

Designing Future Aircraft with Eclipse RCP

EclipseCon Europe 2013

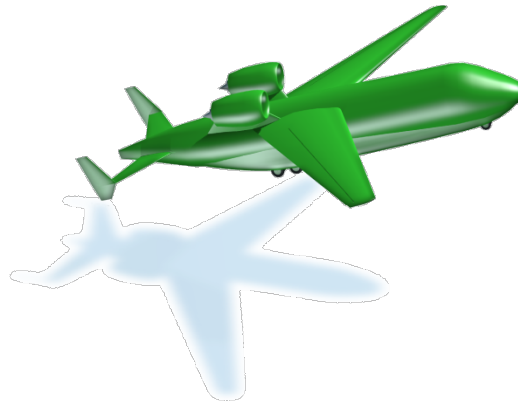
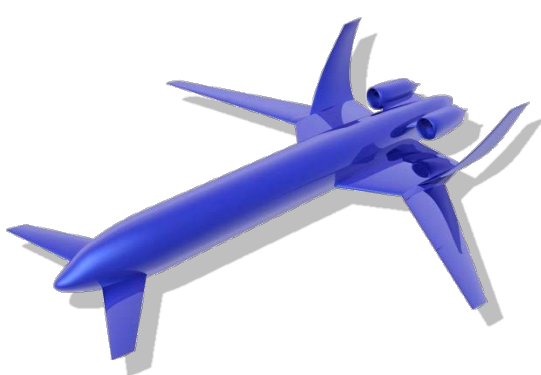
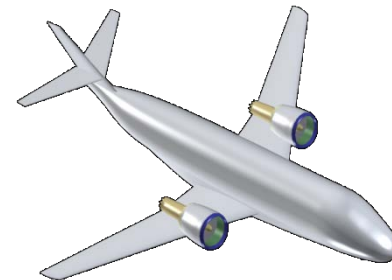
Doreen Seider

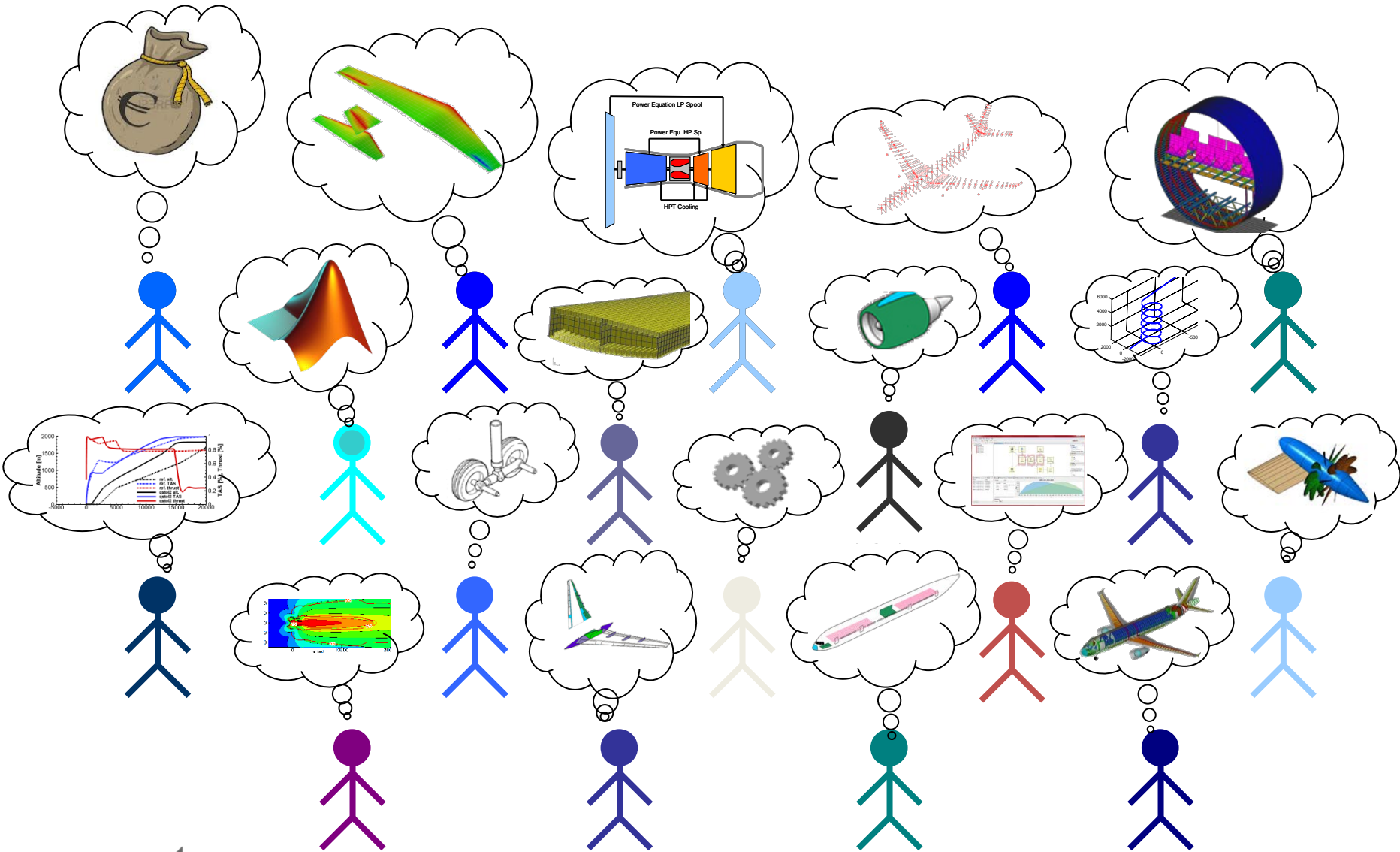
Knowledge for Tomorrow



Future Aircraft Design

- Get new aircraft configurations which
 - are more environment-friendly
 - require less operating costs





