



3DEXPERIENCE®

3DEXPERIENCE Knowledgeware Overview



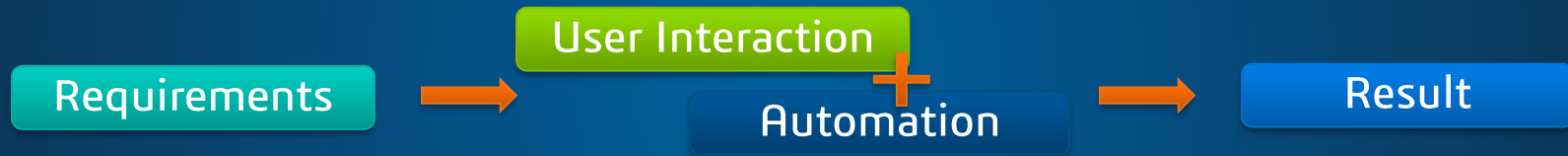
3DEXPERIENCE® CONFERENCE
DESIGN, MODELING & SIMULATION
NOVEMBER 19-21, 2019 | DARMSTADT, GERMANY

November 2019

Peter STUEHN
CATIA Mechanical Systems
Modeling & Simulate Centre of Excellence

Introduction | Knowledge Based Engineering

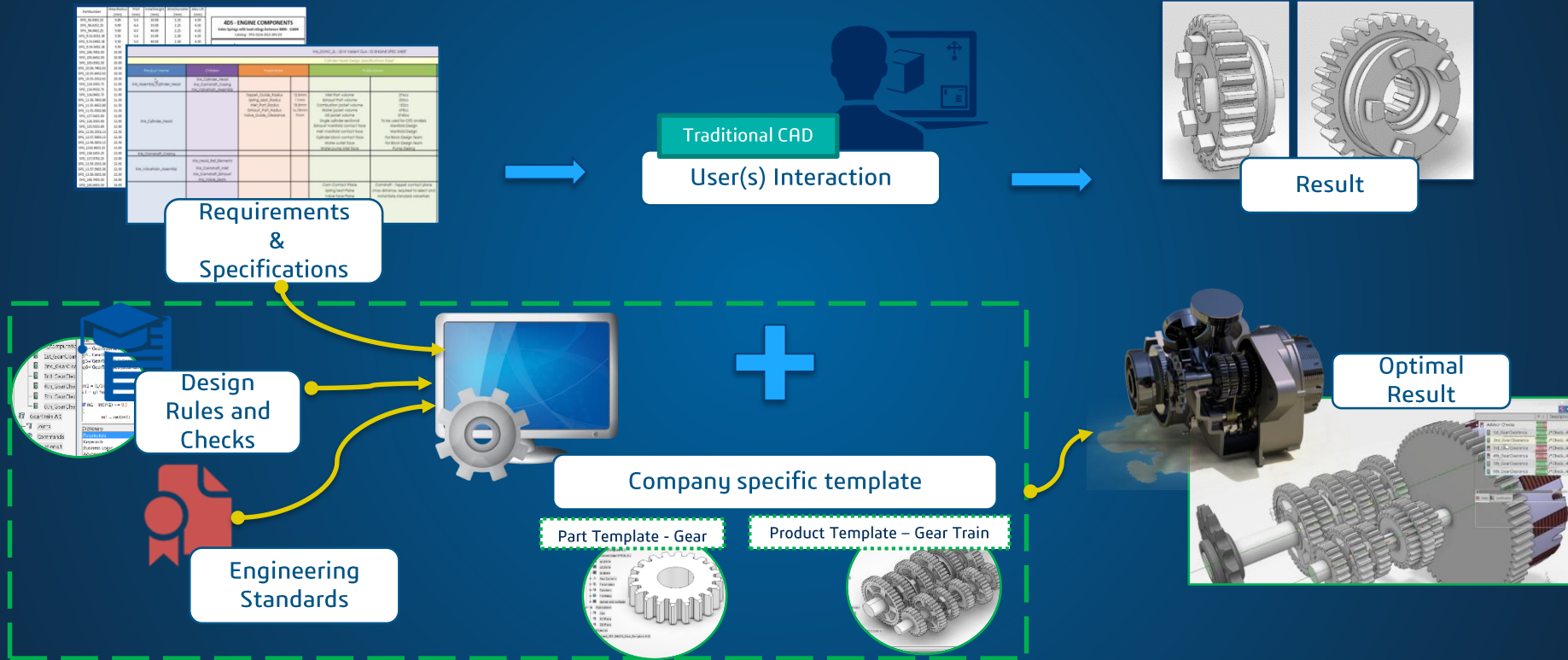
- ▶ Knowledge Based Engineering is the ability to create a design taking into account a company's existing knowledge, but also company standards or international rules.
- ▶ This knowledge is captured and formalized, so that it is directly available to an engineer with a certain level of automation, to be directly beneficial to the user.



