

Wissenschaft & Open Source – It's Complicated

Andreas Schreiber, Carina Haupt

Deutsches Zentrum für Luft- und Raumfahrt e.V.
Abteilung Intelligente und Verteilte Systeme
Köln, Berlin

FrOSCon 12, St. Augustin



Wissen für Morgen



Vorstellung



**Deutsches Zentrum
für Luft- und Raumfahrt**
German Aerospace Center

Simulations- und Softwaretechnik, Köln/Berlin
Abteilungsleiter Intelligente und verteilte Systeme

Institut für Datenwissenschaften, Jena
Gruppenleiter für Secure Software Engineering



**Co-Founder
Data Scientist
Patient**



Das DLR Deutsches Zentrum für Luft- und Raumfahrt



- Forschungseinrichtung
- Raumfahrt-Agentur
- Projektträger



Standorte und Personal

Circa 8.000 Mitarbeiterinnen und Mitarbeiter arbeiten in 39 Instituten und Einrichtungen in 20 Standorten.

Büros in Brüssel, Paris, Tokio und Washington.



Free and Open Source Software

*„Free as in ‘freedom’,
not as in ‘free beer’“*

Softwareentwicklung im DLR

Einige Zahlen...

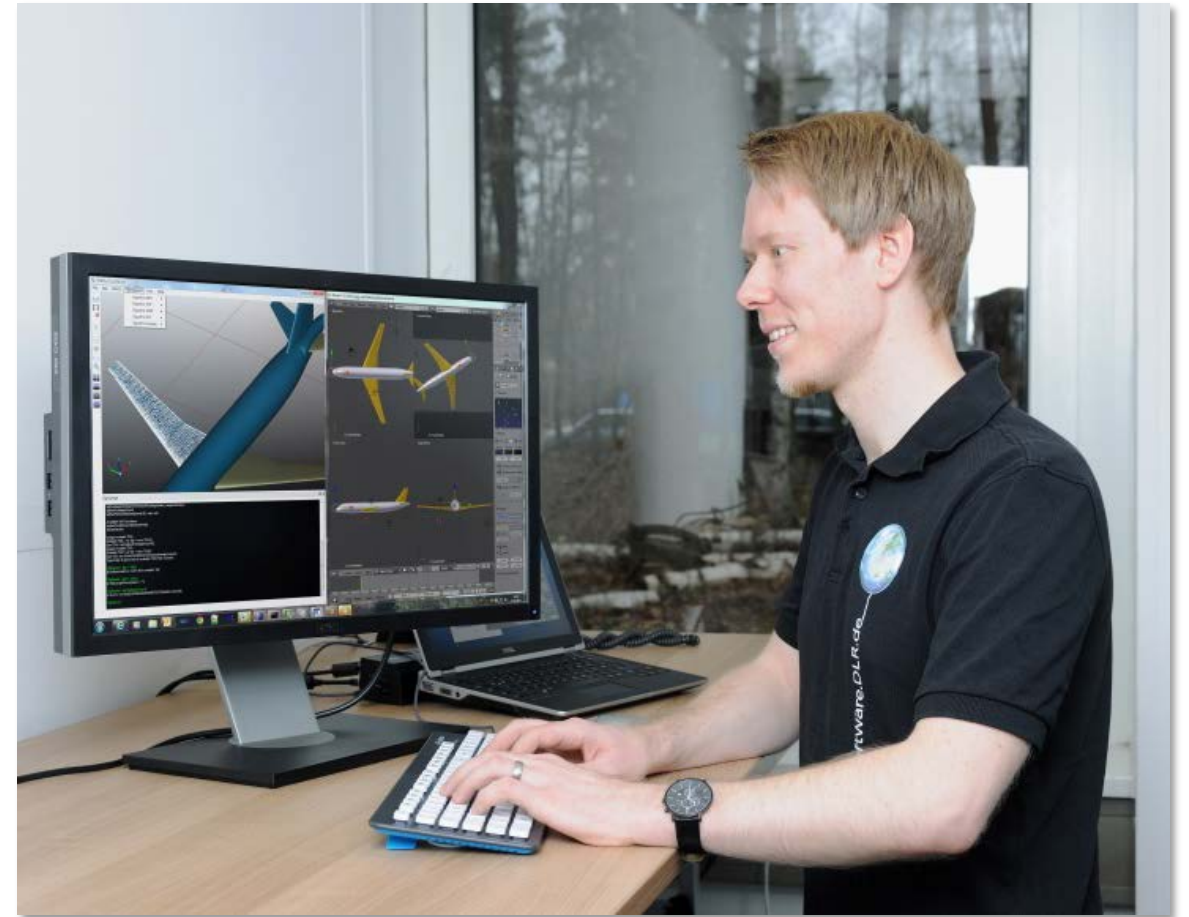
- Mehr als 1500 Mitarbeiter entwickeln Software
- Mehr als 150 Mio EUR Personalkosten pro Jahr für Softwareentwicklung
- DLR ist eines der größten „Softwarehäuser“ in Deutschland