RCE: Software for Future Aircraft Design

- DLR (German Aerospace Center) develops software for future aircraft design called RCE (Remote Component Environment)
- RCE enables multidisciplinary collaboration to help experts from different disciplines to solve overall aircraft design task in common
- We built RCE on Eclipse RCP and made it open source (EPL)

The logo for RCE (Remote Component Environment) features the letters 'R', 'C', and 'E' in a bold, white, sans-serif font. Each letter is contained within a dark red, stylized rectangular frame with pointed corners and a notch at the top and bottom, resembling a gear or a component connector.

Outline

- Short introduction of RCE
- Selected aspects of RCE regarding Eclipse RCP
 - Modularity
 - Usability
 - Distribution management
- Example projects at DLR using RCE

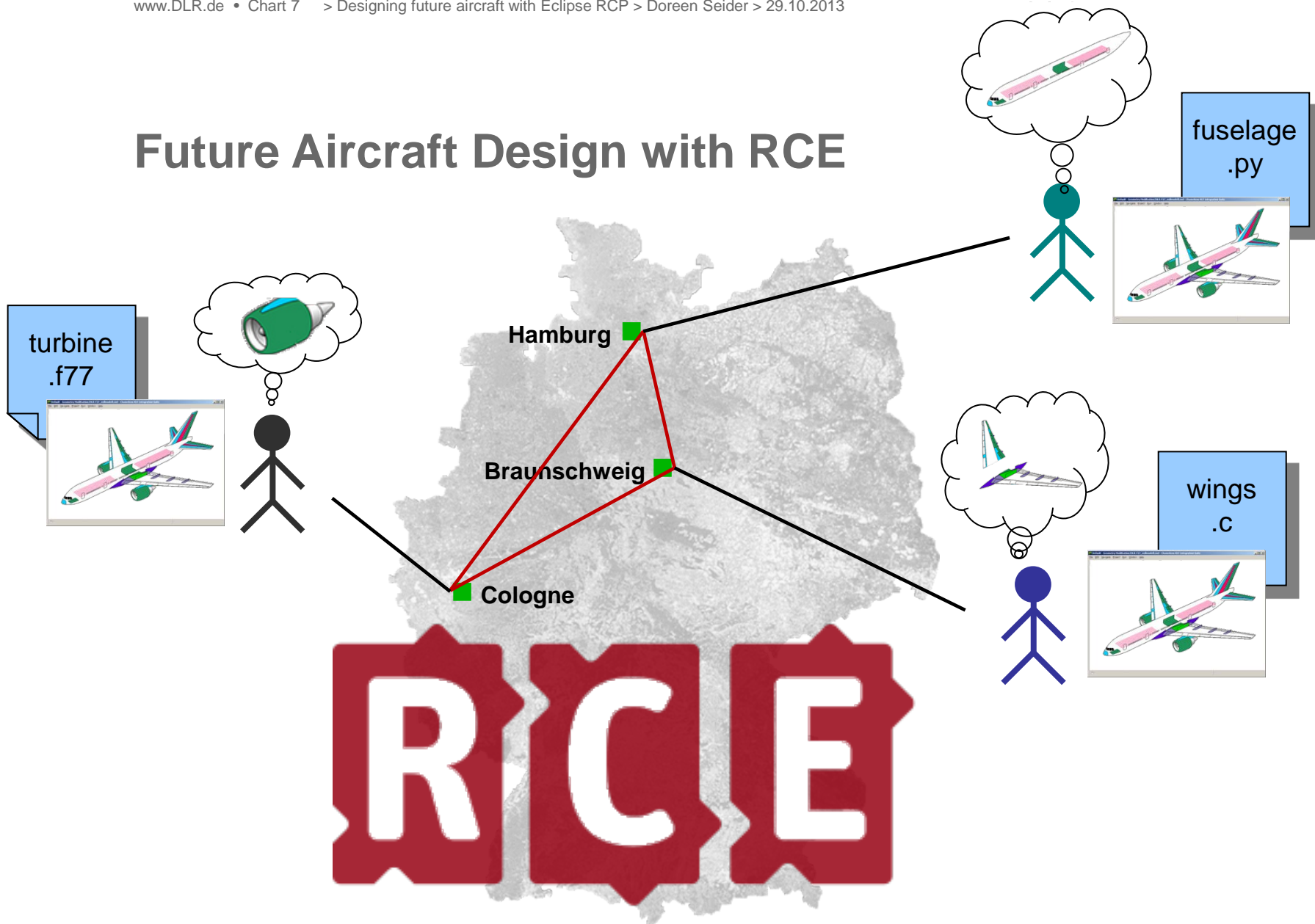


Why Eclipse RCP?

