

PaR – Processes as Requirements
for Jama

Version: 13. February 2021

Text & Cover-Design: ©2020-2021 Copyright Ralf Bürger

Offered for license under the Attribution-ShareAlike International (CC BY-SA 4.0) license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. Content referenced by footnotes has additional restrictions. By utilizing this “PaR – for Jama”, you acknowledge and agree that you have read and agree to be bound by the terms of this license and the terms of referenced content.

Publisher: Ralf Bürger
SSE – Systematic Software Engineering
Wilhelmstraße 24
45527 Hattingen
eMail@RalfBuerger.de

Print: epubli – a service of neopubli GmbH, Berlin

Community: <https://ProcessesasRequirements.info>

Acknowledgements: To the people and companies that support me:
Sascha Kobus & Frank Zielke (KoDeCs GmbH)

"First, don't be afraid. ...

Second, do what you think is right. ...

Finally, build a community.

No one does big things by themselves."

(President Obama, 18.May.2020)

Table of Contents

PARADE OF CHALLENGES SHOWING THE NEEDS	5
PARIS (PAR INFORMATION SYSTEM)	6
PARTOUT - TOOL FEATURES TO SATISFY THE NEEDS OF PAR	7
FEATURE 1: DEFINITION OF REQUIREMENT ITEM TYPES	7
FEATURE 2: IMPLEMENTATION OF THE PARIS MAP	8
FEATURE 3: EVALUATION OF PROJECT MATURITY.....	10
FEATURE 4: COMPLIANCE CHECKS BY STANDARDS COVERAGE.....	12
FEATURE 5: SUPPORT FOR PROCESS VERSIONS	13
FEATURE 6: REUSE OF REQUIREMENTS SETS	14
FEATURE 7: SYNCHRONIZATION OF REQUIREMENTS SETS.....	15
FEATURE 8: DEFINITION AND MANAGEMENT OF VARIABILITY	15
PARTIAL IMPORT, EXPORT, BACKUP	17

PaRade of challenges showing the needs

I have identified **5 challenges** from my coaching of large projects over the past few years. They all can be addressed by changing the way to deal with bulky standards and processes. Bringing these deeply into the projects makes them more intrinsic and natural.

From those challenges I derived needs for a methodical framework. These are outlined on **The Page** (see at the beginning of this document) that is also available as **The Slide**. The whole **PaR** framework is described in a nutshell in **The Booklet**. For more details **The Book** is available.

This methodical framework requires some tool features for realization in organizations. Most modern requirements management tools that are in use in those organizations have at least the basic features on board that are requested here. Nonetheless it is sometimes tricky to configure the tools correctly.

Implementing multiple regulatory standards and setting up a reusable corporate development project process is still a lot of work and needs to be configured correctly. Often it is good advice to get help, support, coaching and maybe also manpower for setup from the experts. **PaR** offers help from the community of experts as described on the website.

This document shows an exemplary implementation for the tool

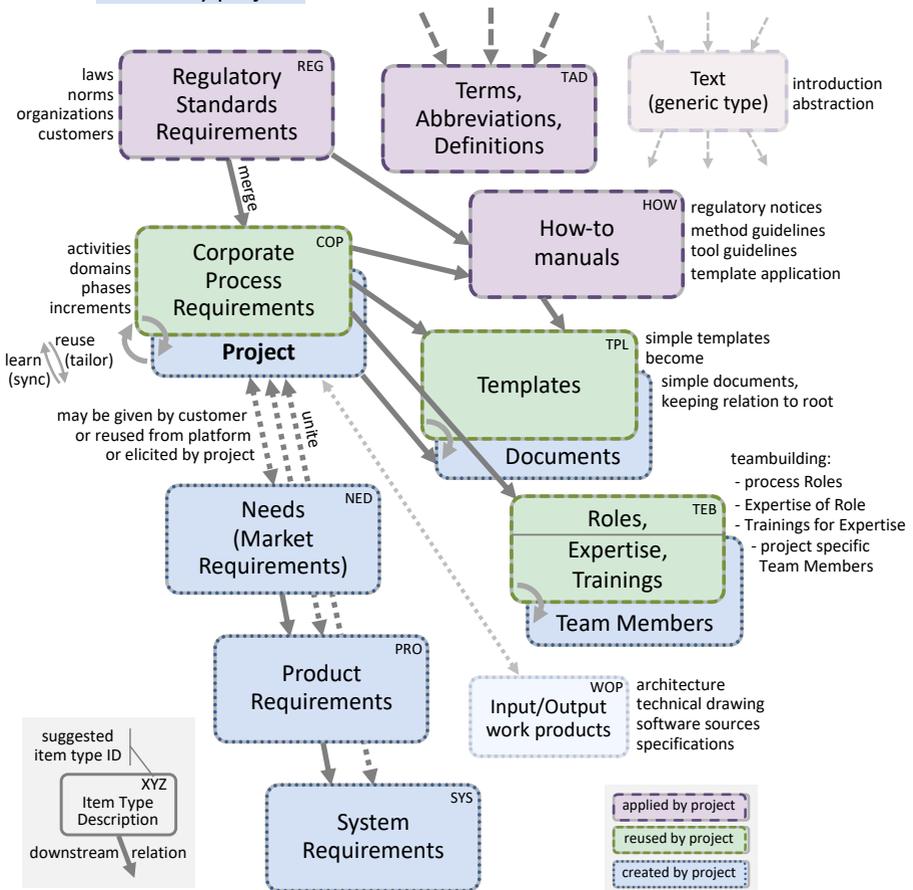
Jama Connect together with the **Jama** business partner **KoDeCs GmbH** – namely Sascha Kobus & Frank Zielke.