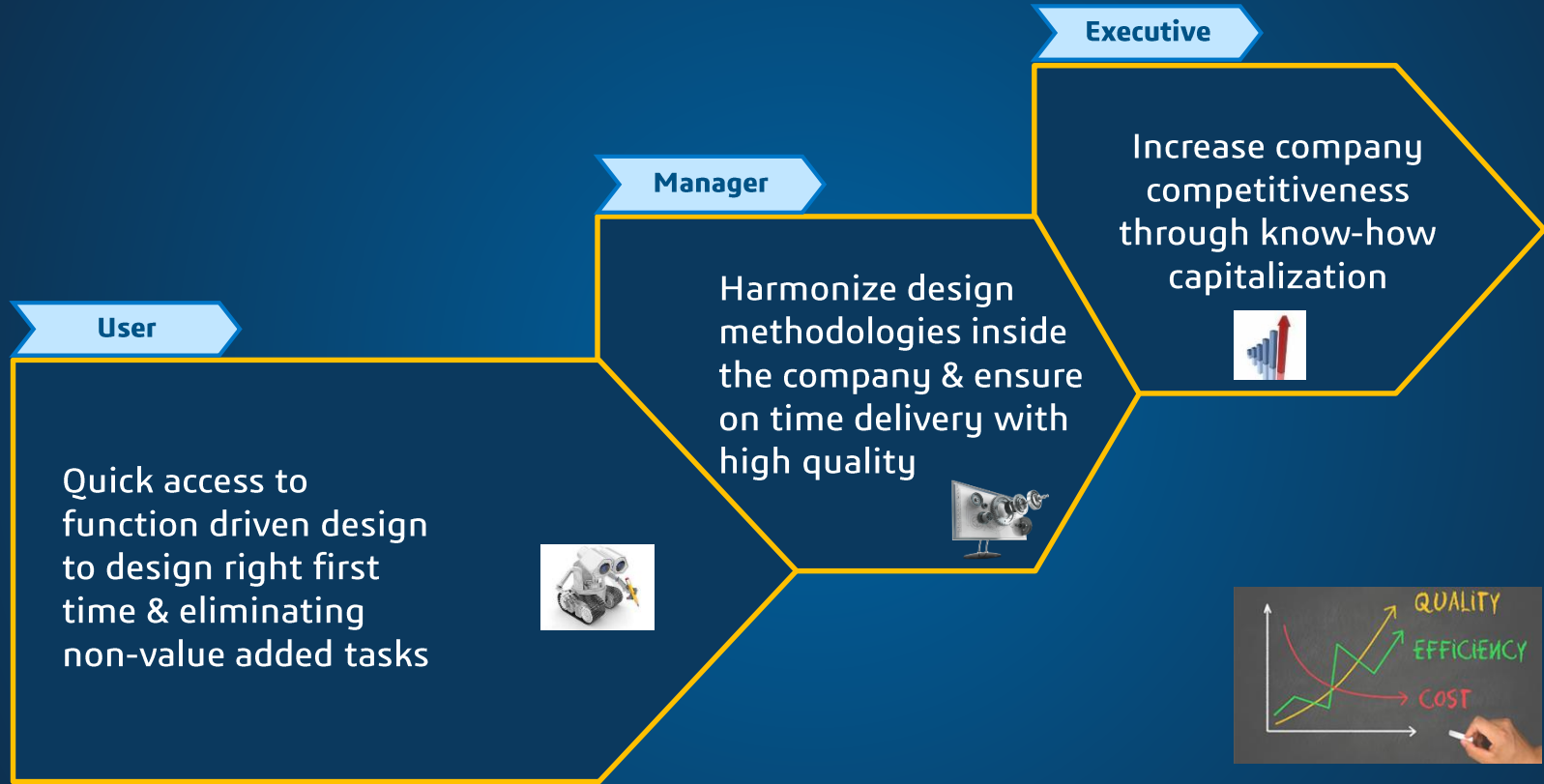
Challenges faced in successfully utilizing company knowledge:

- ▶ To achieve a robust and efficient engineering process
- ▶ To maintain, manage and share this knowledge across all stakeholders and programs

What is Knowledge Based Engineering with CATIA ?



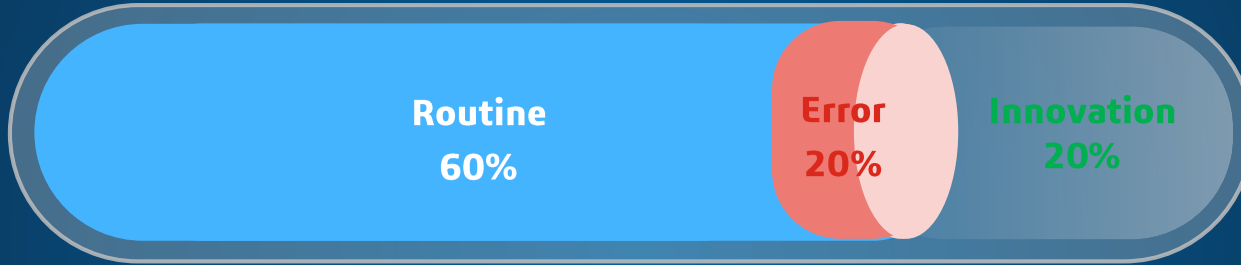
Knowledge Based Engineering | Who benefits?



