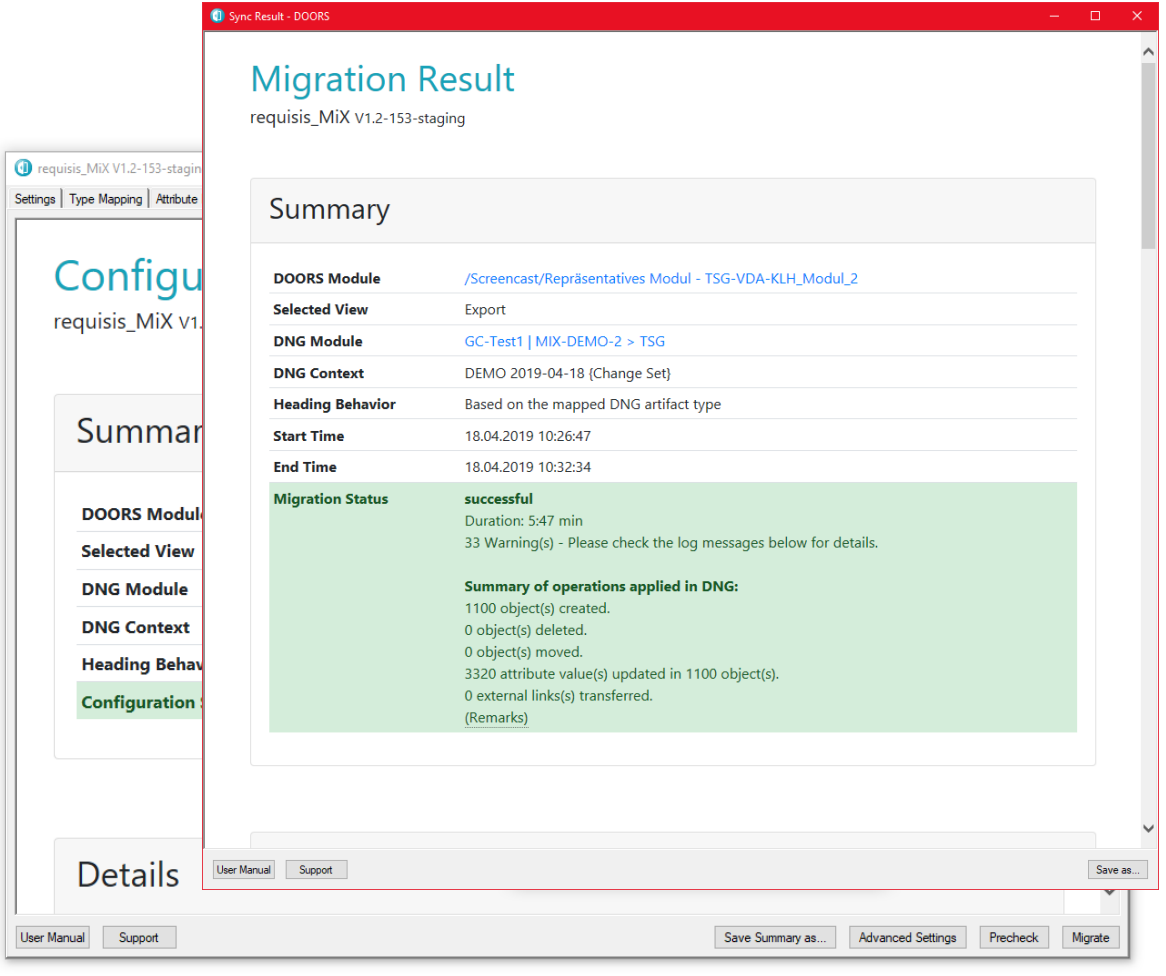
# requisis_MIX: Pre-check

- Pre-check for possible data loss caused by tool differences, like
  - OLEs in attributes mapped to other attributes than primary text
  - Text formatting in other attributes
  - Unsupported text formatting
- Can be saved to disk