PaRis (PaR information system)

This PaR information system satisfies the discovered needs. It can be implemented in tools by a corresponding set of requirements item types with an item type relationship model. The PaRis is explained in **The Booklet** and **The Book**. We show it here only for quick lookup.

- Some requirements are simply applied by projects without change, for detailed lookup or guiding help.
- Other requirements are rather inputs to be reused by projects, also to be modified or extended.
- Reused items, including certain work products, finally become items that are created by projects.



PaRtout - Tool Features to Satisfy the Needs of PaR

This set of features defines what a tool should bring in to be able to fully support the methodical framework with the described PaRis. These features are explained in [The Booklet](#) and in more detail in [The Book](#).

Feature 1: Definition of requirement item types

In this **Jama PaR** implementation, we define the following item types according to the PaRis (shown in alphabetical order of the shortcut):

	K Corporate Process Requirement	K Corporate Process Requirements	K_COP	Corporate processes as requirements
	K How-to manual	K How-to manuals	K_HOW	Guiding team members in doing something step by step
	K Need	K Needs	K_NED	Needs (e.g. market needs or single customer expectations) lead to product requirements
	K Product Requirement	K Product Requirements	K_PRO	Requirements for the product derived from needs (customer requirements for suppliers)
	K Regulatory Standards Requirement	K Regulatory Standards Requirements	K_REG	Regulatory standards (e.g. ASPICE) and norms (e.g. ISO 9001) and customer standards as requirements
	K System Requirement	K System Requirements	K_SYS	Requirements for the system under development derived from product requirements
	K Team Building	K Team Building	K_TEB	Team building by roles, competencies, skills, authorities, etc.
	K Template	K Templates	K_TPL	For easing and standardizing operational work in projects
	K Term Abbr Def	K Terms Abbrs Defs	K_TAD	Terms, Abbreviations, Definitions

The named shortcuts are taken from the diagram, and the prefix “K_” has been added, because the implementation happened in a **Jama** cloud sandbox of the company **KoDeCs** as **Jama** business partner.

The “Text” item type is generic in **Jama** and needs not to be defined explicitly. We use only one item type K_TEB for Team Building and naming convention to separate Roles, Expertise, Trainings and Team Members. We do not use WOP items separately and link from the COP items to the work products directly, either by description or by http-Link.

The following screenshot shows the field definitions of the item type K_NED as an example, with a defined picklist for the variation point types, with defined extra fields for different views (Situations) and with the possible configurations for “Required” (mandatory fields), “Suspect” (dirty-flag for changes) and “Sync” for reuse-synchronization.

The field `legacy_ID` is used for roundtrip ReqIF exports. For this purpose it must be a Text Field, it must be Read-Only, but it must “Allow API Overwrite”. This field must be added to every item type, including Text, Component, Set and Folder. Then a GUID is created with first export that can be used for later roundtrips (import/export) to synchronize between multiple systems. For this mechanism a separate DataExchange tool with an own license is needed.

Item type fields											✎	↑	↓	✖	➕ Add field
Order	Label	Unique Field Name	Field Type	Control	Pick List	Read Only	Allow API Overwrite	Required	Suspect	Sync	API-ID				
1	Project ID	documentKey	Text Field			✓		✓			1361				
2	Global ID	globalid	Text Field			✓		✓			1364				
3	Name	name	Text Field					✓	✓	✓	1367				
4	Variation Point	variation_point	Pick List		K Variation Point			✓	✓	✓	1454				
5	Technical Situation	technical_situation	Text Box	Rich Text							1433				
6	User Situation	user_situation	Text Box	Rich Text							1436				
7	Environmental Situation	environmental_situation	Text Box	Rich Text							1439				
8	Recognized Need	need	Text Field					✓	✓	✓	1442				
9	Optional Epic Description	description	Text Box	Rich Text							1370				
10	Release	release	Release						✓		1376				
11	legacy_ID	legacy_ID	Text Field			✓	✓				1614				

Feature 2: Implementation of the PaRis map

The following relationship diagram shows the example PaRis map implementation in **Jama**, based on the item types described in the previous chapter.

The diagram shows the relationships only, while hierarchies of the same item types are always possible. Because we use the TEB for roles, expertise, trainings and team members (assigned persons in project), we need a relation here on itself. If we would define separate item types for roles, expertise, trainings and team members, then they could get dedicate relations to each other, which makes it more sophisticated, but also more complicate. We also need a relation to itself for the REG, because the automotive IATF 16949 details the ISO 9001.

Each relation can be defined individually with a relationship type and is then automatically visualized in the diagram. For each relation **Jama** can define, whether missing coverage shall be signaled in list views.

Note the option for any type that is not mentioned in the rule set to be related to any other type. Since TPL and TAD and the generic Text are not mentioned they can be related to any other type. Thereby any item may have explanations for Terms, Abbreviations and Definitions, and also any item may have a related Template or explanatory Text.

