

INSTITUTE FOR SOFTWARE SYSTEMS ENGINEERING



Alexander Egyed



**JOHANNES KEPLER
UNIVERSITÄT LINZ**

YOU ARE OUR USER!



[This Photo](#) by Unknown Author is licensed under [CC BY](#)

Mechanics

Computations

Electrical

Requirements

Trade-Off Exploration

Design

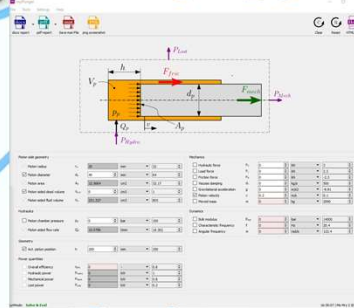
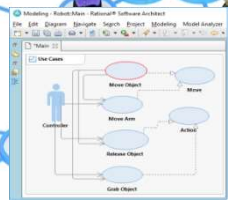
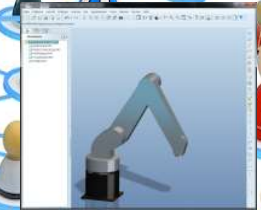
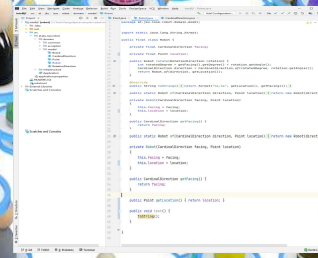
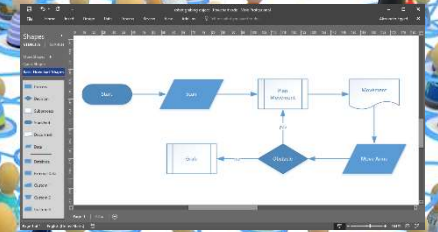
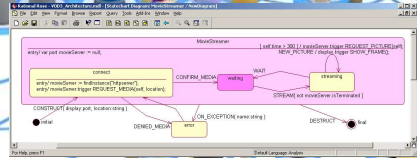
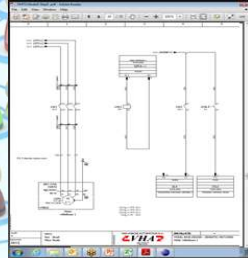
Simulation

Planning

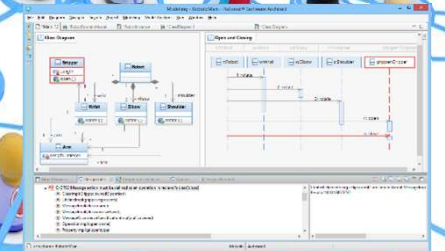
Programming

...

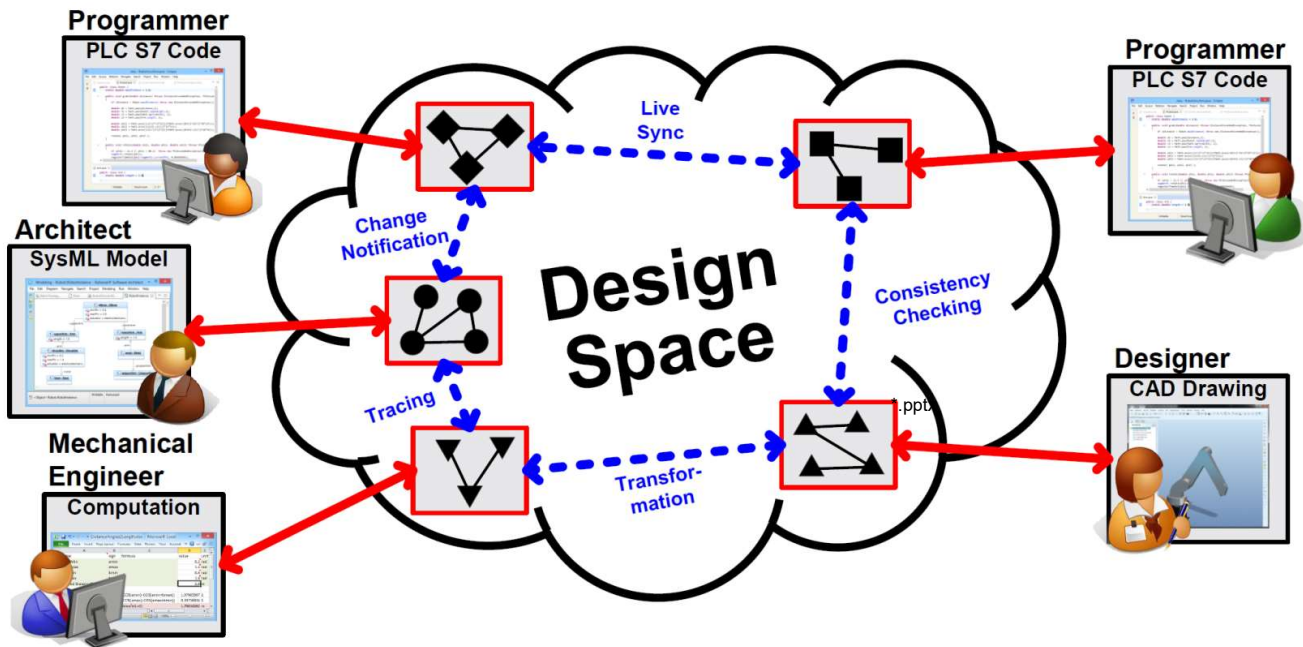
Amazing Amount of explicit and implicit
Engineering Knowledge



name	sign	formula	value
alphaMin	amax		0.2 rad
alphaMax	amax		1.0 rad
betaMin	bmin		0.0 rad
betaMax	bmax		2.8 rad
desiredGraspDistance	does		2m
maxNormalizedRadius	r1	(COS(alpha)-COS(alpha+betaMax))	1.9705907
minNormalizedRadius	r1	(COS(alpha)-COS(alpha+betaMin))	0.98748204
minArmLength	x	does(r1-r2)	1.17154403m