- Decision was made in 2006 as the development of RCE had started
- Reason was (mainly) OSGi, providing a component model, which
 - Sounded promising
 - Was standardized
- On a second note, it was important that basic „stuff“ can be re-used and don't needed to be implemented from scratch



Future Aircraft Design with RCE



Graphical User Client of RCE

The screenshot displays the RCE (Rapid Control Execution) graphical user interface. The main workspace shows a workflow diagram with the following components and connections:

- Input Wing** feeds into **Wing**.
- Wing** feeds into **Mesh** and **Optimizer**.
- Mesh** feeds into **Extract**.
- Extract** feeds into **AeroCluster**.
- AeroCluster** feeds into **Wing**.
- Wing** feeds into **Control**.
- Control** feeds into **Optimizer** and **Merger**.
- Optimizer** feeds into **Control**.
- Control** feeds into **Mass**.
- Mass** feeds into **Merger**.
- Merger** feeds into **Converger**.
- Converger** feeds into **Control**.

The interface includes a **Project Explorer** on the left showing the project structure, a **Palette** on the right with various tool icons, and a **Workflow List** at the bottom left. The **Workflow List** contains the following data:

Name	Status
MDO_V1.3_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	CANCELED
MDO_V1.1_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	RUNNING
MDO_V1.2_2013-10-21_1...	FINISHED
MDO_v1.0_2013-10-21_1...	PAUSED

The **Optimizer** window at the bottom right displays a plot titled **MDO_V1.2_2013-10-21**. The plot shows the function $f(x)$ versus x . The x-axis ranges from 0.05 to 1.18, and the y-axis ranges from 0.86 to 0.97. Two curves are shown: a blue curve and a green curve. The blue curve starts at approximately (0.05, 0.86) and peaks at (0.4, 0.97). The green curve starts at approximately (0.05, 0.86) and peaks at (0.8, 0.97).



Graphical User Client of RCE

Couple aircraft design tools to executable workflows

The screenshot displays the RCE (Rapid Configuration Environment) graphical user interface. The main workspace shows a workflow diagram with nodes: Merger, Mass, Input Wing, Control, Optimizer, Wing, AeroCluster, Mesh, Extract, and Converger. A red box highlights the workflow diagram and the right-hand palette. The palette includes options like Select, Draw Connection, Open Connection Editor, Data Flow, Execution, Excel, Script, Simple Wrapper, Studies, Under Test, Cluster, Design of Experiments, and User Integrated Tools. Below the workflow diagram, there is a 'Workflow List' table and a 'Property Value' table. The 'Workflow List' table shows the status of various workflow instances. The 'Property Value' table shows the configuration for the selected workflow. A line graph titled 'MDO_V1.2_2013-10-21' is displayed, showing two data series (blue and green) plotted against x and f(x).

Name	Status
MDO_V1.3_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	CANCELED
MDO_V1.1_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	RUNNING
MDO_V1.2_2013-10-21_1...	FINISHED
MDO_v1.0_2013-10-21_1...	PAUSED

Property	Value
showLegend	false
showTitle	true
title	MDO_V1.2_2013-10-21_...
traces	Trace [2]
XAxis	XAxis [1]
YAxis	YAxis [1]



Graphical User Client of RCE

Extend RCE with external aircraft design tools and publish them for others

The screenshot displays the RCE software interface. The main workspace shows a workflow diagram with the following nodes and connections:

- Input Wing** connects to **Wing**.
- Wing** connects to **Mesh** and **Optimizer**.
- Mesh** connects to **Extract**.
- Extract** connects to **AeroCluster**.
- AeroCluster** connects to **Wing**.
- Wing** connects to **Control**.
- Control** connects to **Optimizer** and **Merger**.
- Optimizer** connects to **Control**.
- Control** connects to **Mass**.
- Mass** connects to **Merger**.
- Merger** connects to **Converger**.

The right-hand side features a **Palette** with the following items:

- Select
- Draw Connection
- Open Connection Editor
- Data Flow
- Execution
- Excel
- Script
- Simple Wrapper
- Studies
- Under Test
- Cluster
- Design of Experiments
- User Integrated Tools
- Mass
- Wing

At the bottom, the **Workflow List** table shows the following data:

Name	Status
MDO_V1.3_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	CANCELED
MDO_V1.1_2013-10-21_1...	FINISHED
MDO_V1.3_2013-10-21_1...	RUNNING
MDO_V1.2_2013-10-21_1...	FINISHED
MDO_v1.0_2013-10-21_1...	PAUSED

The **Graph** titled "MDO_V1.2_2013-10-21" plots $f(x)$ against x . The x-axis ranges from 0.05 to 1.18, and the y-axis ranges from 0.86 to 0.97. Two curves are shown: a blue curve peaking at approximately 0.97 and a green curve peaking at approximately 0.96.