CATIA Know-How Capitalization | Impact

Knowledge Based Engineering (KBE) is key technology to enhance competitiveness

Traditional
engineering
approach



KBE
approach

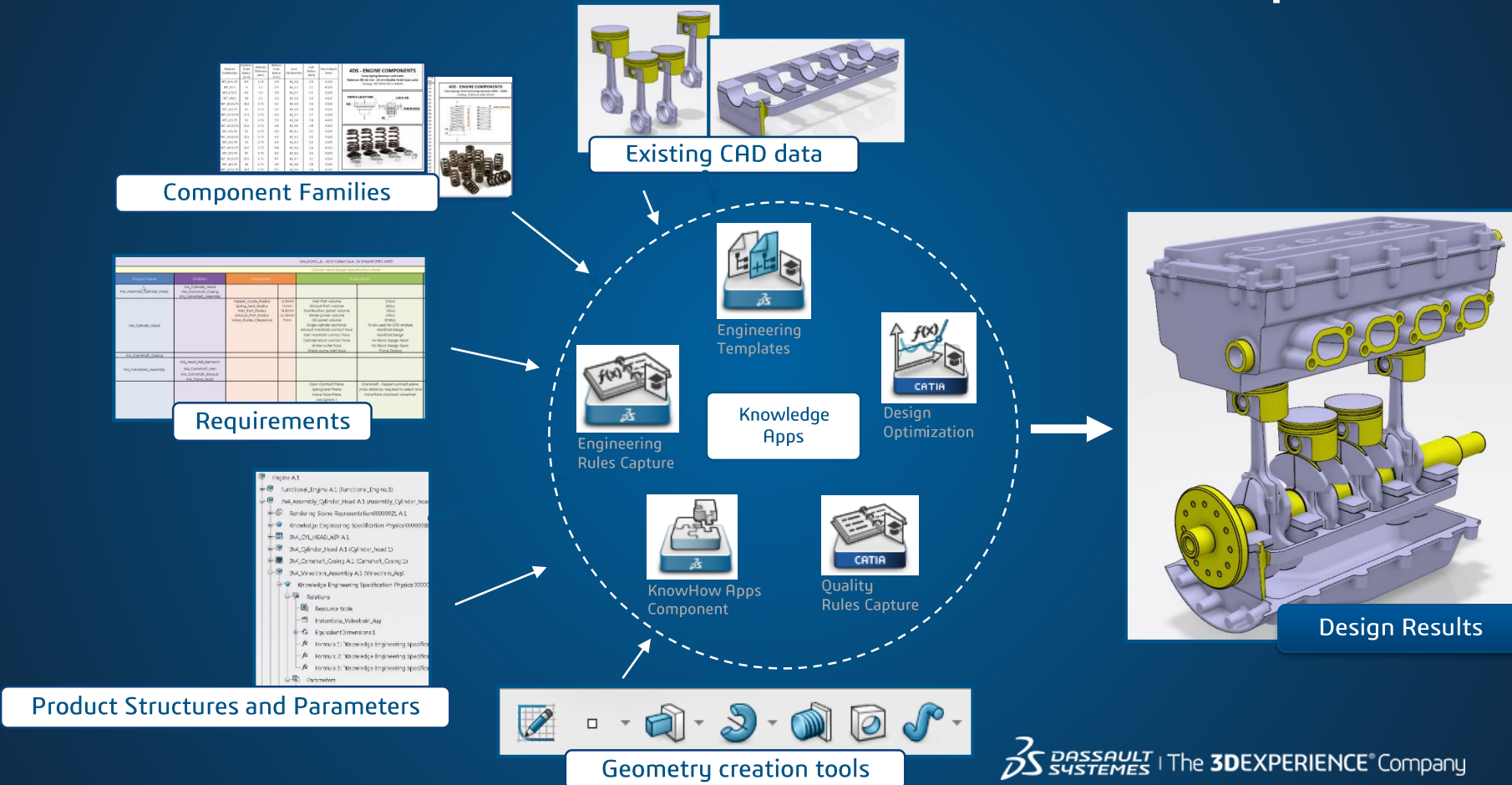


20% ~ 40%
overall
savings

Aim of CATIA Knowledge applications

- ▶ CATIA Knowledge apps include a set of tools intended to assist engineering decisions. They automate design and detect predefined design errors to achieve maximum process productivity.
- ▶ These apps enable users to:
 - ▷ **Automate** product definition and create generic models in order to increase efficiency
 - ▷ **Capture** corporate engineering knowledge and easily share know-how among all users
 - ▷ Ensure **compliance** with corporate standards
 - ▷ **Guide and assist** users through their design tasks
 - ▷ Allow early attention to final design specifications **preventing costly redesigns**
- ▶ Requirements, standard component catalogs, compliance rules, product structures, parameters, geometry creation tools, existing 3D data can all be accessed and utilized to create quick and robust designs.

How is KBE addressed on the 3DEXPERIENCE platform ?



Multi Discipline Automated Engineering – 2019x

Product Optimisation Designer (KPO)

A sophisticated set of applications to optimize the product definition process with respect to key performance indicators.

Template Designer (KDI)

Define and reuse smart engineering templates, and manage your company IP with knowledgeware rules and checks.



Engineering
Rules Captur



Engineering
Templates



Design
Optimization



Quality Rules
Capture

Quality Rules Designer (KHC)

Create and deploy enterprise rules, standards and best practices. Enrich by utilizing Quality Rules Administrator functionalities.

Design Apps Developer (KAC)

Define and manage custom apps developed on the 3DEXPERIENCE.



Know-how
Apps
Component



Know-how
Apps UE



Know-how
Apps
Creation



Know-how
Apps
Resources

Design Apps User (KAX)

Use custom apps developed with Design Apps Developer Role within the 3DEXPERIENCE



Custom Apps
authored using
KAC role

3D Generative Innovator (XGG)

Design, explore and validate variants of complex, repetitive and non-regular shapes and patterns with an intuitive algorithmic approach

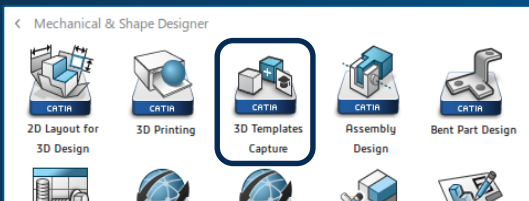


CATIA
xGenerative
Design

3D Templates Capture for Mechanical Designers

Basic knowledge functions like creating **user parameters**, **formulas**, **design tables** and **defining reports** are available with CATIA design roles.