K Processes as Requirements [Applies to KoDeCs PaR Process Platform ×, KoDeCs PaR Project "Setup BMS Platform" ×, & 1 more... - Hide All ↑ Add project]

All Projects

KoDeCs PaR Process Platform × KoDeCs PaR Project "Setup BMS Platform" × KoDeCs PaR Product Platform "BMS" ×

Add Project

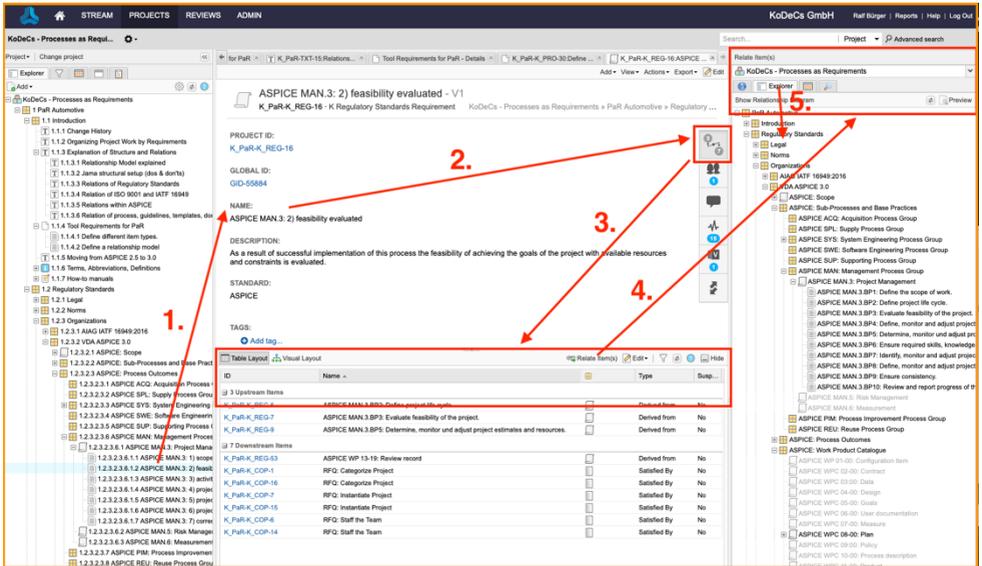
Upstream	Downstream	Relationship Type	For Coverage
<input type="checkbox"/> K Regulatory Standards Requirement	→ <input type="checkbox"/> K Regulatory Standards Requirement	Derived from	No
<input type="checkbox"/> K Regulatory Standards Requirement	→ <input type="checkbox"/> K Corporate Process Requirement	Satisfied By	Yes
<input type="checkbox"/> K Corporate Process Requirement	→ <input type="checkbox"/> K Team Building	Allocated to	Yes
<input type="checkbox"/> K Team Building	→ <input type="checkbox"/> K Team Building	Related to	No
<input type="checkbox"/> K Need	→ <input type="checkbox"/> K Product Requirement	Satisfied By	Yes
<input type="checkbox"/> K Corporate Process Requirement	→ <input type="checkbox"/> K Product Requirement	Caused by	No
<input type="checkbox"/> K Product Requirement	→ <input type="checkbox"/> K System Requirement	Satisfied By	Yes
<input type="checkbox"/> K Corporate Process Requirement	→ <input type="checkbox"/> K How-to manual	Related to	No
<input type="checkbox"/> K Corporate Process Requirement	→ <input type="checkbox"/> K Need	Caused by	No
<input type="checkbox"/> K Corporate Process Requirement	→ <input type="checkbox"/> K System Requirement	Caused by	No
<input type="checkbox"/> K Regulatory Standards Requirement	→ <input type="checkbox"/> K Team Building	Satisfied By	No

[Add relationship](#)

Allow any item type not mentioned in this rule set to be related to any item type including those in the rule set.

When creating items in **Jama**, then the relationship model is considered, i.e. only modeled relations are possible, and upstream relations are separated from downstream relations. The following example window shows how comfortable relations can be defined.

When an item is selected in the project tree (1.), then it is displayed with all fields in the middle, together with the widgets (2.). With a click on a widget the corresponding information is displayed below, in our case the relations (3.). We see that upstream and downstream relations are separated, both with bi-directional traceability. For relating an item according to the relationship model a history or tree can be selected (4.). From the tree (that can also show another **Jama** requirements project) a double-click is enough to define a new relation.



Feature 3: Evaluation of project maturity

The following screenshot shows in a list view that in the COP item type we define a field “Status in Project”, that is ignored when the process is designed, but filled, when the process is reused in the project.

This field has a picklist and covers also the statuses “to be determined” and “not applicable”. The former shows what is already planned or still open. The latter shows what has been tailored by the project. This can be used for coverage analysis as well as for progress analysis.

The list view also gives a good overview of “effort estimated” for the already planned process steps and “effort needed” for the ongoing or closed process steps. This can be checked by filters and views for consistency also.

Often in projects the planning and the doing happens in different tools by different roles, e.g. in Microsoft Project by the Project Manager and in IBM Rational DOORS by the Systems Engineer (or Requirements Manager or Feature Owners). This often is also the reason for inconsistency, also in progress, e.g. when doing happens before planning. Bringing the process into the core project work tool unites planning with doing and progress.

KoDeCs PaR Project "Se..."