Softwareentwicklung im DLR

Charakteristiken

- Im DLR entwickelte Software ist in der Regel Nicht-Standard-Software
 - Individualsoftware
 - Oft spezielle und spezifische Anforderungen
- Die „Entwickler“ haben oft keine Ausbildung in Software-Entwicklung



Softwareentwicklung im DLR

Vielfalt

- Vielzahl an eingesetzten Software-Technologien
 - z.B. über 30 Programmiersprachen
- Sowohl Open-Source-Software als auch proprietäre Software
 - Viele verschiedene Lizenzen
- Große Anzahl an Software-Projekten
 - Überblick über existierende Software ist schwierig



The screenshot shows the website software.DLR.de. The header includes the DLR logo and a search bar. The main content area lists several software projects:

- BACARDI**: Knowledge and Data Management. The Backend Catalog for Relational Debris Information (BACARDI) is the DLR's approach to a space debris database. The custom middleware components are implemented in Python using ZeroMQ and Protocol Buffer technology.
- Simulation Model Library**: Simulation and Modeling. Simulation Model Library (SimMoLib) is a distributed system to manage a library of simulation models. SimMoLib's main goal is to promote the preservation of knowledge that lies in simulation and calculation models and encourage reuse of those models.
- Virtual Satellite**: Simulation and Modeling. Designing space systems and planning space missions relies on many separated phases and disciplines. The virtual satellite aims at closing the gaps in the development life-cycle and between disciplines by using model-based systems engineering.
- SUMO**: Simulation and Modeling. SUMO is an open source, highly portable, microscopic and continuous road traffic simulation package designed to handle large road networks.

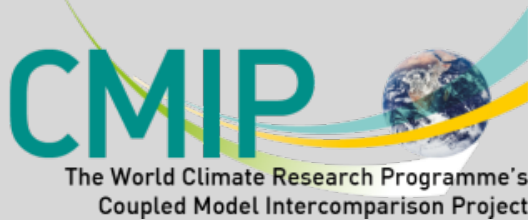
The footer of the website identifies it as the German Aerospace Center (DLR).