3D Templates capture app is also part of most mechanical design roles. It provides powerful replication functionalities in the form of Powercopies and User Defined Features. Further existing Engineering Templates can be instantiated as well.



PowerCopy & UDF share three important concepts:

- **Components** are intended to be duplicated.
- **Inputs** are elements not added as components but pointed at through an external link by a component. They must be valued during instantiation.
- Components can have **parameters** that are listed in the Parameter tab during the definition. Publishing a parameter allows the user to set it during instantiation.



A power copy can be seen as a **smart Copy/Paste**: All **“power copied” features are displayed in the tree and can be modified**. There is no link with reference after instantiation and it uses the Copy/Paste and the Replace technologies.

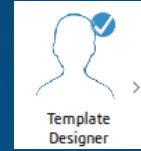


A user feature is a simple way to create custom feature: original group of features are seen in the instance as one unique feature. The point of using user feature is to **create new geometrical features from existing features and protect the intellectual property**. By default, every internal feature (component) is hidden and cannot be modified in the instance. The user of the user feature can only see its external view which is made of its inputs, parameters and outputs.

Multi-Discipline Automated Engineering

Role

Template Designer (KDI)



Define and reuse smart engineering templates, and manage your company IP with Knowledgeware rules and checks, automate design-tasks

Apps



Engineering
Rules Captur



Engineering
Templates

Functions



Rule...



Check...



Reactions



Knowledge
Pattern

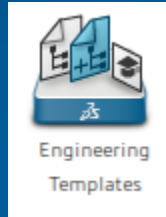


Actions



Engineering Template

Template Designer | Engineering Templates

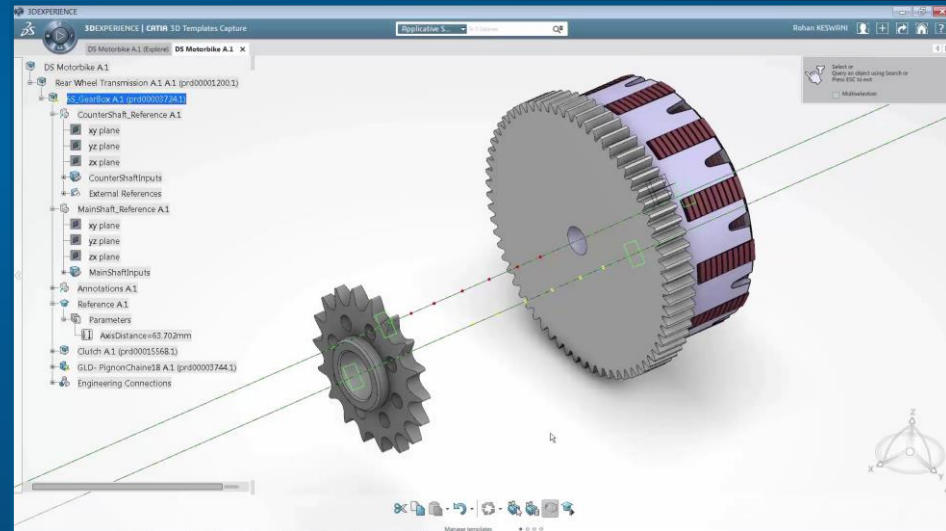


Template Designer | Engineering Templates

Interactively create intelligent knowledge templates: the encapsulation of feature, part and assembly specifications of any level of complexity within 3D Templates Capture allows the capture of the design methodology defined interactively, as well as the reuse across the extended enterprise. Templates can be extracted from existing features, part and assembly designs or created specifically. They can contain not only geometry, but also any associated parameters or relations, including embedded intelligence of design rules, design tables and checks, providing the ability to encapsulate the specifications of sophisticated adaptive features.

Edit and maintain the 3D templates: the management of captured knowledge is easier. You can to edit and maintain templates in the same interactive way as modifying any part design (no programming skills required). This allows the templates to be adapted to the changing requirements of the organization, to accommodate improved methodologies or customer requirements.

Store 3D templates in catalogs for re-use: once defined, 3D templates can be stored in catalogs to allow easy access and management for reuse across the extended enterprise.





New Tab 1 x

Add Tag

Who	When	What
Where	Why	How

Tag Name

Enter Tag

Add Tag

6WTags

Search Tags

Who 3

Last Modified By | 1

Organization | 1

Responsible | 1

When 2

Creation Date | 1

Modification Date | 1

What 8

Demo 1 x

Collaborative Review | 2

2 elements selected

3DSearch

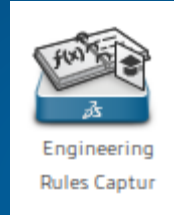
Last Index: 4/5/2016 10:41:13 AM - @searchfor:my:non:indexed:content