STREAM PROJECTS REVIEWS ADMIN KoDeCs GmbH Ralf Burger | Reports | Help | Log Out

Project: Change project

Learn more Dashboard: KoDeCs PaR Project "Se..." Corporate Processes

Corporate Processes View

ID	Name	Status in Project	Budget estimated (€)	Budget needed (€)	Set Key	Effort estimated (hrs)	Effort needed (hrs)
BMS-CP-19	RFQ						
BMS-SET-35	RFQ: Project Organization				K_COP		
BMS-CP-39	RFQ: Nominate Project Leader	done	300	200		3	2
BMS-CP-40	RFQ: Staff the Team	ongoing	1600	300		16	3
BMS-CP-41	RFQ: Instantiate Project	done	100	100		1	1
BMS-SET-50	Platform definition				K_COP		
BMS-CP-58	Perform brainstorming sessions	ongoing	3200	0		32	0
BMS-CP-59	Report to management	open	400	0		4	0
BMS-CP-60	Align feedback with scoping	open	800	0		8	0
BMS-CP-61	Document the platform approach	open	1600	0		32	0
BMS-SET-36	RFQ: Project Scoping				K_COP		
BMS-CP-42	RFQ: Obtain the needs for the new product or product update	completed					
BMS-CP-43	RFQ: Commonly understand the needs	ongoing					
BMS-CP-44	RFQ: Define the purpose and scope of the product as...	open					
BMS-CP-45	RFQ: Categorize Project	to be determined					
BMS-SET-37	RFQ: Requirements Elicitation				K_COP		
BMS-CP-46	RFQ: Obtain or define the product requirements.	ongoing					
BMS-CP-47	RFQ: Cluster product requirements to features	ongoing					
BMS-CP-53	RFQ: Nominate owners for the features.	to be determined					
BMS-CP-48	RFQ: Establish bidirectional traceability of needs and...	ongoing					
BMS-SET-49	RFQ: Requirements Analysis				K_COP		
BMS-CP-54	RFQ: Derive first system requirements from the prod...	ongoing	4000	200		40	2
BMS-CP-55	for feature "Thermal Control"	open					
BMS-CP-56	for feature "AC charging"	open					
BMS-CP-57	for feature "DC charging"	open					
BMS-CP-58	for feature "Low voltage support"	open					
		not applicable			K_COP		
					K_TEB		

Page 1 of 2

KoDeCs PaR Project "Setup BMS Platform" View

Action status overview

Action status Status in Project

to be determined open ongoing completed done not applicable

Action status: to be determined

ID	Name	Status in Project	Effort
BMS-CP-45	RFQ: Categorize Project	to be determined	
BMS-CP-43	RFQ: Nominate owners for the featu...	to be determined	

Displaying items 1 - 2 of 2

Missing estimations

ID	Name	Status in Project	Effort
BMS-CP-42	RFQ: Obtain the needs for the new...	completed	
BMS-CP-43	RFQ: Commonly understand the ne...	ongoing	
BMS-CP-44	RFQ: Define the purpose and scope...	open	
BMS-CP-46	RFQ: Obtain or define the product re...	ongoing	
BMS-CP-47	RFQ: Cluster product requirements...	ongoing	
BMS-CP-48	RFQ: Establish bidirectional traceab...	ongoing	
BMS-CP-55	for feature "Thermal Control"	open	
BMS-CP-56	for feature "AC charging"	open	
BMS-CP-57	for feature "DC charging"	open	
BMS-CP-62	for feature "Low voltage support"	open	

Displaying items 1 - 10 of 10

Unused roles (team members missing?)

ID	Name	Description
BMS-TEB-42	Role: System Engineer	Expert for the product as t
BMS-TEB-43	Role: Student	a helping hand

Displaying items 1 - 2 of 2

Action status: not applicable

In **Jama** filters can be combined with list views (as shown above), but also with sections in the dashboard (as shown beside).

Here we see a pie chart as status overview, a list of process activities still to be determined, process activities with progress and without estimations, and roles where no person is assigned yet. Those filters, diagrams and lists can be used in many ways to get a good impression of the real project maturity.

Feature 4: Compliance checks by standards coverage

The previous chapter already showed how filters, lists and diagrams can be used in **Jama** for compliance also, with the example of tailored process activities or missing role actors.

As also mentioned earlier in the relationship model of **Jama** the relations can be declared for coverage monitoring (see screenshot further up). This can be used in compliance features like the “trace view”. The following screenshot shows for example the compliance of the corporate process (on the right side) to ISO 26262, defined as regulatory standards requirements (shown on the left side). We see that coverage for some aspects is still missing, so that a project reusing this corporate process can also not be compliant to the regulatory standard.

SOURCE ITEMS			1 LEVEL DOWN		
K Regulatory Standards Requirement (56)			K Corporate Process Requirement (26)		
Project ID	Name		Project ID	Name	Explanation
K_PaR-K_REG-154	ISO 26262-2: 5.4.2.1 sustain a safety culture		K_PaR-K_COP-1	RFQ: Categorize Project	Categorize the project into A, B or ...
			BMSP-K_COP-45	RFQ: Categorize Project	Categorize the project into A, B or ...
			K_PaR-K_COP-38	RFQ: Ensure Safety Manager aut...	If functional safety according to IS...
			BMSP-K_COP-65	RFQ: Ensure Safety Manager aut...	If functional safety according to IS...
K_PaR-K_REG-155	ISO 26262-2: 5.4.2.2 maintain rules and processes for...		K_PaR-K_COP-1	RFQ: Categorize Project	Categorize the project into A, B or ...
			BMSP-K_COP-45	RFQ: Categorize Project	Categorize the project into A, B or ...
K_PaR-K_REG-128	ISO 26262-2: 5.4.2.3 maintain processes to communi...			Add Coverage for K_PaR-K_REG-128	
K_PaR-K_REG-156	ISO 26262-2: 5.4.2.4 maintain a process to resolve an...			Add Coverage for K_PaR-K_REG-156	
K_PaR-K_REG-157	ISO 26262-2: 5.4.2.5 perform required functional safet...			Add Coverage for K_PaR-K_REG-157	
K_PaR-K_REG-158	ISO 26262-2: 5.4.2.6 provide the required resources		K_PaR-K_COP-38	RFQ: Ensure Safety Manager aut...	If functional safety according to IS...
			BMSP-K_COP-65	RFQ: Ensure Safety Manager aut...	If functional safety according to IS...