COLLABORATIVE ENGINEERING



interdisziplinär

- Informatik
- Mechatronik
- Prozessmanagement
- Sozial Networks
- Projektmanagement

	A	B	C	D	E
1	name	sign	formula	value	unit
2	alphaMin	amin	(COS(amin)-COS(amin+bmax))	1.97005907	1
3	alphaMax	amax	(COS(amax)-COS(amax+bmin))	1.4	rad
4	betaMin	bmin		0.4	rad
5	betaMax	bmax		2.8	rad
6	desired GraspingDistance	ddes		1.78016282	m
7					
8	maxNormedRadius	r1	(COS(amin)-COS(amin+bmax))	1.97005907	1
9	minNormedRadius	r2	(COS(amax)-COS(amax+bmin))	1.4	rad
10	minArmLength	x	ddes/(r1-r2)	1.78016282	m

```

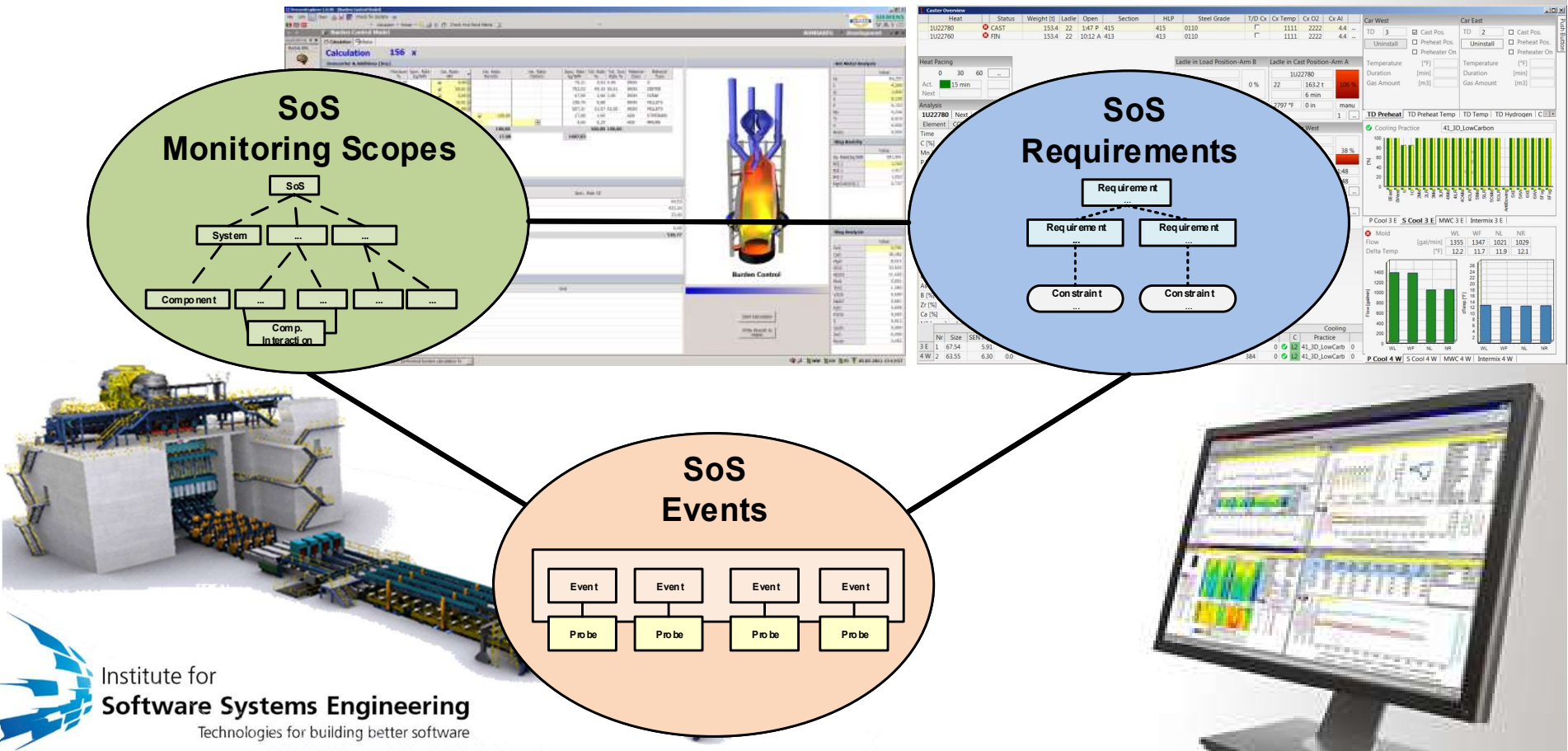
public class Robot {
    static double maxDistance = 2.8;

    public void grab(double distance) throws DistanceExceededException, PhiExceededException {
        if (distance > Robot.maxDistance) throw new DistanceExceededException();
        double d2 = Math.pow(distance, 2);
        double h2 = Math.pow(Robot.softHeight, 2);
        double c2 = Math.pow(Math.sqrt(d2-h2), 2);
        double l2 = Math.pow(Arm.Length, 2);

        double phi1 = Math.acos((c2/(2*(c2+l2)))+Math.acos((d2+c2-h2)/(2*d2*c2)));
        double phi2 = Math.acos((l2+l2-c2)/(2*l2*l2));
        double phi3 = Math.acos((c2)/(2*(c2+l2)))+Math.acos((d2+h2-c2)/(2*d2*h2));
        rotate(phi1, phi2, phi3);

        public void rotate(double phi1, double phi2, double phi3) throws PhiExceededException {
            if (phi1 < 0 || phi1 > 80.2) throw new PhiExceededException();
            registerTimeOut(phi1-segment.currentPhi1, 0.86666666);
        }
    }
}
  