Graphical User Client of RCE

See results of workflow runs

The screenshot displays the RCE graphical user interface with several key components:

- Project Explorer:** Shows a project named 'MDO Project' containing workflow files MDO_V1.0.wf through MDO_V1.4.wf.
- Workflow Data Browser:** A tree view showing workflow runs for 'AeroCluster', 'Wing', and 'By Component'. It lists multiple instances of 'Workflow: MDO_V1.2_2013-10-21_08:42:43' and other related workflows.
- Workflow Console:** Displays the output of an optimizer. The message includes:

```
<<<< Function evaluation summary (interface1): 16 total (16 new, 0 dupl...
<<<< Best parameters =
      4.09375000000e-001 TR
<<<< Best objective function =
      -9.74300000000e-001
<<<< Best data captured at function evaluation 12
<<<< Iterator coliny_cobyla completed.
<<<< Single Method Strategy completed.
DAKOTA execution time in seconds:
  Total CPU = 3.851 [parent = 3.851, child = 0]
  Total wall clock = 3.85122
Optimization successfull!!
```
- 3D Model:** A blue aircraft model is shown in a 3D view, with a 'Mesh' button and 'Mass'/'Wing' options visible.
- Optimizer Chart:** A line graph titled 'MDO_V1.2_2013-10-21' showing two data series (blue and green) over a range of 'x' from 0.05 to 1.18. The y-axis ranges from 0.86 to 0.88.



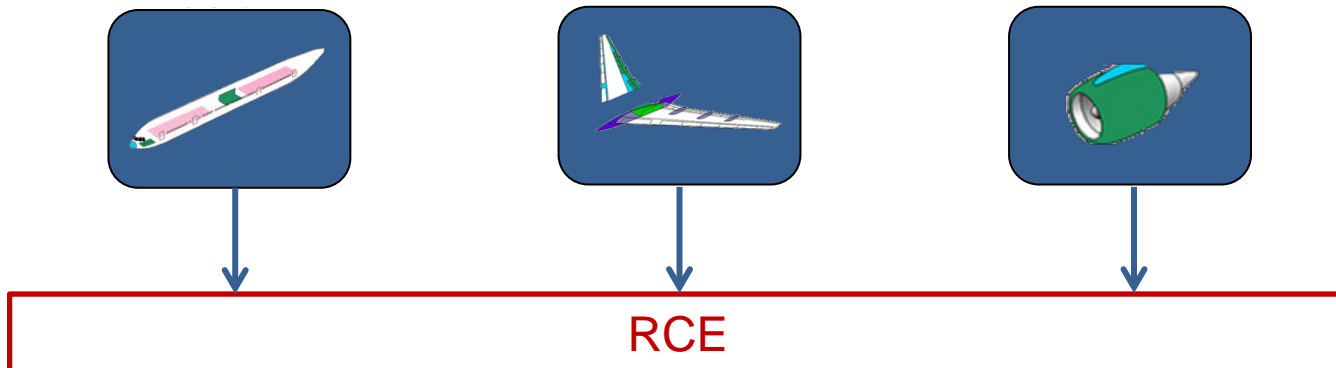
RCE Enables Multidisciplinary Collaboration

- RCE enables coupling of distributed aircraft design tools to a workflow
- Requires integration of aircraft design tools into RCE at runtime
 - Modularity and dynamic
 - OSGi: “Set of specifications that define a dynamic component system for Java”



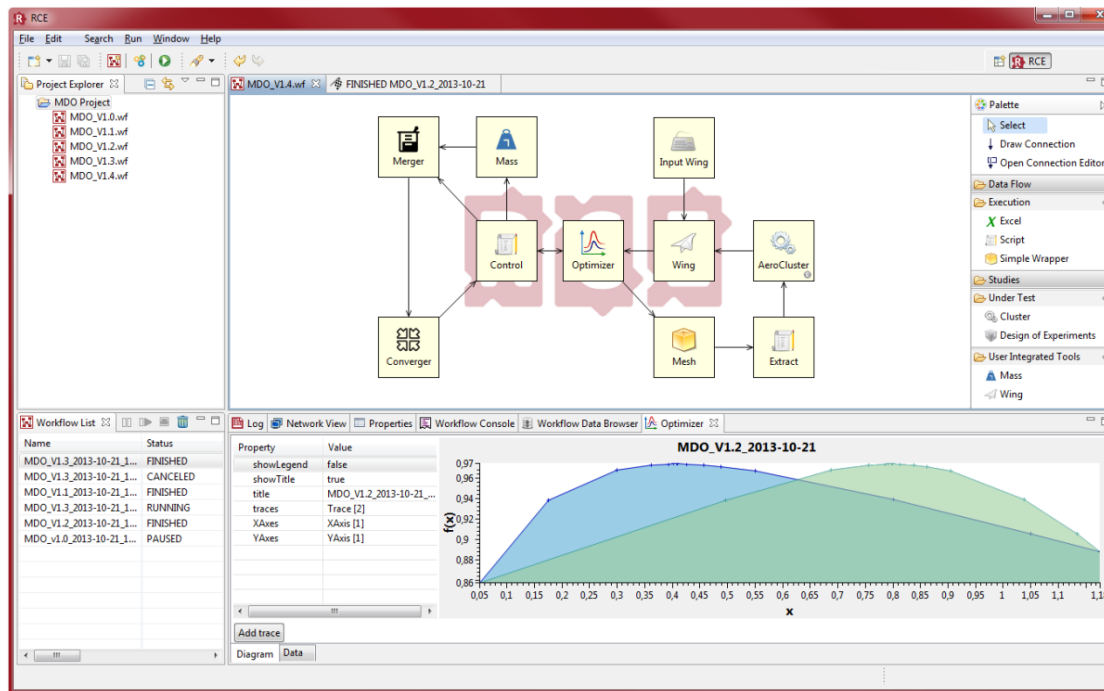
Using OSGi to Integrate Tools at Runtime