In the same way we can check compliance of a project to the corporate process: When the project deletes items instead of setting the status to “not applicable” then we see the missing coverage. In this “trace view” in **Jama** we can also filter on item types and configure the columns to be shown.

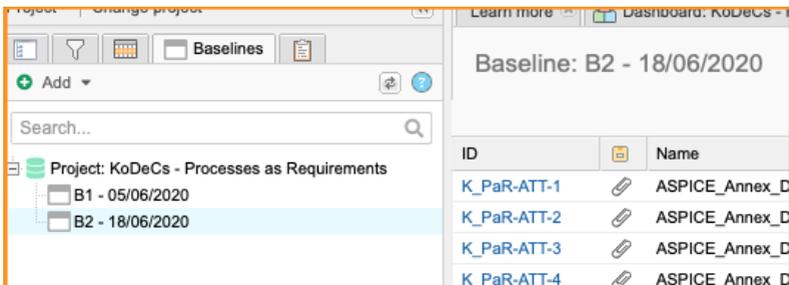
In addition, **Jama** offers a “Coverage Explorer” that can be configured in a hierarchical fashion over multiple levels. For example, the following screenshot shows the corporate process (COP) activities, that have been reused in a project (“BMSP-...” for Battery Management System Platform), with the relations to the teambuilding (TEB) items, which are the assigned

roles in the process definition. The next level shows the relation to teambuilding again, because we use this item type for different purposes (as already explained further up). So, we see where trainings and expertise are assigned from the process definition, but we also see where a “Team Member” is assigned to a role taking this process activity in the project. Here the Coverage Explorer is used to check for compliance of the project against the defined corporate process. Quality guys will love it ;-)

Component: RFQ		Level 1 Downstream		Level 2 Downstream	
ID	Name	ID	Name	ID	Name
BMSP-CMP-18	RFQ	---	---	---	---
BMSP-SET-35	RFQ: Project Organization				
BMSP-K_COP-39	RFQ: Nominate Project Leader	BMSP-K_TEB-60	Role: Program Manager	BMSP-K_TEB-72	Expertise: Interfacing with Parties
				BMSP-K_TEB-79	Team Member: Sascha Kobus
				BMSP-K_TEB-68	Expertise: Driving a Plan
				BMSP-K_TEB-67	Expertise: Driving the Process
				BMSP-K_TEB-72	Expertise: Interfacing with Parties
				BMSP-K_TEB-69	Expertise: Organizational Talent
				BMSP-K_TEB-81	Team Member: Ralf Bürger
				BMSP-K_TEB-75	Training: Product Owner
				BMSP-K_TEB-76	Training: Scrum Master
				BMSP-K_TEB-68	Expertise: Driving a Plan
				BMSP-K_TEB-67	Expertise: Driving the Process
BMSP-K_COP-40	RFQ: Staff the Team	BMSP-K_TEB-61	Role: Project Leader (Product Owner)		
BMSP-K_COP-65	RFQ: Ensure Safety Manager authority	BMSP-K_TEB-61	Role: Project Leader (Product Owner)		

Feature 5: Support for process versions

In **Jama** baselines can be created, but they can be used only to compare them with the current version or older baselines or to reset current items from a baseline. It is not possible to reuse baselines. Therefore a copy or export of a baselined process version needs to be done to be able to further apply the copy in a project or to re-import the earlier export. **Jama** currently is working on a branching/merging feature on top of the baseline feature that enables branching at some time and later merge that branch into a project.



Feature 6: Reuse of requirements sets

In **Jama** it is possible to reuse requirements sets on every level, also with a synchronization functionality, also applicable for the relations, tags, links, etc. The "external" items can either stay or also be reused (mirrored) and also an automatic addition of a relation to the reuse source is possible. This functionality will get even better with upcoming branching/merging and variability management functionalities. Nonetheless it is difficult to synchronize later added relations, because either all relations are reused (and thereby most of them duplicated) or none (and the new ones are missing). So, for bulk work this is good, for dedicated changes it is often easier to repeat them manually.

In **Jama** the reuse functionality only works with the current process version. It would help to support older versions or baselines or branches also, but this is yet to come.

Feature 7: Synchronization of requirements sets

In **Jama** synchronization of reused items is possible, also for relations, but existing relations are then duplicated. So, we prefer to synchronize without relations to focus on the content. If relations have been added on the source, this must be repeated manually on the destination.

• reuse the complete Corporate Standards process (e.g. "Automotive Supplier Process") and rename it to become the project
• assign the people to the roles

DIRECTIONS:

Number of Synchronized items: 1

ID	Global ID	Name	Project	Sync Status
K_PaR-K_COP-15	GID-56127	RFQ: Instantiate Project	KoDeCs - Processes as Requireme...	(Current Item)
K_PaR-K_COP-7	GID-56127	RFQ: Instantiate Project	KoDeCs - Processes as Requireme...	Out Of Sync

Synchronize changes to this item

Feature 8: Definition and management of variability

Jama does not yet support variability management for variant creation and platform setup. Upon the current baseline concept, a later version shall provide a branching/merging functionality and then a further later version will use this concept for branching on variation points to merge multiple choices into a final product or process variant.