```

CD LABOR MEVSS



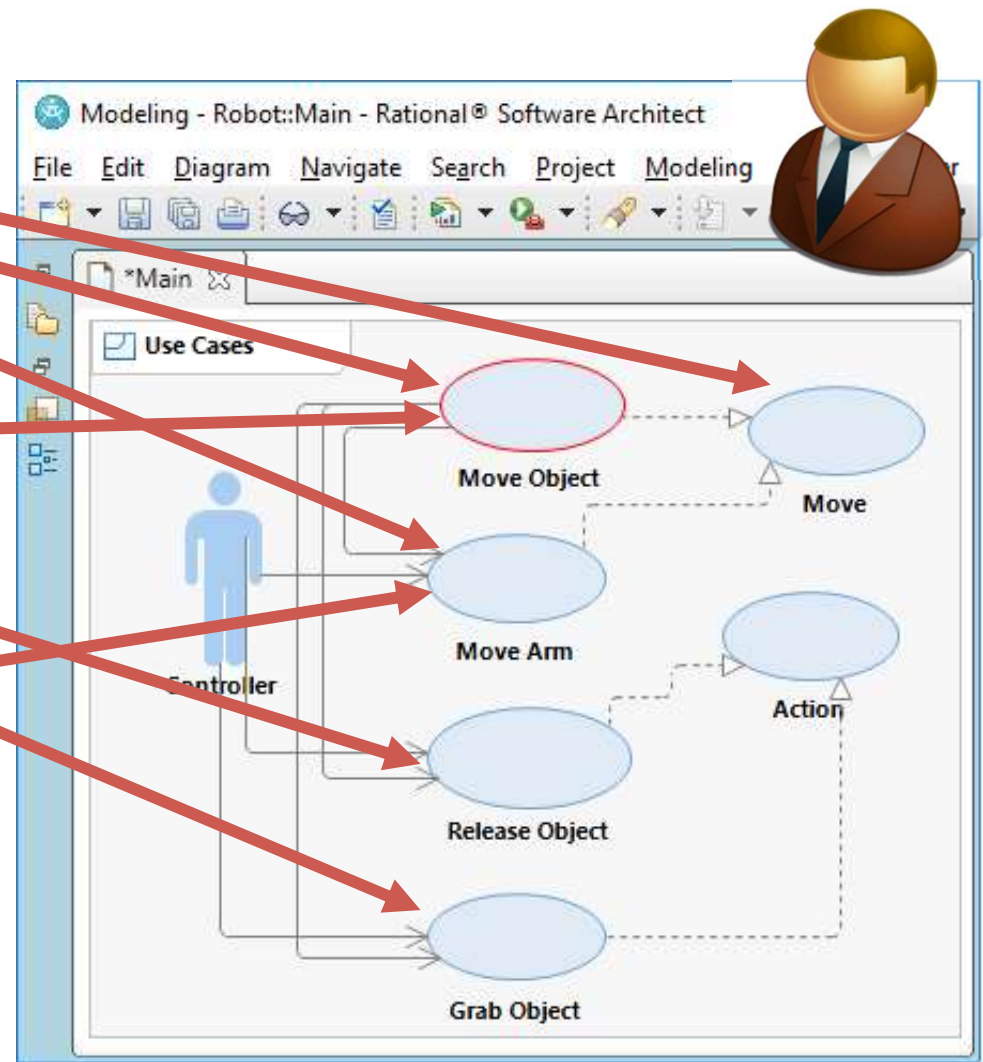
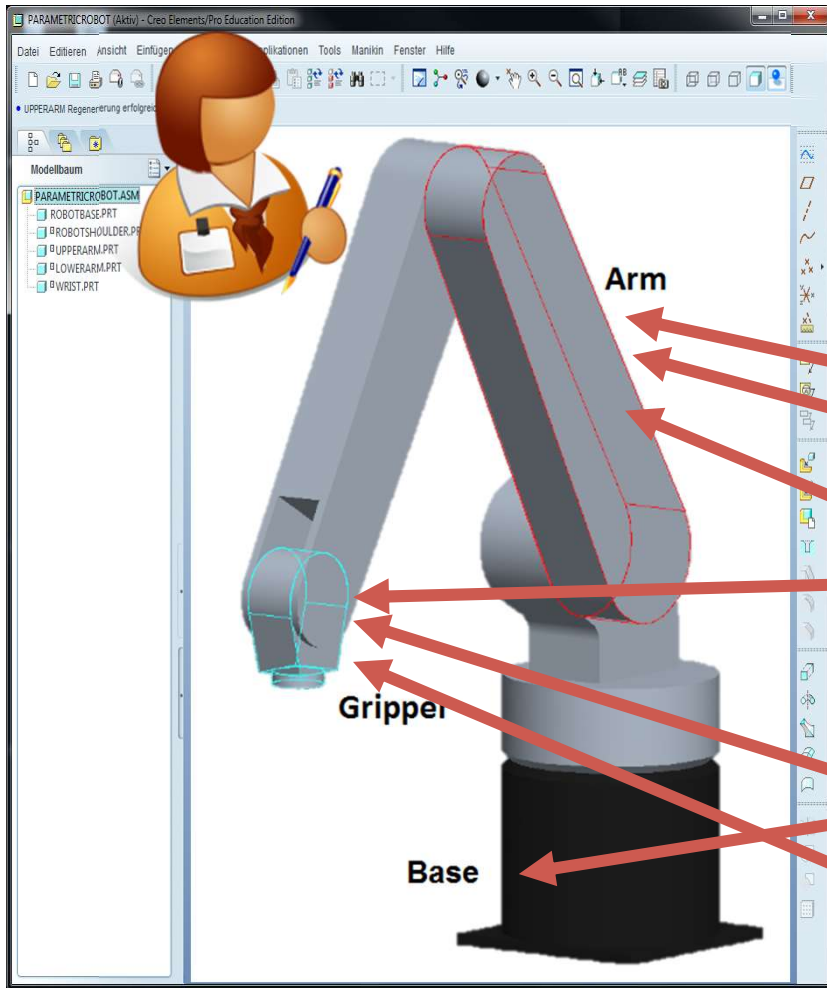
VHA CASE STUDY