✓ 2 Results | 1163



	A...	Title	Description	Name	Modification Date	Creation Date	Type	Responsible
1		PS_EngineeringTemplate3DPart_Belt001		ToDelete.prd-in...	2/27/2017 1:18...	2/27/2017 1:18...	Physical Pr...	Peter Stuehn
2	<input checked="" type="checkbox"/>	PS_EngineeringTemplateAssembly_001		ToDelete.prd-in...	2/27/2017 1:18...	2/27/2017 1:18...	Physical Pr...	Peter Stuehn
3		PS_3D Shape Skeleton_EngineeringTemplateAssembly_001		ToDelete.3sh-l...	2/27/2017 1:18...	2/27/2017 1:18...	3D Shape S...	Peter Stuehn
4	<input checked="" type="checkbox"/>	PS_Template_BeltDrive001		tpl-Interfix to be...	2/27/2017 11:4...	2/27/2017 11:1...	Multi-Disc...	Peter Stuehn
5		PS_3D Shape Skeleton00002618		3sh-Interfix to b...	2/27/2017 11:3...	2/27/2017 10:0...	3D Shape S...	Peter Stuehn
6		PS_3D Shape Skeleton_EngineeringTemplateAssembly_001		3sh-Interfix to b...	2/27/2017 11:3...	2/27/2017 8:42...	3D Shape S...	Peter Stuehn
7		PS_Chapter002_Functions		PS_Chapter00...	2/27/2017 11:2...	10/17/2016 1:2...	Catalog Ch...	Peter Stuehn
8		PS_EngineeringTemplate_Test_Bearing001		prd-Interfix to b...	2/27/2017 10:2...	2/27/2017 10:2...	Physical Pr...	Peter Stuehn
9		PS_EngineeringTemplate_Test_Plate001		prd-Interfix to b...	2/27/2017 10:0...	2/27/2017 10:0...	Physical Pr...	Peter Stuehn
10		PS_EngineeringTemplate_TestAssembly001		prd-Interfix to b...	2/27/2017 10:0...	2/27/2017 10:0...	Physical Pr...	Peter Stuehn
11		PS_EngineeringTemplate3DPart_Belt001		prd-Interfix to b...	2/27/2017 9:10...	2/27/2017 9:10...	Physical Pr...	Peter Stuehn
12		PS_EngineeringTemplate3DPart_Shaft001		prd-Interfix to b...	2/27/2017 8:55...	2/27/2017 8:55...	Physical Pr...	Peter Stuehn
13		PS_EngineeringTemplate3DPart_Shaft002		prd-Interfix to b...	2/27/2017 8:55...	2/27/2017 8:55...	Physical Pr...	Peter Stuehn
14		PS_EngineeringTemplateAssembly_001		prd-Interfix to b...	2/27/2017 8:42...	2/27/2017 8:42...	Physical Pr...	Peter Stuehn
15		PS_Chapter121_Details		PS_Chapter12...	2/24/2017 3:10...	2/23/2017 2:10...	Catalog Ch...	Peter Stuehn
16		PS_Details_Drawing00000490		drw-Interfix to b...	2/24/2017 3:08...	2/23/2017 1:49...	Drawing	Peter Stuehn
17		PS_Chapter21		PS_Chapter21	2/24/2017 9:42...	9/16/2016 9:27...	Catalog Ch...	Peter Stuehn
18		PS_Template000102_SMD		tpl-Interfix to be...	2/24/2017 9:40...	2/24/2017 9:40...	Multi-Disc...	Peter Stuehn
19		Drawing_PS_StartModel_SMD_01		drw-Interfix to b...	2/24/2017 9:33...	2/24/2017 9:04...	Drawing	Peter Stuehn
20		PS_StartModel_SMD_01		prd-Interfix to b...	2/24/2017 9:04...	2/24/2017 9:04...	Physical Pr...	Peter Stuehn
21		WeightDetection_PS_Drawing00000252		drw-Interfix to b...	2/24/2017 7:52...	2/24/2017 7:52...	Drawing	Peter Stuehn
22		WeightDetection_PS_Assembly_SMD_01		prd-Interfix to b...	2/24/2017 7:52...	2/24/2017 7:52...	Physical Pr...	Peter Stuehn

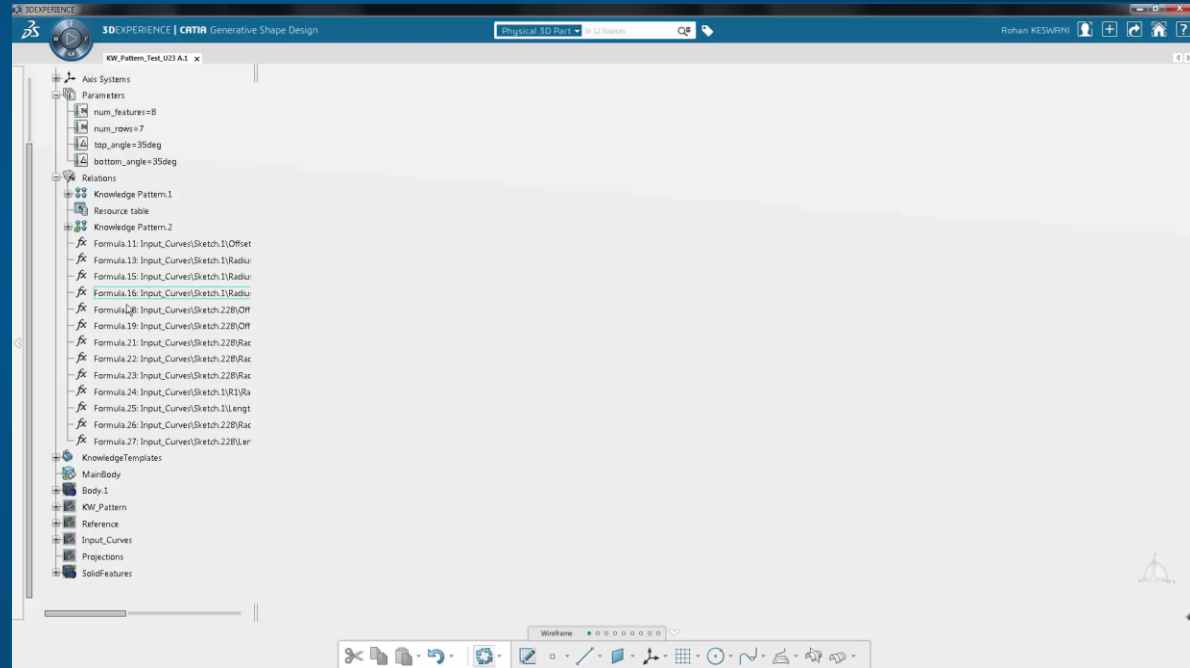
Template Designer | Engineering Rules Capture





A **Knowledge Pattern** is a feature that lets you instantiate templates into 3D Part documents in batch mode. It also enables you to manage the instantiated objects in an associative way and allows to generate automatically repetitive structures with differences between each instance of the structure.