# Perform the Migration

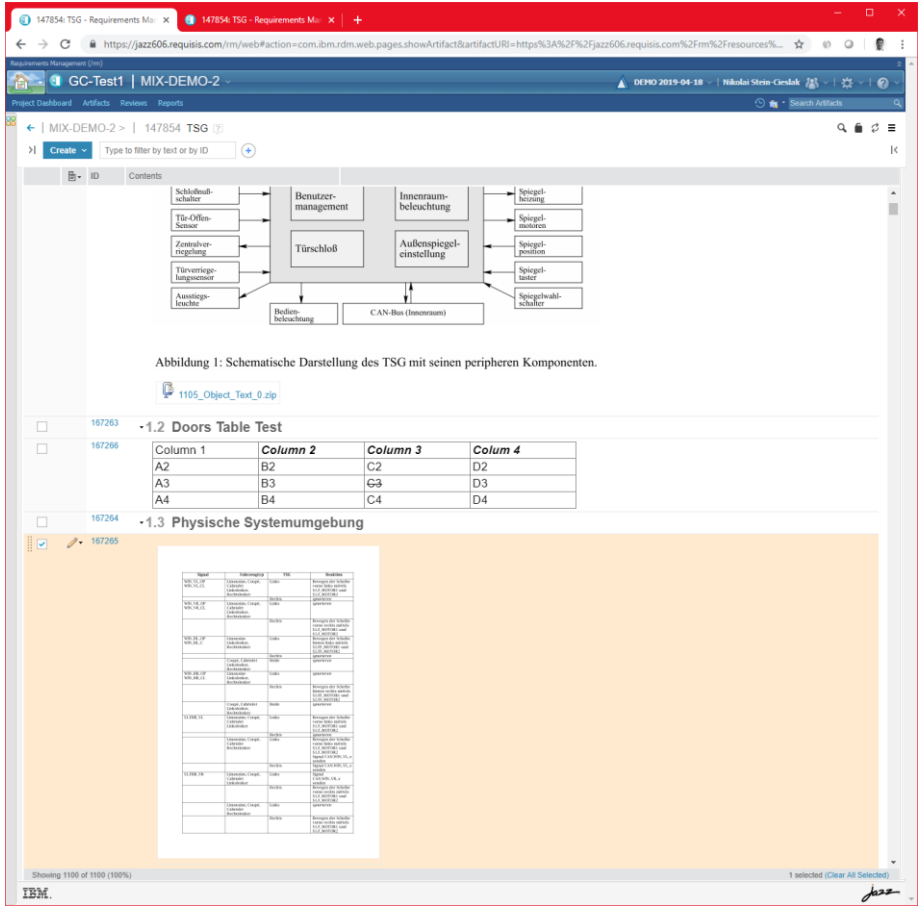
- Perform the Migration
- Migration Result
  - All relevant configuration settings
  - Information about data-loss
  - Migration Results
  - can be saved to disk



# Finishing the Migration

- Review the results in DNG
- Deliver change-set if satisfied with the result

- Lock-down Modules in DOORS if desired



# Updating Shadow-Modules

- **If data type model has changed**
  - Simply “close” the gaps by modifying the mapping
- **If data type model was not changed**
  - Just push on “migrate” or
  - Use dxl-Api to migrate
- **Automate as an scheduled task**
  - Execution in Batch-Mode is possible

# Other use-cases

- **Migration of Variants**
  - Multiple filtered Views into multiple Streams by re-using same artifacts