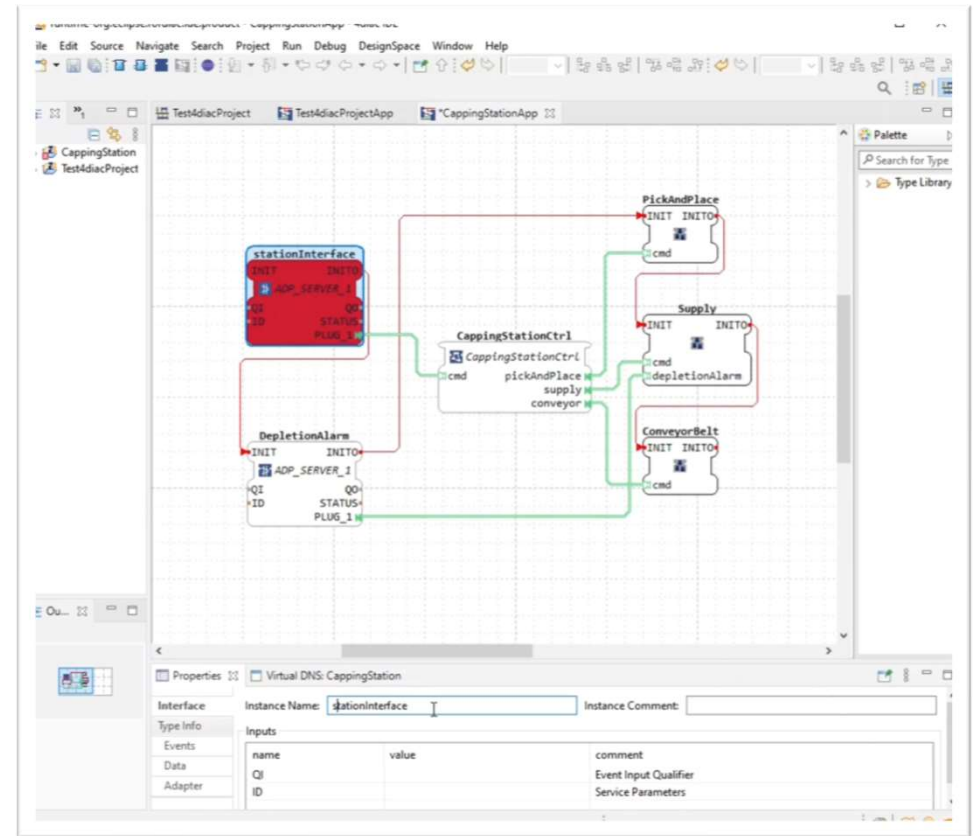
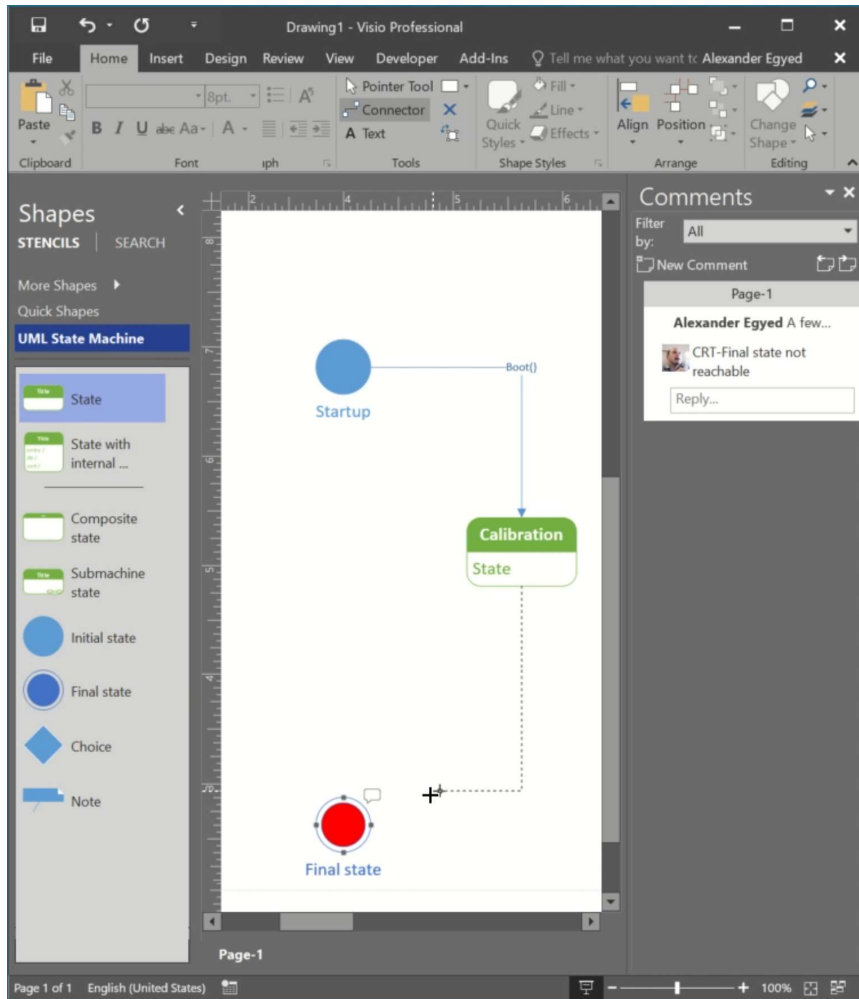
Conveyor Belt