A knowledge pattern uses the standard EKL language with some additional specific methods and a specific list that provides a very powerful associativity level. For example, you can iterate on an input list containing points, instantiate a user feature, select its point input, finalize the user feature instantiation, then begin a new loop only on the created user features and use them as inputs of other user features. In this example, a knowledge pattern is used to design the front grill of a car. Each feature created here is unique and follows the car's characteristic design lines.



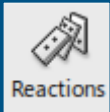
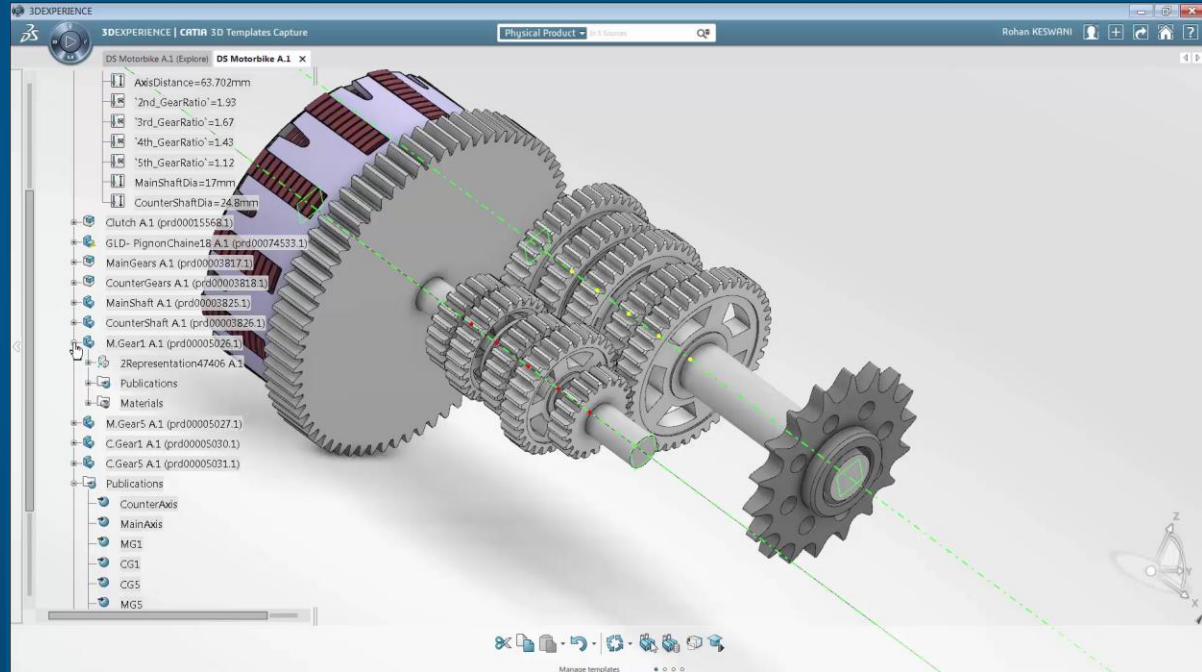


A **Rule** is a set of instructions, generally based on conditional statements, whereby the relationship between parameters is controlled.

In addition, depending on the context described by the rule instructions, actions can be executed:

- To set a value or a formula to parameters, including feature activity.
- To display information windows.
- To affect points, curves and surfaces and thus allow contextual and automatic topological changes.

In this example, a rule decides the number of teeth on mating gears based on the desired gear ratio and distance between shafts.



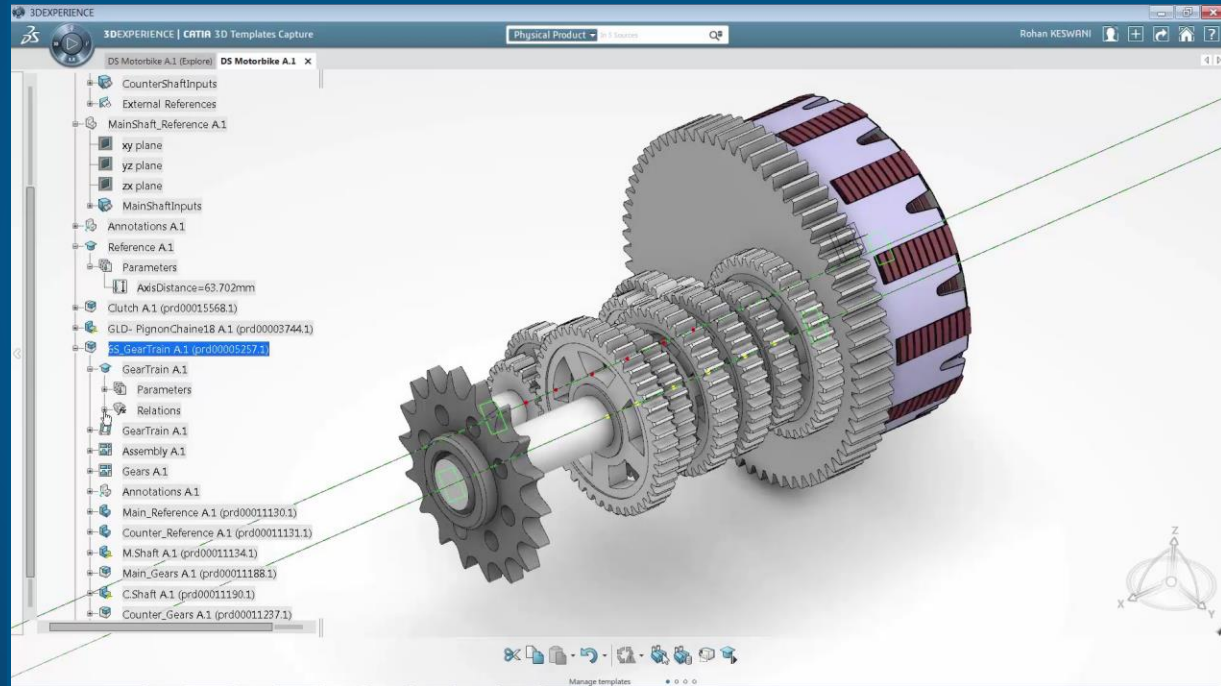
Reactions are similar to rules but are triggered only by certain events like modification of specific attributes, before an update is executed, etc.