Beispiele für Open-Source-Software aus dem DLR

Entwickelt und genutzt gemeinsam mit externen Partnern

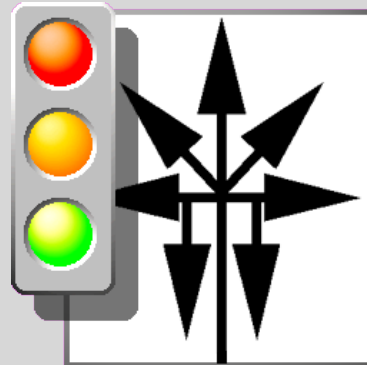
ESMValTool

- Earth System Model eValuation Tool
- www.esmvaltool.org



SUMO

- Simulation of Urban Mobility
- www.sumo.dlr.de

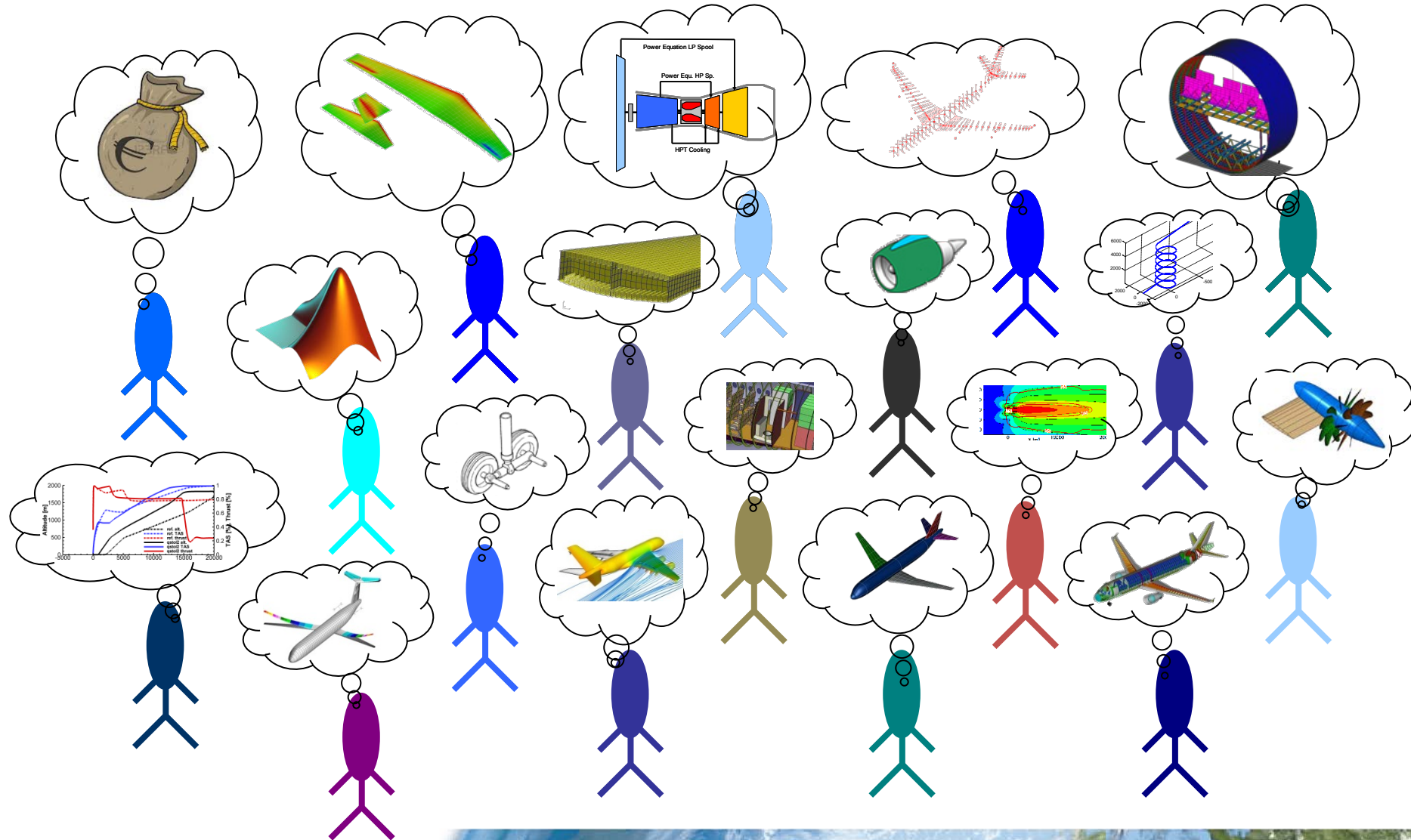


RCE

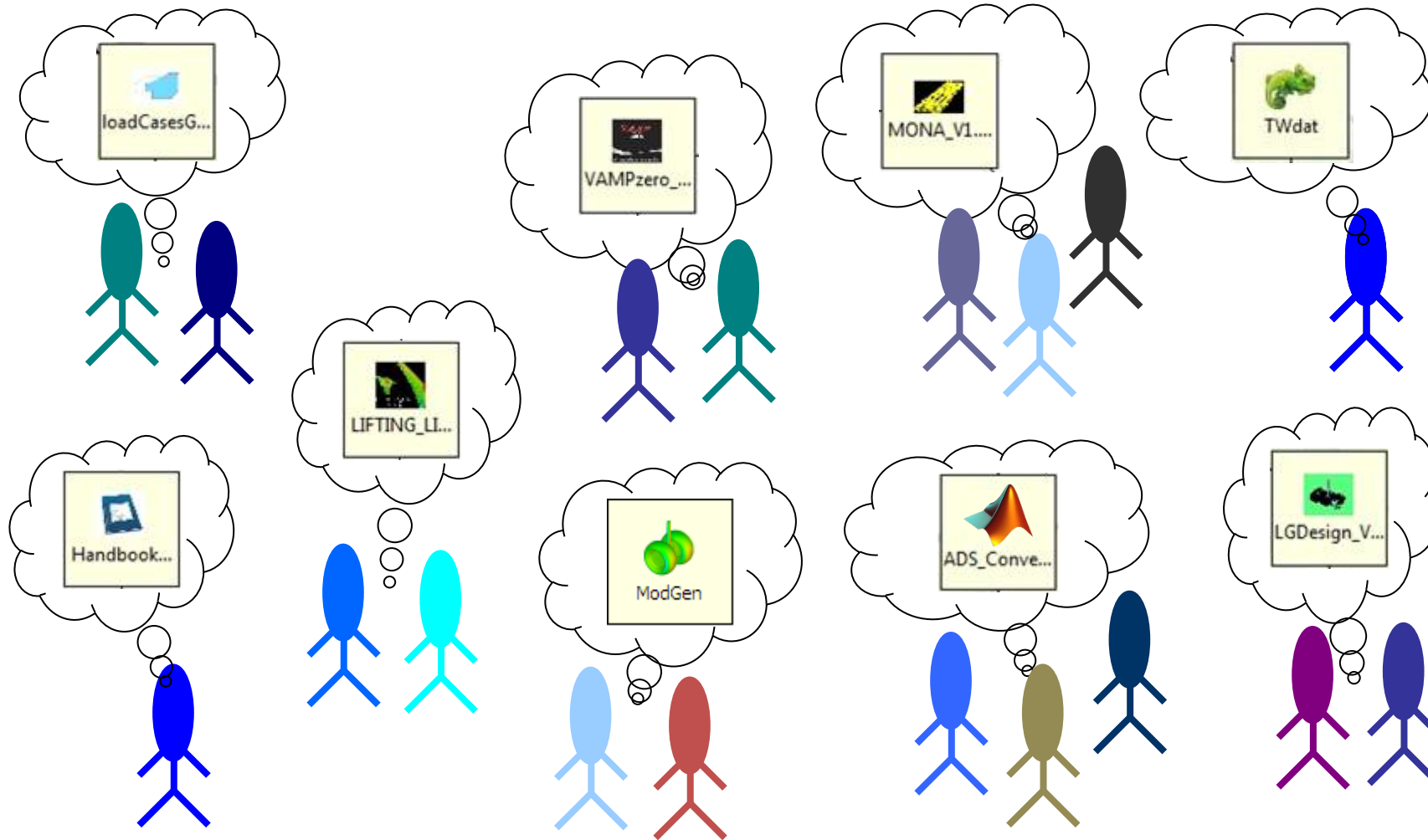
- Remote Component Environment
- rcenvironment.de