Robotic Arm

MORE LINK EXAMPLES



4DIAC (IEC 61499 FOR I4.0)



Trace Matrices

Trace Matrix Type: 1

Trace Matrix Instance: TraceMatrix-implementedBy/implements

Create Trace Matrix Type

Requirement

JavaMethod	Requirement	U	U
	Pick up an object		U
Arm.Arm		U	U
Arm.getFragileDuration	U -> PT	U	U
Arm.getMaximumObjectHeig		U	U
Arm.getMinimumObjectHeig		U	U
Arm.getRegularDuration	U -> PT	U	U
Arm.liftArm		U	U
Arm.lowerArmToObject		U	U
Arm.setFragileDuration		U	U
Arm.setMaximumObjectHeig		U	U
Arm.setMinimumObjectHeig		U	U
Arm.setRegularDuration		U	U
ArmJoint.ArmJoint		U	U
ArmJoint.liftArm		U	U
ArmJoint.lowerArm		U	U
Base.main		T	U
Base.run		T	U
Gripper.Gripper		U	U
Gripper.getCloseDuration		U	U
Gripper.getFingerLength		U	U
Gripper.getMaximumDiamete		U	U
Gripper.getMinimumDiamete		U	U
Gripper.getOpenDuration		U	U
Gripper.grabObject		T	U
Gripper.releaseObject	N -> PT	U	U
Gripper.setCloseDuration		U	U
Gripper.setMaximumDiamete		U	U
Gripper.setMinimumDiamete		U	U
Gripper.setOpenDuration		U	U
GripperJoint.GripperJoint		U	U
GripperJoint.close		T	U
GripperJoint.getMaximumAng		U	U
GripperJoint.getMinimumAng		U	U
GripperJoint.open		N	U
GripperJoint.setMaximumAng		U	U
GripperJoint.setMinimumAng		U	U

Recommended Trace

Trace Error due to Code Error

RobotArm src > components > Gripper > releaseObject

```

public class Gripper {
    private double fingerLength;
    private int closeDuration;
    private int openDuration;
    private GripperJoint gripperJoint;
    private double minimumDiameter;
    private double maximumDiameter;

    public Gripper(double fingerLength, GripperJoint gripperJoint) {
        this.fingerLength = fingerLength;
        this.gripperJoint = gripperJoint;
    }

    public void grabObject(double objectDiameter) {
        if (objectDiameter < minimumDiameter || objectDiameter > maximumDiameter)
            throw new IllegalArgumentException("The object size exceeds requirements");
        double opening = objectDiameter / 2;
        double distance = Math.sqrt(Math.pow(fingerLength, 2) + Math.pow(opening, 2));
        double angle = Math.acos(distance / fingerLength);
        gripperJoint.close(angle, closeDuration);
    }

    public void releaseObject() {
        gripperJoint.close(desiredAngle: 0, openDuration);
    }

    double getMinimumDiameter() { return minimumDiameter; }
    void setMinimumDiameter(double minimumDiameter) {
        this.minimumDiameter = minimumDiameter;
    }
    double getMaximumRadius() { return maximumRadius; }
    void setMaximumRadius(double maximumRadius) {
        this.maximumRadius = maximumRadius;
    }
    double getMinAngle() { return minAngle; }
    void setMinAngle(double minAngle) {
        this.minAngle = minAngle;
    }
}