Currently only a specific field can be applied and requirements with sub-requirements can show the variability.

We called the field simply "Variation Point"; it should be defined for the product requirements (NED, PRO, SYS) for product platform and for COP for process platform. For the picklist of the field we defined "none", "optional" as opposite of mandatory, "alternative" to select one, "choice" for n out of m. More about platforms as methodical approach in **The Book**.

After reuse the not-needed requirements must be deleted manually, tool support is currently not available. Developing a plugin here or API based functionality does not make any sense, because **Jama** is working on a build-in feature.

To support finding the variation points quickly a tag can be declared, added to the items and used for filtering. The following screenshots show this rather manual approach for variability management in **Jama**.

The screenshot displays the KoDeCs interface for managing requirements. On the left, an Explorer pane shows a hierarchical tree of requirements. A specific requirement, '1.4.1.2.3.1 BMS: The battery shall be chargeable via AC up to x kW.', is highlighted in red. Below the tree, a 'Variation Point(2)' dialog is open. On the right, the main workspace shows the selected requirement: 'BMS: The battery shall be chargeable via AC up to x kW. - V4'. The text 'x kW' is highlighted in red. Below the requirement text, a 'VARIATION POINT: alternative' dialog is open, showing a list of options: 'none', 'optional', 'alternative', and 'choice'. The 'alternative' option is selected. Below this, there are fields for 'RELEASE:' and 'TAGS:', with 'Variation Point' entered in the tags field.

For other tools (e.g. IBM Rational DOORS) external tool support is available to manage variability, e.g. by Biglever with Gears or by pure-systems with pure::variants. This option is not yet available for **Jama**.

Even when **Jama** has built a feature for variability management it would make sense to get support by those tools to ensure traceability of variation points from requirements over architecture down to the software code, and then up the V cycle to support different binding times. Our long years of experience in variability management for platform-based product development showed that either an external tool has to be linked into the complete tool landscape or a holistic tool suite with built-in variability management must be chosen, which does not exist yet.

PaRtial Import, Export, Backup

For migrating processes from existing process design tools to the requirements engineering tool it is essential to have some import options. Then all processes and sub-processes can be migrated step by step, and finally maybe the expensive licenses for the process design tools can be saved (the requirements engineering tools are needed anyway).

Some architectural design tools can work quite good with requirements. Therefore, it makes sense to also transfer process requirements to those tools. This requires functionalities for partial exports.

At least it makes sense to perform partial backups from time to time. Of course, the complete databases and clouds are saved regularly by central tool administrators of the IT departments, but saving a process release now and then should be possible.

The screenshot displays the 'Data Exchange for Jama Software' application window. The main interface is titled 'DATA EXCHANGE' and includes a 'jama software' logo. The 'Export' tab is active, showing the following configuration:

- Purpose: Exchange Export Only
- ReqIF Format: Jama ReqIF [STANDARD]
- Export From: Location Filter Baseline
- Project: KoDeCs - Processes as Requirements [48] (with a 'Select' button)
- Location: Processes as Requirements (PaR) [10760] (with an 'Include in Export' checkbox checked)
- Mapping File: PaR_ReqIF_Jama_Export.json (with 'Remove' and 'Edit' buttons)
- Export to: /Users/rabu/Documents/SSE/Kontakte/KoDeCs/Process (with a 'Select' button)
- An orange 'Export' button is at the bottom.

On the right side, there is a log window titled 'Error Logs' and 'Execution Logs'. The 'Execution Logs' section shows a successful export process:

```
Export started at 2020-06-11T07:53:49.916Z
Retrieving mapping configurations...
Successfully retrieved mapping configurations
Fetching location data for item ID: 10760
Fetching data from project: KoDeCs - Processes as Requirements
Fetching Items
Fetching Relationships
Fetching Release Info
Fetching Filter Data
Fetching Baselines
Serializing ReqIF content...
Successfully extracted attachment [/Users/rabu/Documents/SSE/Kont...
ReqIF export complete at 2020-06-11T07:54:04.332Z. Elapsed Time
```