- A tool is (un-)registered as OSGi service once a specific configuration file is dropped into (removed from) a pre-defined folder
- OSGi service registry serves as aircraft design tool registry



Experiences with Usability Regarding Eclipse RCP

- Rich Client Platform helps us to make RCE more usable by adopting existing design decisions made for Eclipse RCP



Who uses RCE?

- Scientists and aerospace engineers
- Persons who are
 - no software developers but develop software
 - smart and love their work
 - wearing suits at conferences and workshops ;)



Who uses RCE?



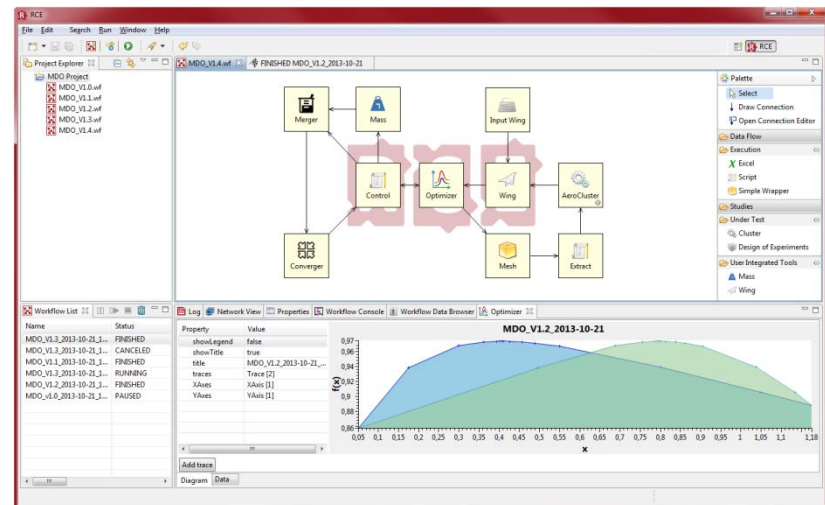
Experiences with Usability Regarding Eclipse RCP

- In terms of usability, users are divided into Eclipse IDE users and Non-Eclipse IDE users
- Some usability concepts we started with, worked out well for first user group and didn't work out at all for second one
- Two examples...



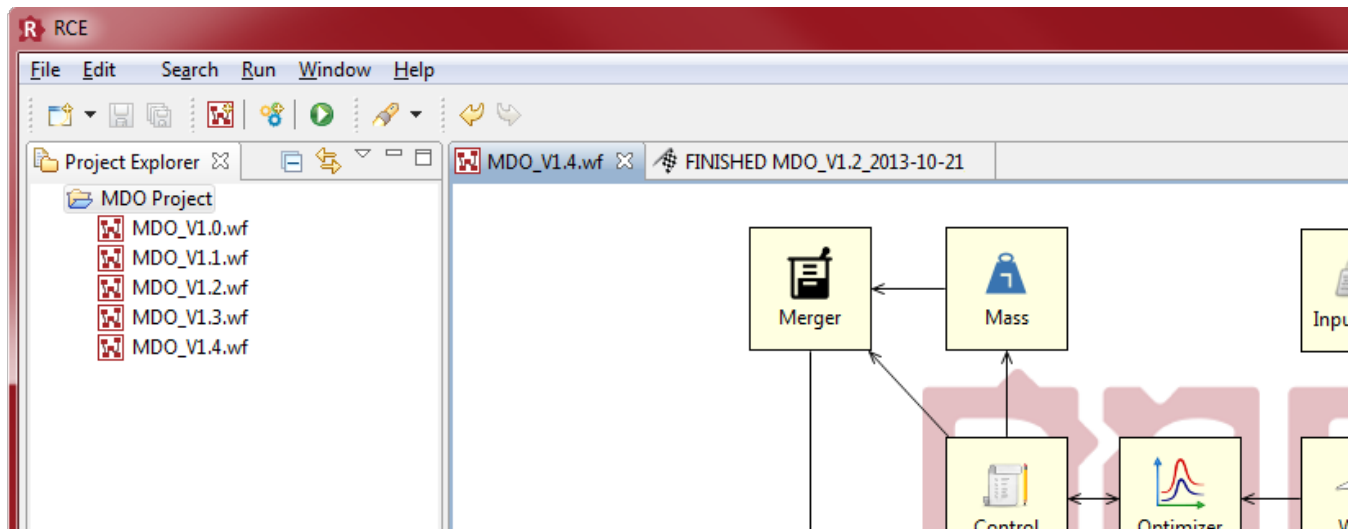
Views and Perspectives Concept

- Non-Eclipse IDE users get confused and lost, the others like the power of perspectives
- May be a question of training courses
- Our approach: We reduced the perspectives to exactly one and open all relevant views by default