```

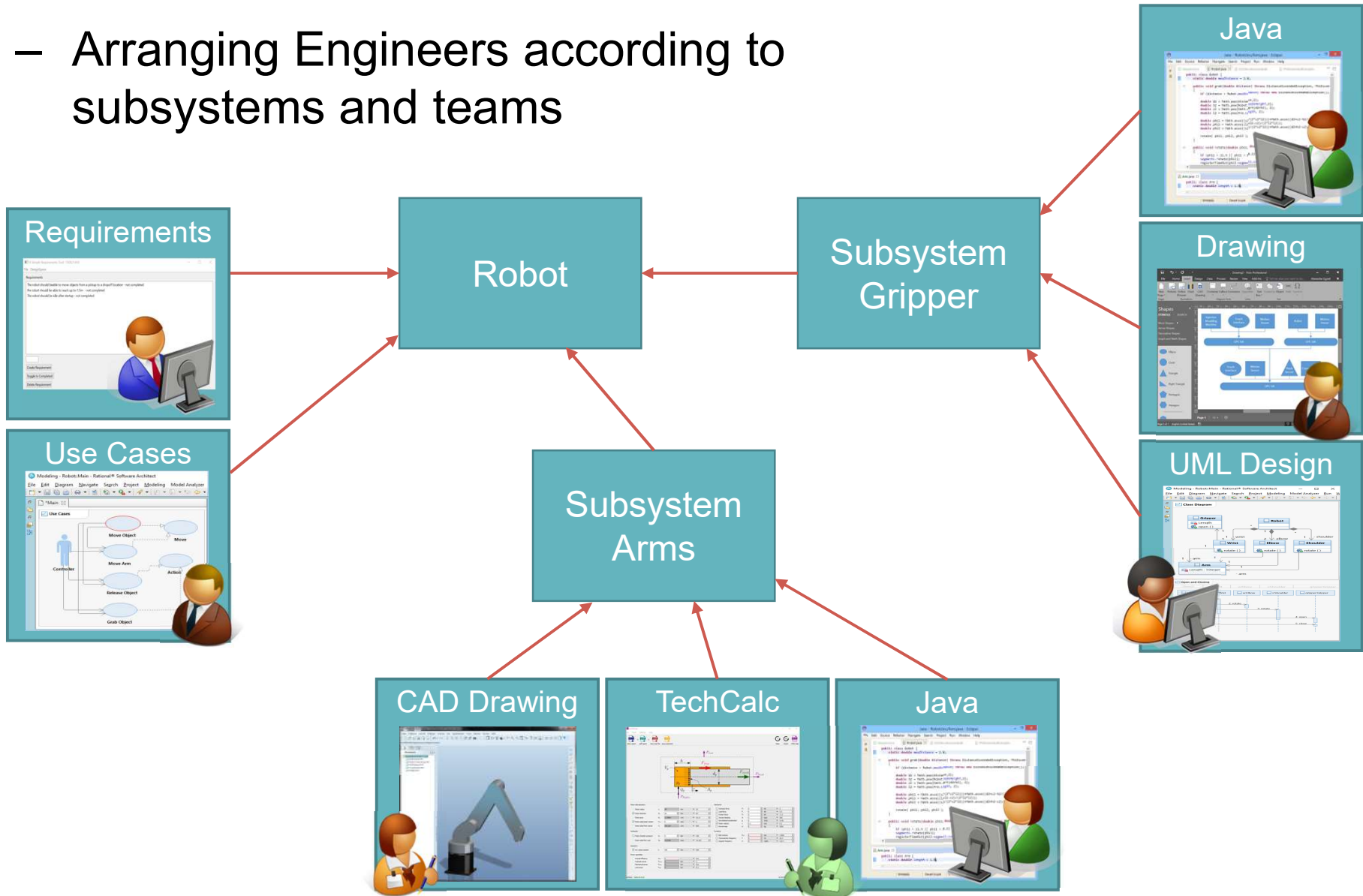
Push Local version has been pushed to the DesignSpace.

Push: Local version has been pushed to the DesignSpace. (a minute ago)