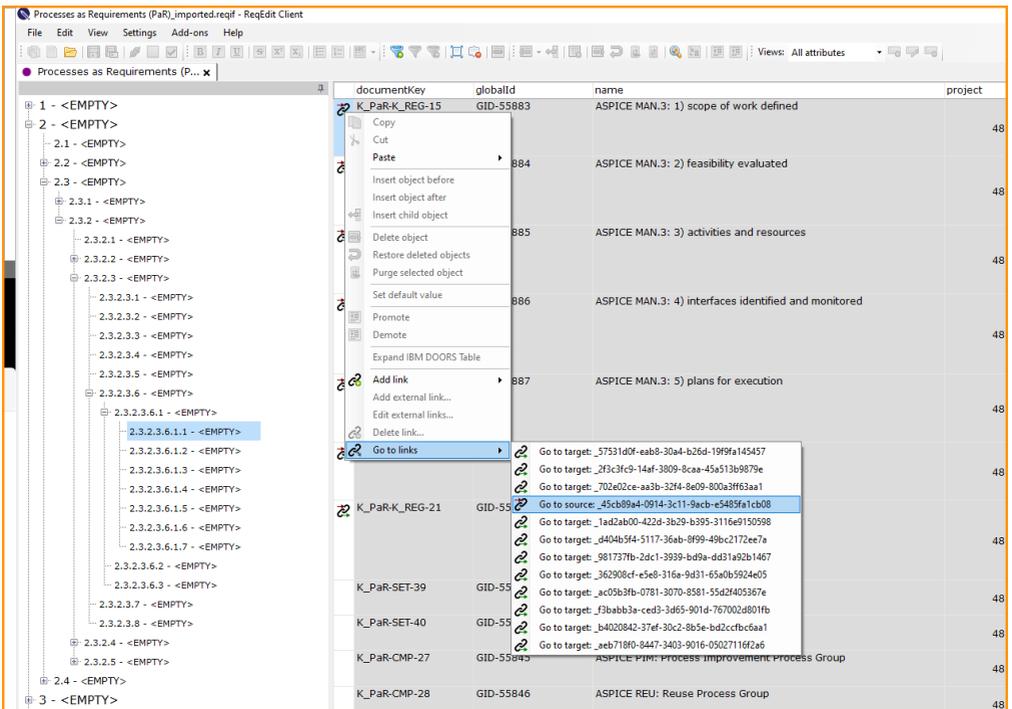
Below the main interface, a 'Data Exchange Result' dialog box is open, titled 'Successful Exchange'. It contains a table with the following data:

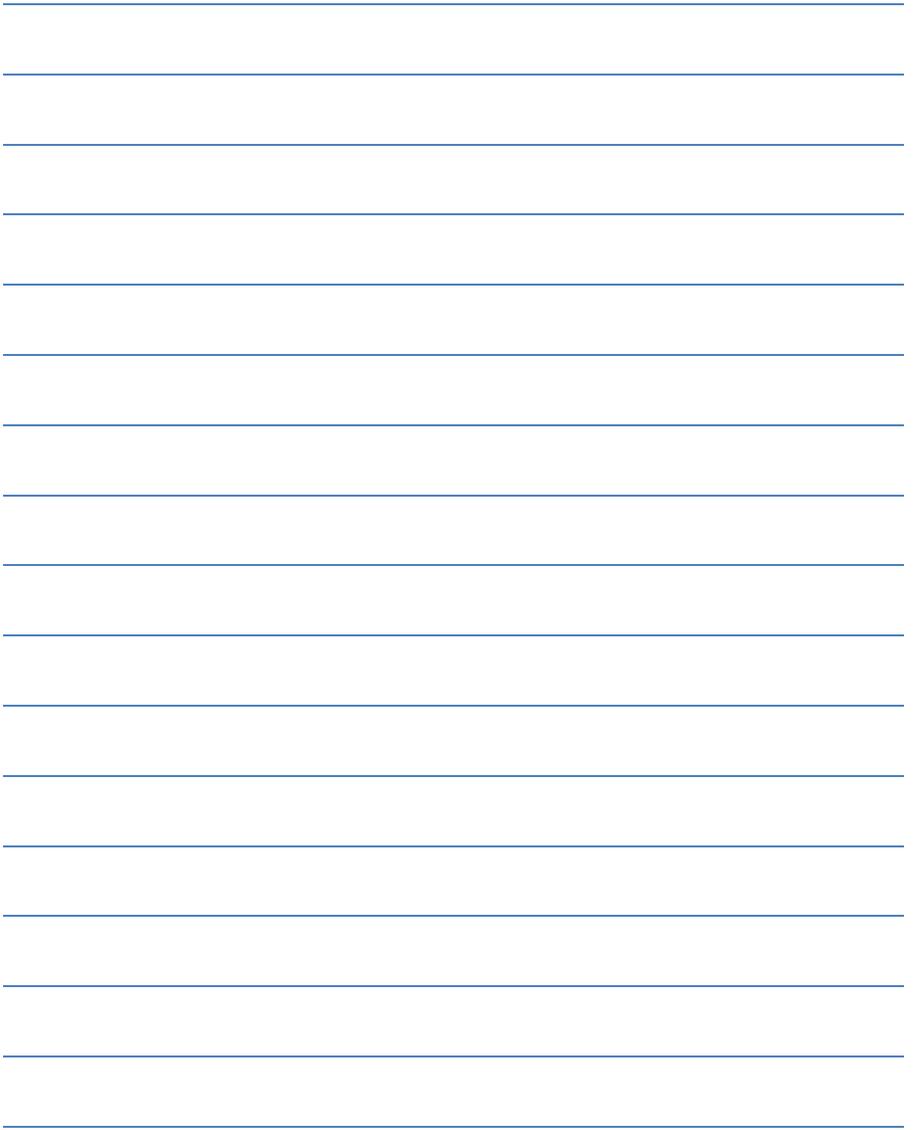
	Detected	Converted	Exported	
Items	386	386	386	
Specifications	1	1	1	
Relationships	195	195	195	
		Downloaded	Failed	
Attachments	7	1		