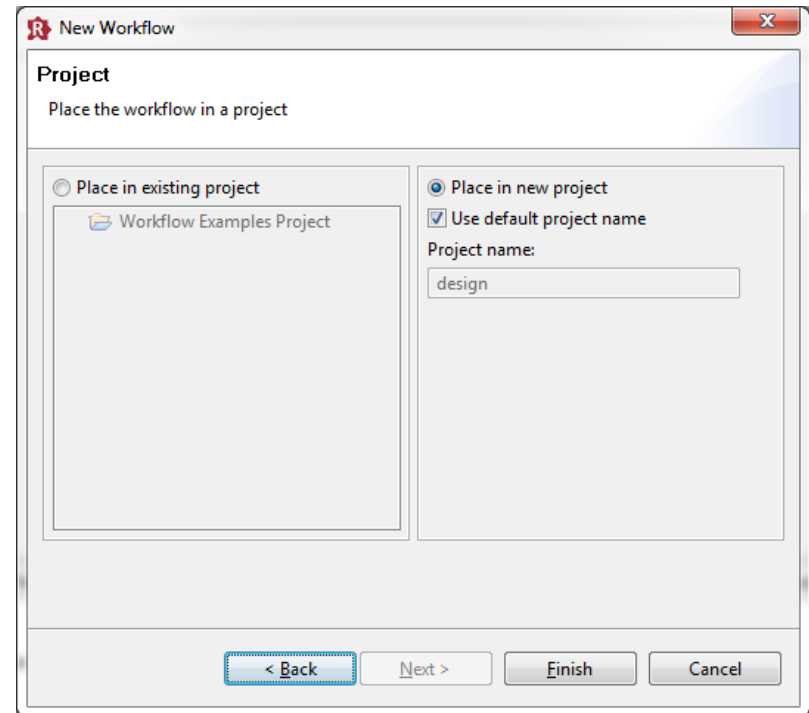
Project-based Concept

- Every workflow is a .wf file in the project explorer
- Non-Eclipse IDE users get lost if they want to create a workflow for the first time – why must I create a project (first) if I want a workflow?



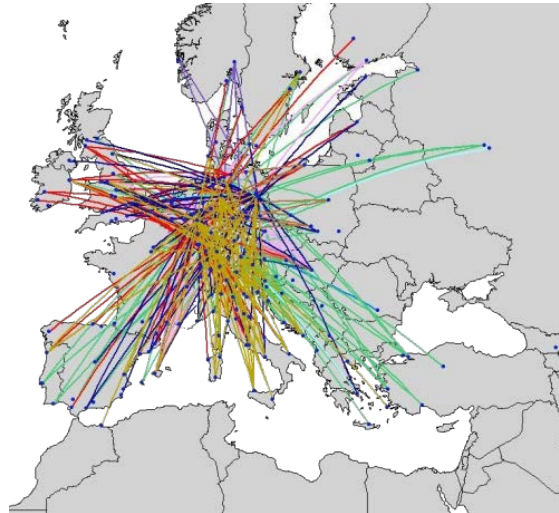
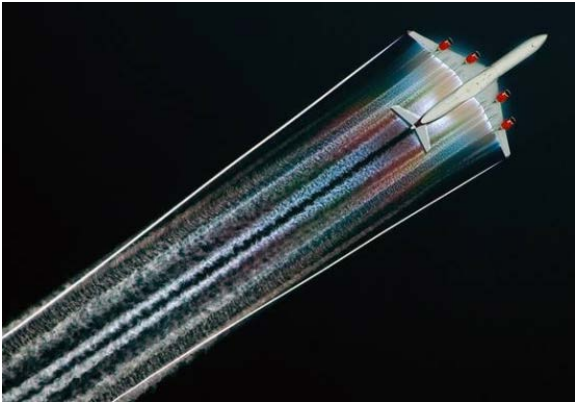
Project-based Concept

- We conducted a user study within a master thesis regarding the workflow creation task
- It was fun and very helpful
- Result: Dedicated workflow wizard „hiding“ the project creation



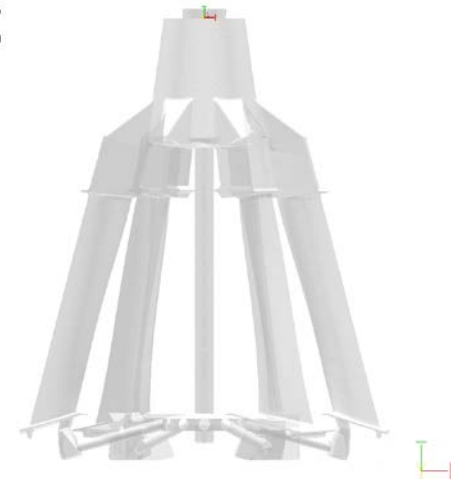
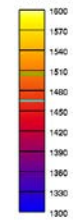
Managing different Distributions of RCE