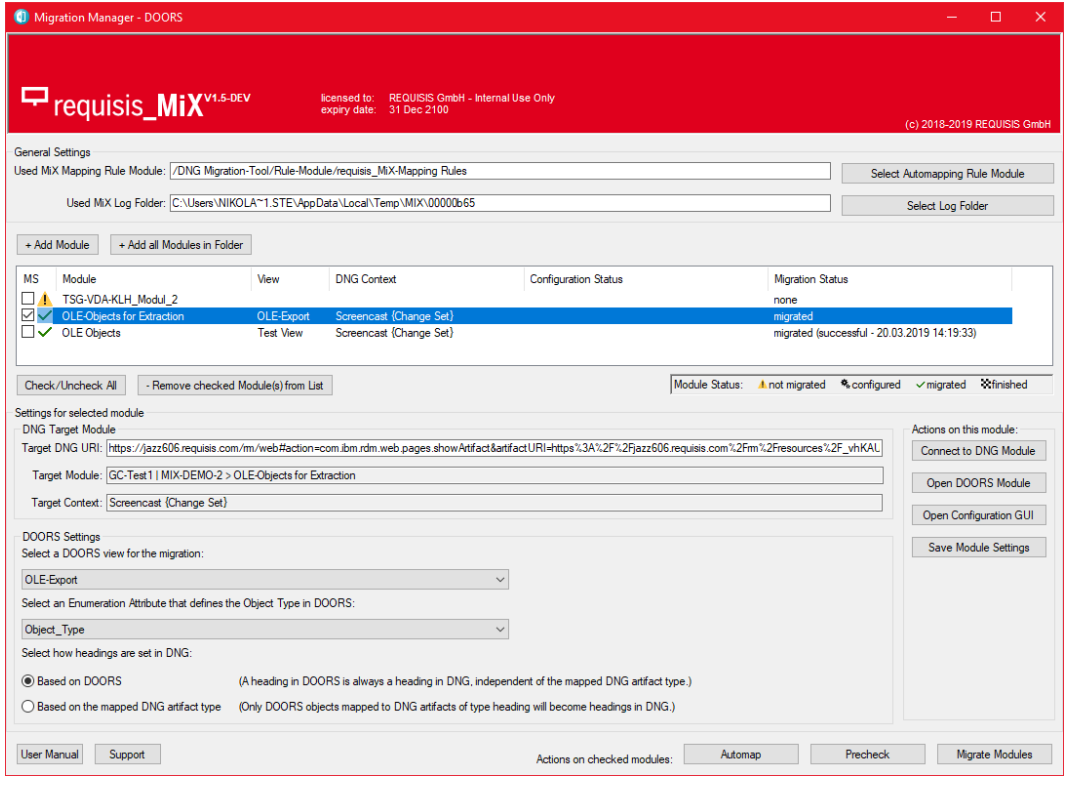
- **Advanced Update-Use-Cases**
  - Only new objects
  - Only some attributes
  - Only attribute values (no structure updates)

# Benefits of our approach and requisis_MiX

- **Reduce effort of migration:**
  - Time:
    - Very low manpower needed
    - No preprocessing of the DOORS data needed
    - Quick configuration due to automatic mapping
    - Very fast data transfer
  - Money:
    - No migration expert needed
- **Do-it-yourself-migration**
- **Quality:**
  - Clean DNG type model:
    - No modification of the DNG type model
    - All required transformations are based on mapping rules
  - Rule based migration:
    - Reduces configuration mistakes
    - Allows upscaling to large amounts of modules
  - Pre-check: Detect issues before migrating
  - Reusable Content:
    - Convert OLE-Objects into native format
    - Merge legacy attributes to collector attributes
    - Migrate into different base types if needed (e.g. enumeration to string)
  - Repeatable Migration:
    - Updates and Re-Dos with just one click

# Future work for requisis_MiX

- **Migration Manager (version 1.5)**
  - Configure multiple-modules at a time
  - Keep oversight over migration status
  - Run multiple migrations at once
- **Sync DNG->DOORS (version 2.0)**
  - Synchronization back to DOORS 9
  - Shadow-Copies to DOORS 9
  - Exchange-Processes
  - Complete Round-trips



# Clean and sustainable migration from DOORS 9 to DOORS Next Generation

## Other Tools

# Other Tools allowing Map, transform & migrate

- **Individual Standard**
  - Mapping based on ReqIF-Files
  - Transformation of ReqIF-Files according to preconfigured Mapping on Known RDF-URIs
  - Workflow: Export from DOORS, Transform files, Import in DNG
- **Opshub**
  - Syncing between DOORS and DNG
  - No details publicly available
- **IBM & Persistent tools (not indented for customer use)**
  - No details publicly available
- **Rivernorth Solutions (used as part of their service)**
  - Custom migration tool for requirements and test cases
- **Any more?**


Thank you for your attention
If you would like to get this presentation, email me:
Nikolai Stein (nikolai.stein@requisis.com)
For more Information on requisis_MiX:
https://requisis.com/en/mix