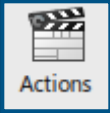


You can create **Checks** to make sure that some conditions are fulfilled.

A check is a multiline statement that you can write either by typing directly the appropriate syntax in the editor box or by selecting items from the editor dictionary list. It is a Boolean expression.

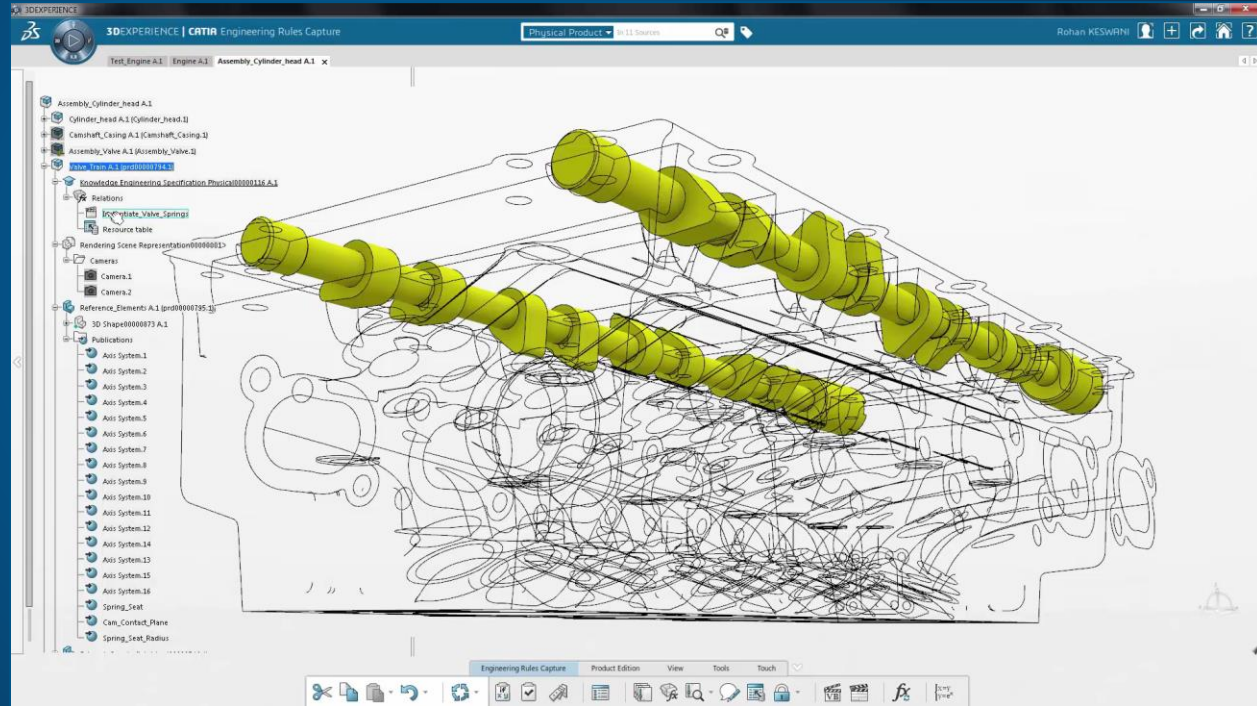
In this example, checks are created to ensure appropriate clearance with tolerance is always maintained between mating gears





You can create an **Action** that lets you perform interactively what the Enterprise Knowledge Language enables you to do.

In this example, valve springs are selected, instantiated and positioned in the assembly. Driving parameters and bounding elements are selected from the existing geometry. These serve as inputs for an Action to calculate spring dimensions, select them from a catalog and insert them in the cylinder head.



Design Apps Developer / User



Difference between Developer and User roles (KAC vs. KAX)

Know-how Apps Authoring Suite is a series of apps that allow defining and managing custom applications developed within **3DEXPERIENCE**. The definition of such apps is stored in database and can be easily deployed on final user seats.

The Developer role allows for the definition and management of apps. Whereas, the User role only allows users to utilize the operations and processes contained within the app. All rules, templates, actions and other objects utilized by the app are hidden and cannot be modified by someone with only a user license.

Owing to the **3DEXPERIENCE** Platform, the Developer can quickly access and manage apps according to any changes in the company's design processes or best practices and the same will be instantly available to the user without any cumbersome installation or deployment procedures.