- RCE has different applications with wide range of requirements



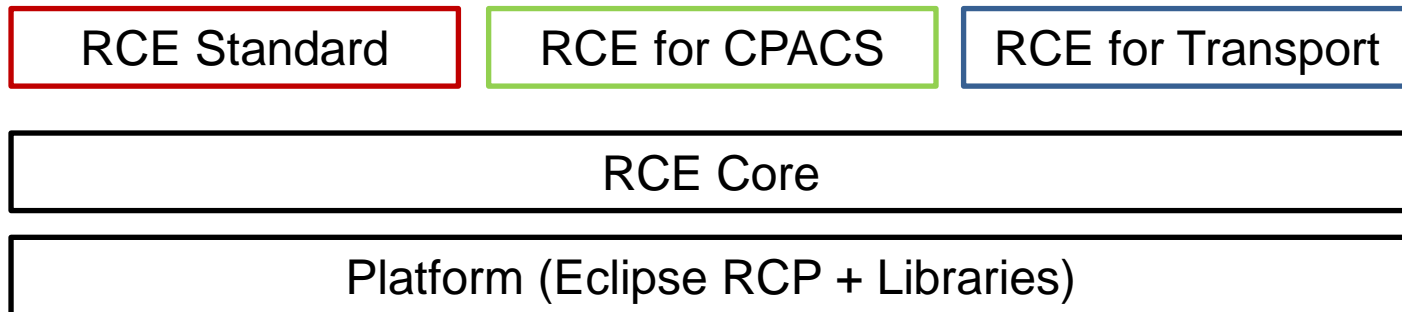
Time [s]: 0.000000

Temperature [°C]



Managing different Distributions of RCE

- We release three different distributions of RCE to have a minimalist distribution for each application
- We used Eclipse IDE distributions as guide line, but: no release train



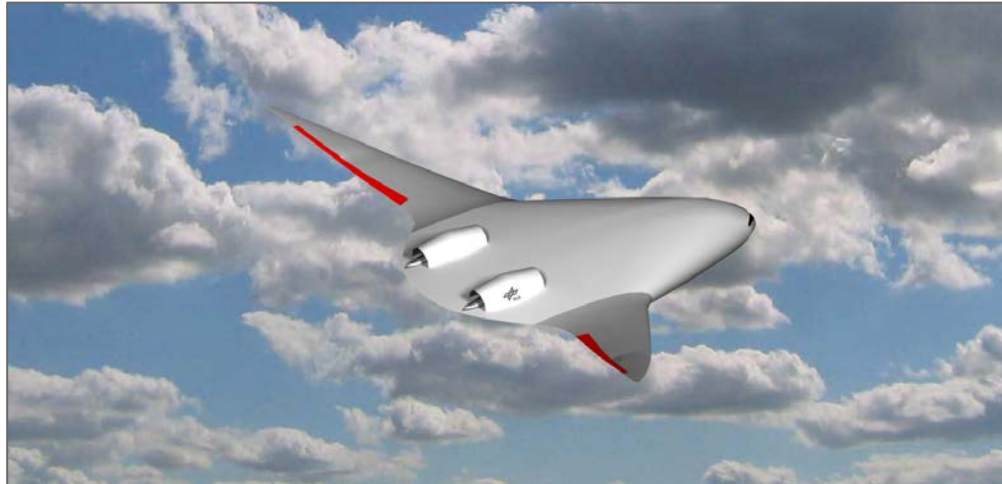
Managing different Distributions of RCE

- p2 infrastructure helps us
 - to compose the distributions, which share common code base
 - to provide a built-in update mechanism with less effort
- We build with Tycho
- Setting up all the stuff was a process...



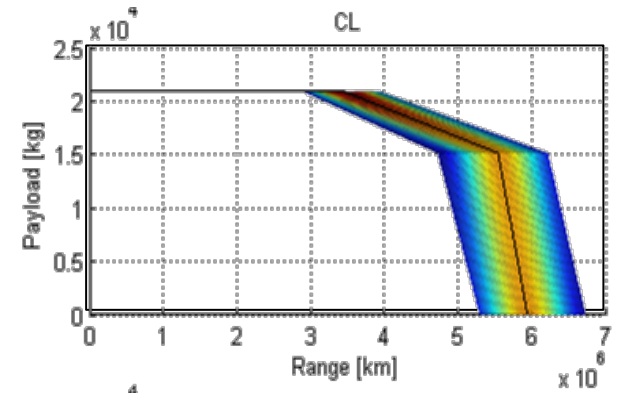
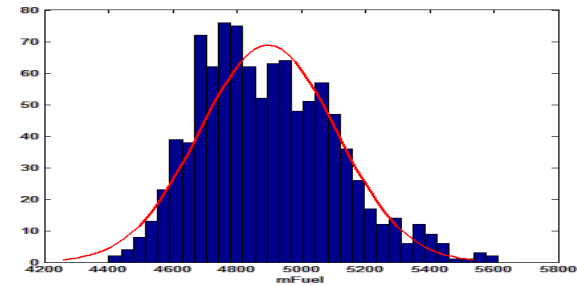
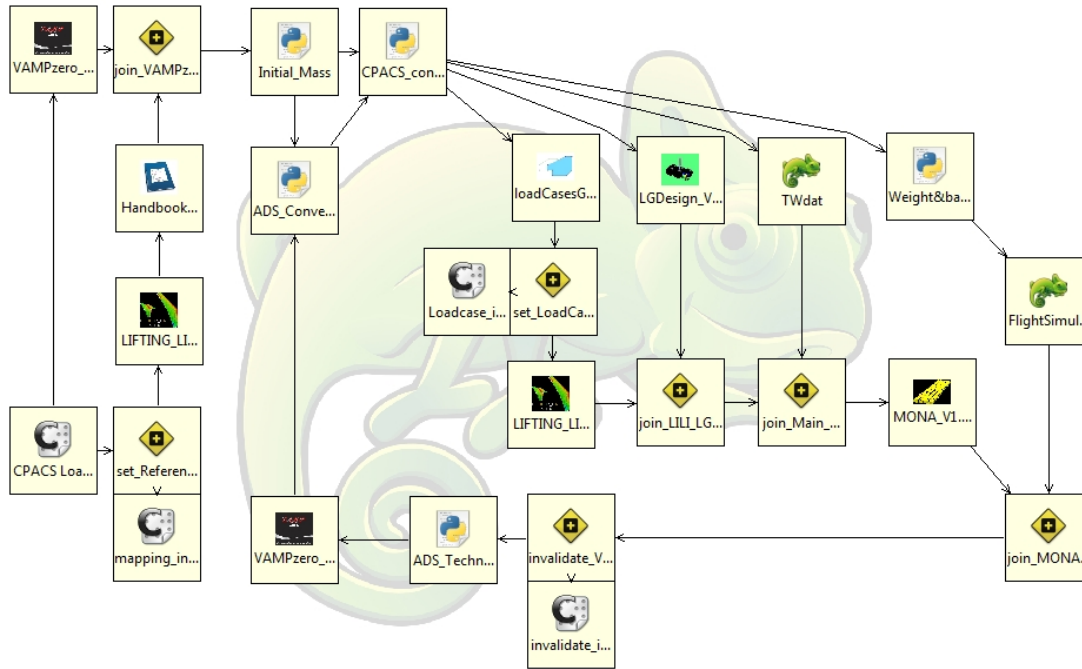
Current Projects at DLR: FrEACs