26:31 (9 chars) CRLF UTF-8 4 spaces

DESIGNSPACE DEMO 8

- Arranging Engineers according to subsystems and teams



BUILDING THE FIRST ROBOT

Robot 1

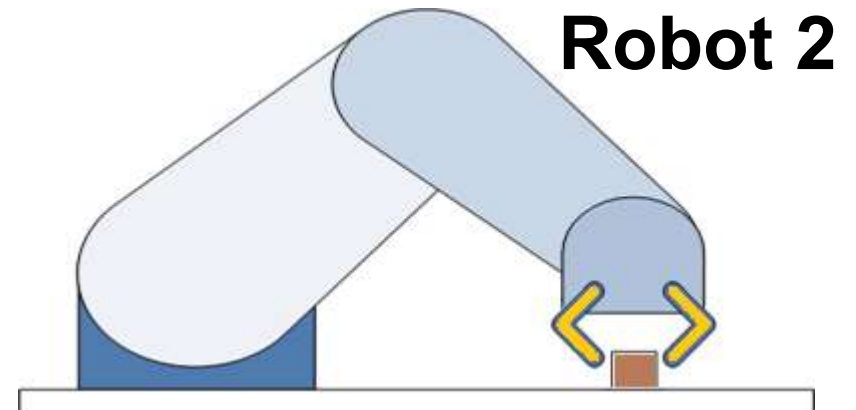
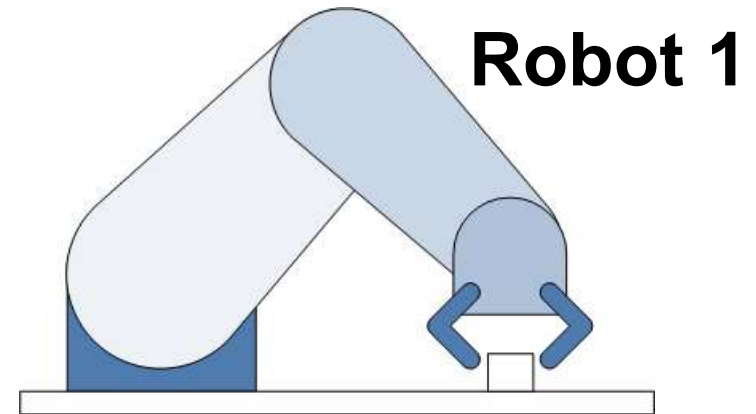
- CAD
- Computations (Matlab)
- EPlan
- Modelica
- SysML / UML
- Test/Use Case Scenarios
- Configuration Databases
- Commissioning Parameters
- Source Code
- ...

The collage illustrates the multi-disciplinary software development process for Robot 1. It includes:

- Microsoft Excel:** A spreadsheet for parameter management with columns for name, sign, formula, value, and unit. Key parameters include alphaMin (0.2 rad), alphaMax (1.4 rad), betaMin (0.4 rad), betaMax (2.8 rad), and minArmLength (1.27154487 m).
- Rational Software Architect:** A SysML Use Case diagram showing interactions between a Controller and actions like Move Object, Move Arm, Release Object, and Grab Object.
- Eclipse IDE:** Java source code for a Robot class, featuring methods for grab and rotate, with complex trigonometric calculations for joint angles and distances.
- 3D CAD Model:** A 3D rendering of a two-link robotic arm.
- Mathematical Diagrams:** Geometric diagrams showing force vectors (F₁, F₂, F₃) and distances (d₁, d₂) on a robot arm structure.

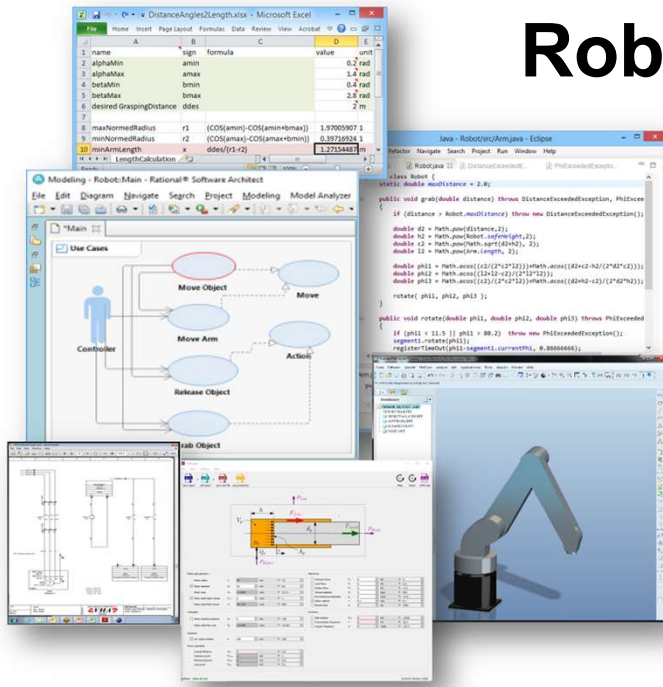
BUILDING THE SECOND ROBOT

- The second robot is “kind of like” the first robot
- A max distance the robot needs to be able to reach is 2.8 meters
- Also it needs a different gripper