Overall, compared to costly 3rd party implementations, Design Apps roles provide a very quick and robust way to deliver and manage customized automations for the extended enterprise.

Design Apps Developer

The Design Apps Developer role provides Know-How Apps Suite which is a series of apps that allow interactively defining and managing custom applications developed within 3DEXPERIENCE.

The definition of such apps is stored in the database and can be easily deployed for end users

Custom apps allows companies to capture and deploy best practices across all stakeholders.



Custom Apps

- Define
- Deploy
- Manage



Capture Best Practices

- Corporate IP
- Company specific
- Evolve continuously



Interactive Definition

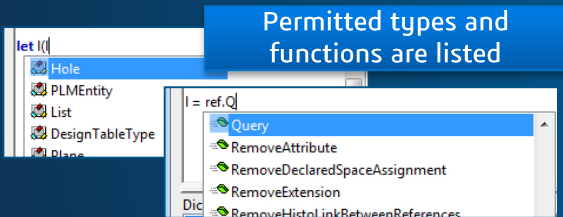
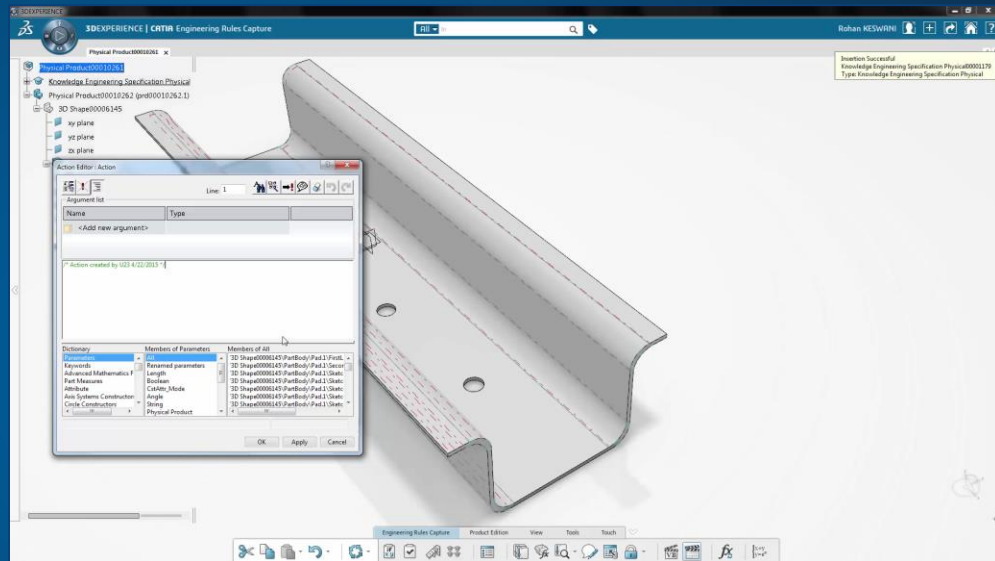
- Commands
- Workflows
- No IT expertise

Understanding the EKL

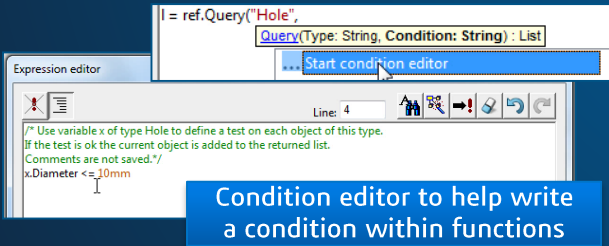
Enterprise Knowledge Language (EKL) is a programming language intended for the development of applications and their integration in a more general context.

EKL provides a way to:

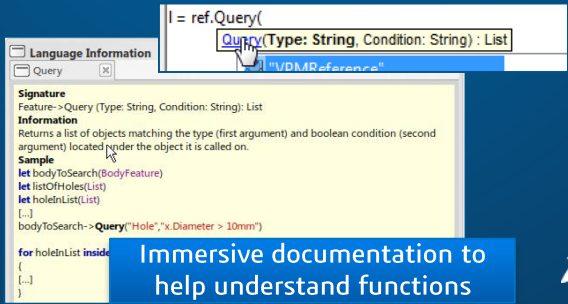
- Embed decisions into the definition of a template to process information (from inputs to outputs)
- Embed logic into the definition of an engineering process to operate pre defined actions



Permitted types and functions are listed



Condition editor to help write a condition within functions



Immersive documentation to help understand functions

The companion course - **Introduction to Enterprise Knowledge Language** does a great job to not only explain the EKL but basic knowledge concepts as well. <https://companion.3ds.com>

Knowledge Based Engineering - Benefits



Power - 57 kW
Speed Ratio 1 - 4:1
Speed Ratio 2 - 2.9:1
Material - Steel

quick function driven designs



more time for innovation



first time right



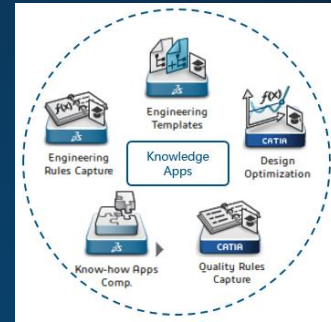
avoid expensive redesigns



reduce time on standard tasks



increase overall productivity





3DEXPERIENCE Conference for Design, Modeling & Simulation 2020

WHEN November 10 - 12, 2020

WHERE Darmstadtium, Darmstadt,
Germany

WEB 3ds.com/events/



WE ARE LOOKING FORWARD TO SEE YOU AGAIN!