- Future Enhanced Aircraft Configurations
- Evaluate new aircraft configurations such as the blended wing body aircraft



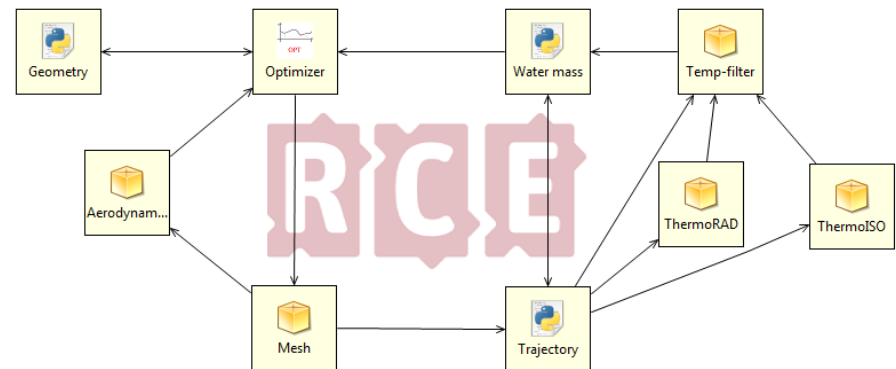
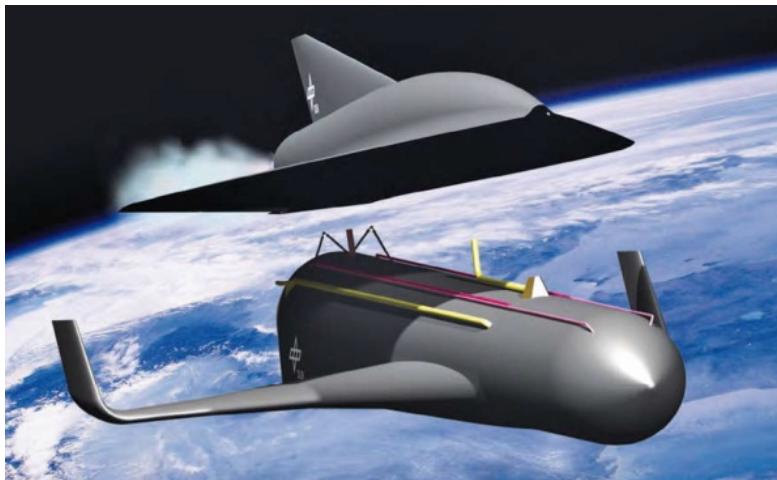
Current Projects at DLR: FrEACs

- Consider uncertainties in the workflow: How reliable are the results?



Current Projects at DLR: THERMAS

- Device thermal protection system of the SpaceLiner during atmospheric re-entry
- SpaceLiner is innovative concept between aviation and space travel for ultra fast passenger transport: Europe - Australia in 90 min



Summary

- Eclipse RCP helps significantly to design future aircraft
- Underlying OSGi enables the integration of external aircraft design tools
- Eclipse RCP enforces the development of usable software
- Extensible character, p2, and Tycho allows minimalist distributions
- Not all good concepts of the software engineering world can be adopted for scientists and aerospace engineers
 - ...but that makes developing RCE so interesting :)





Doreen Seider

German Aerospace Center (DLR), Simulation and Software Technology, Cologne