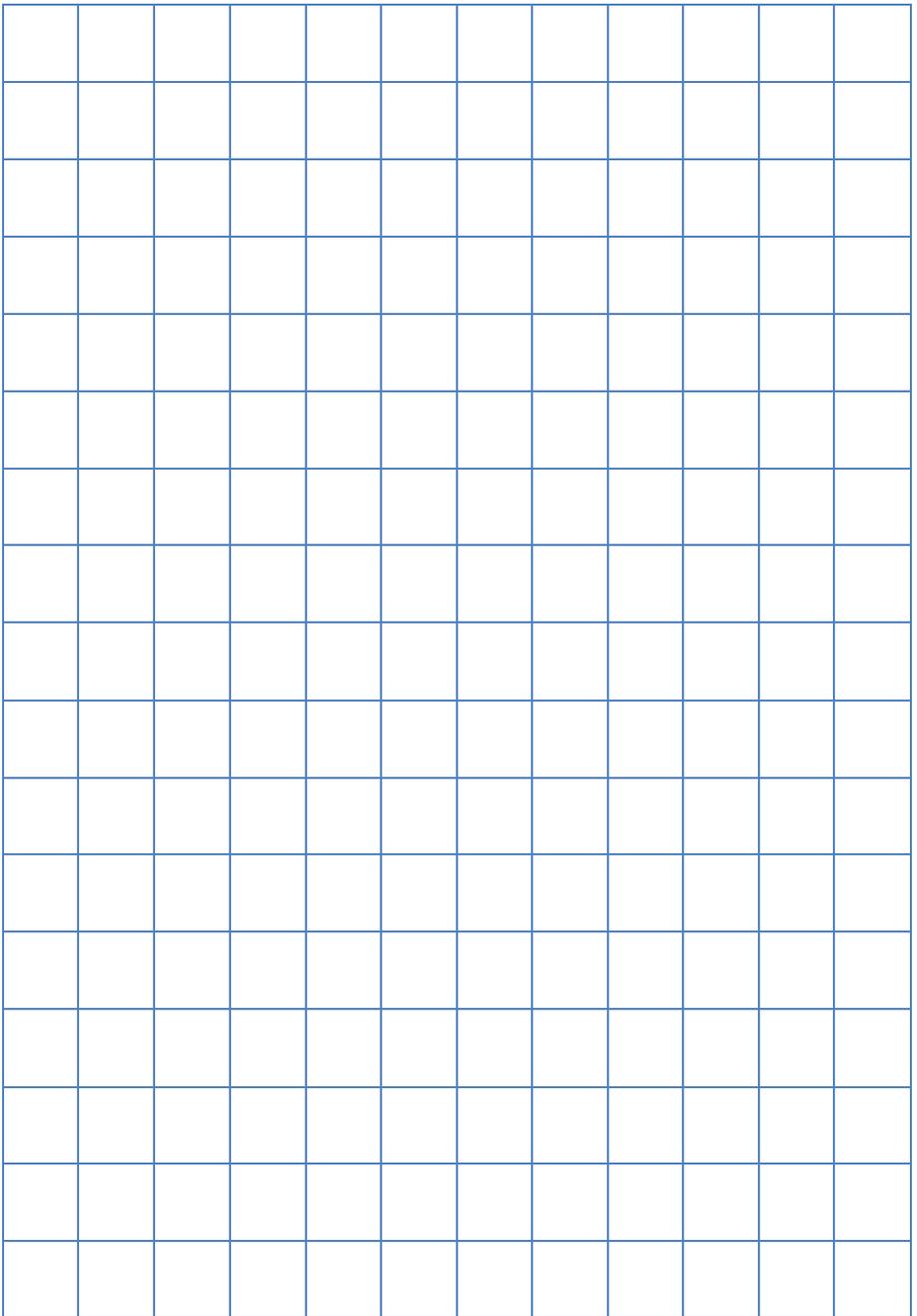
The dialog box also features 'Save' and 'OK' buttons at the bottom right.

Jama offers a separate DataExchange module (with separate expensive license also), that can import and export also ReqIF format. This can be used for Import, Export and Roundtrip, and thereby also for Backup. For Roundtrip also see the description of the item type definition at the beginning of this document. Also check the manual of DataExchange and the hints in the online **Jama** Community.

The **Jama** screenshot above shows a ReqIF export run for Backup purposes. The screenshot below shows a proof of the export with the ReqTeam ReqEdit viewer – even the relations are exported by **Jama** Data Exchange, what makes a really comprehensive backup.







... From the graphical representation in 3 levels we can derive an organizational responsibility that is not that visible today in many of our departments. Setting up the regulatory standards – i.e. the boundary conditions – separately from the processes and then relating it to each other is something we don't do that clear today. But that would be much more efficient also for sure. ...

Central process department, a German automotive supplier

... I'm convinced that PaR is the next step to be more efficient and agile in project even though you have to fulfil A-SPICE, ISO 26262 and ISO 21434. ...

Sascha Kobus, CEO KoDeCs GmbH

... Very promising approach, which exploits the reuse potential for product and process aspects in a unified manner. ...

Dr. Martin Becker, Department Head Embedded Systems Engineering at Fraunhofer Institute for Experimental Software Engineering (IESE)

... The variant development process for architecture is among the best we have seen. We consider this approach to be state of the art and benchmark. Especially the strong link between platform and project architecture ...

Feedback from an expert discussion of a process for platform-based product development, that I created over some years for a German automotive supplier. My basic platform ideas of that process also made their way into the **PaR** approach for process platforms.



Did your processes become a heavyweight backpack to be carried by the projects, rather than a lightweight intrinsic approach that really helps the teams to navigate through the storms of the projects?

It gets better when you design regulatory standards and **Processes as Requirements** that are reused in and improved by the projects.

Ralf.Buerger@ ProcessesAsRequirements.info
<https://ProcessesAsRequirements.info>