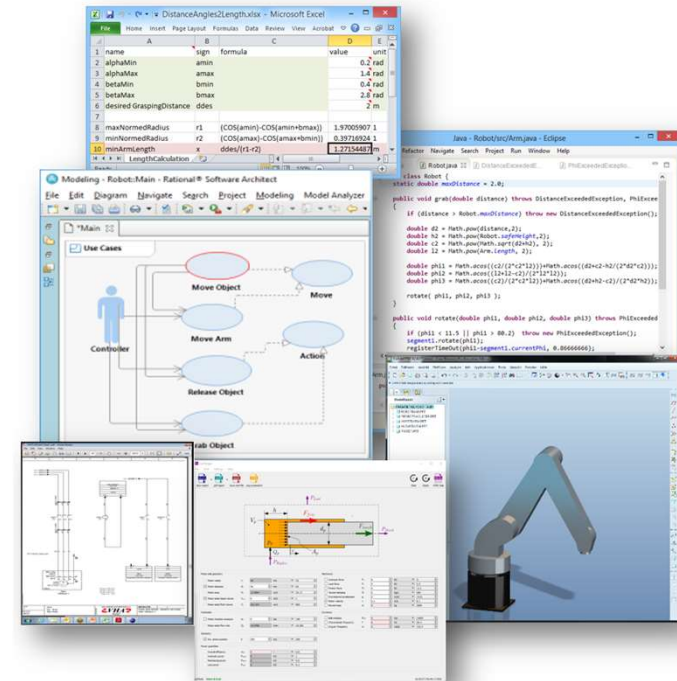


CLONE AND OWN THE ENGINEERING ARTIFACTS OF THE FIRST ROBOT

Robot 1



Clone and Own



Robot 2

CLONE AND OWN THE ENGINEERING ARTIFACTS OF THE FIRST ROBOT

Robot 1



	A	B	C	D	E
1	name	sign	formula	value	unit
2	alphaMin	amin		0.2	rad
3	alphaMax	amax		1.4	rad
4	betaMin	bmin		0.4	rad
5	betaMax	bmax		2.8	rad
6	desired GraspingDistance	ddes		2.8	m
7					
8	maxNormedRadius	r1	$(\text{COS}(\text{amin})-\text{COS}(\text{amin}+\text{bmax}))$	1.97005907	1
9	minNormedRadius	r2	$(\text{COS}(\text{amax})-\text{COS}(\text{amax}+\text{bmin}))$	0.39716924	1
10	minArmLength	x	$\text{ddes}/(\text{r1}-\text{r2})$	1.78016282	m

Clone and Own

The collage includes:

- A screenshot of the Excel spreadsheet from Robot 1.
- A screenshot of the Eclipse IDE showing Java code for a robot class with methods like `grab` and `rotate`.
- A screenshot of the Rational Software Architect tool showing a Use Case diagram with use cases like 'Move Object', 'Move Arm', and 'Release Object'.
- A 3D model of a robotic arm.

Robot 2

WORLD LEADING

Campus

ÖÖNachrichten

„Wir mischen bei den ganz Großen mit“

Kepler-Universität belegt in weltweitem „Software Engineering“-Ranking Platz 35

LINZ. „Wissenschaftler machen zwei Dinge wohl am liebsten: forschen und publizieren“, sagt der frühere JKU-Vizekanzler für Lehre Alexander Egyed. Eben jene Publikationen von Hochschulen und Unternehmen im Bereich Software Engineering nahm das JSS, das „Journal of Systems and Software“, weltweit unter die Lupe. Mit einem sehr erfreulichen Ergebnis für die Linzer Johannes-Kepler-Universität. Sie belegt im internationalen Vergleich den 35. Platz.

„Universitäten werden ja ständig miteinander verglichen und gerankt. Da ist man schon oft sehr froh, wenn man unter den Top 200 oder gar den Top 100 weltweit auftaucht“, sagt Egyed, Leiter des Instituts für Software Systems Engineering. „Aber Platz 35 zu erreichen, das ist schon eine sehr, sehr große Sache.“

Von A* bis D

Doch um die Hochschulen und Unternehmen bewerten zu können, unterteilte JSS erst die Publikationen, abhängig davon, in welchen Foren sie veröffentlicht wurden und wie groß die erreichte Aufmerk-



Ein Platz unter den weltweit Besten

Symbolbild: colourbox

mit den besten Institutionen weltweit mithalten, das macht einen schon sehr stolz“, sagt Egyed. „Das ist aber sicher nicht die Errungenschaft einer Einzelperson, sondern vieler Leute.“

Neben Egyeds Institut war auch das Department of Business In-

Systems Engineering werden Methoden untersucht, um Software-Entwicklung kosteneffizienter, transparenter und zuverlässiger zu machen. So widmet man sich etwa auch den Fachbereichen Architektur, Design und Wartung. Nachdem Software schon fast in



„Bei Rankings ist man froh, wenn man in den Top 100 auftaucht. Aber Platz 35 zu erreichen, ist eine sehr, sehr große Sache.“

■ Alexander Egyed, Leiter „Software Systems Engineering“



„Nur einen Platz vor der JKU liegt Microsoft. Unser nächstes Ziel ist eine Platzierung unter den Top 20.“

WORLD LEADING



Microsoft Academic Search Ranking: in Software Systems

JKU Ranks 16 worldwide



Microsoft Academic Search

Academic > Computer Science > Software Engineering

Top organizations in software engineering

1-101 of 1,907 results

Last 5 Years | All Continents

Organizations

- University College London
- Carnegie Mellon University
- King's College London
- University of Oxford
- Microsoft
- University of Melbourne
- Swinburne University of Technology
- Simula Research Laboratory
- Commonwealth Scientific and Industrial Research Organization
- University of New South Wales
- Ghent University
- Eindhoven University of Technology
- University of Twente
- Australian National University
- University of Nebraska Lincoln
- Johannes Kepler Universität Linz
- Swiss Federal Institute of Technology Zurich
- Politecnico di Milano
- University of Toronto
- Delft University of Technology
- University of Technology Sydney
- Pontificia Universidade Católica do Rio Grande do Sul
- École Polytechnique de Montreal
- Université du Luxembourg
- Fourth Military Medical University
- North Carolina State University
- Massachusetts Institute of Technology

CMU

Oxford

JKU

ETH Zürich

MIT