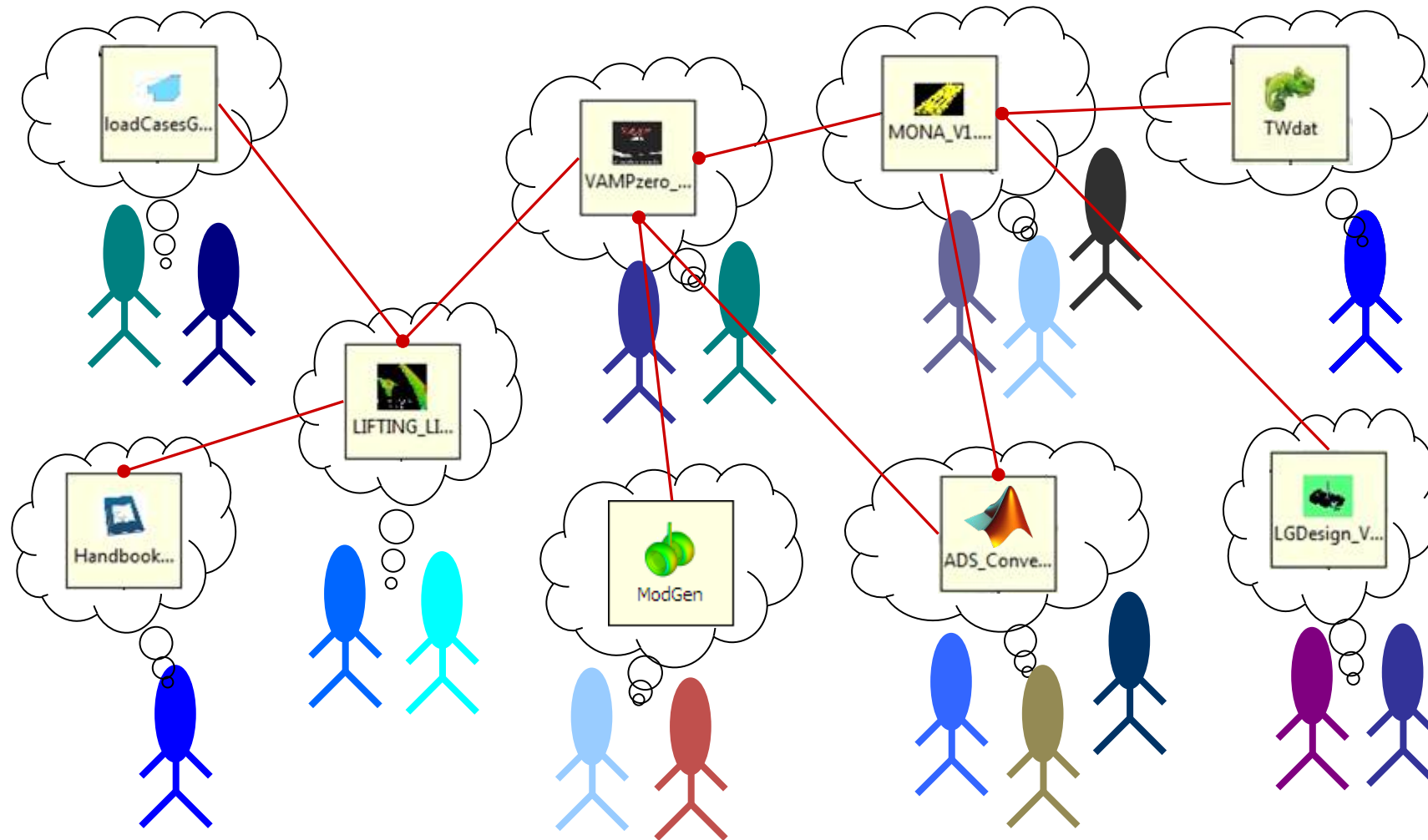
Wissen in den Köpfen



Wissen als Software



Zusammenarbeit



Verteilte Entwurfsprozesse

The screenshot displays the RCE software interface, which is used for managing distributed design processes. The main window shows a workflow diagram with the following components and connections:

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

The **Workflow List** table shows the status of various workflow instances:

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 displays a graph titled **MDO_V1.2_2013-10-21** showing 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. The graph shows two overlapping curves: 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).

Gemeinsamer Entwurf



Probleme der Vergangenheit

- Software wurde mit Lizenzproblemen nach Außen gegeben
- Lizenzkompatibilität der verwendeten Software war nicht erfüllt
- Kollegen wussten nicht, welche Open-Source-Lizenz sinnvoll ist
- Kollegen wussten nicht, welche Bedingungen sie im Source Code erfüllen müssen



Entwickler vs. Lizenzrecht

Ergebnis einer aktuellen Studie:

*“... we conducted a survey that posed development scenarios involving three popular open source licenses (GNU GPL 3.0, GNU LGPL 3.0 and MPL 2.0) both alone and in combination. The 375 respondents to the survey, who were largely developers, gave answers consistent with those of a legal expert's opinion in 62% of 42 cases. **Although developers clearly understood cases involving one license, they struggled when multiple licenses were involved.**”*

Daniel A. Almeida, Gail C. Murphy, Greg Wilson, and Mike Hoyer. 2017.

Do software developers understand open source licenses?.

In *Proceedings of the 25th International Conference on Program Comprehension (ICPC '17)*.

IEEE Press, Piscataway, NJ, USA, 1-11. DOI: <https://doi.org/10.1109/ICPC.2017.7>



Warnungen und Hinweise zu Open Source (2012)

1. Open-Source-Lizenzbedingungen haben grundsätzlich Rechtsgültigkeit, d.h. ein Verstoß gegen diese Bedingungen stellt eine Urheberrechtsverletzung dar.
2. Die unbefugte Weitergabe von Open-Source-Software an Dritte (z.B. i.R.v. Aufträgen) kann zu gravierenden Haftungstatbeständen führen. Das gleiche gilt, wenn im Rahmen von Drittmittel-Projekten verschwiegen wird, dass vom DLR entwickelte Software Open-Source-Komponenten enthält.
3. Open-Source-Software wirkt in der Regel infizierend: Abhängig von den Lizenzbedingungen (z.B. GPL) kann bei einer engen Koppelung von selbst entwickelter Software mit von Dritten entwickelter Open-Source-Software der Effekt eintreten, dass die Lizenzbedingungen der Open-Source-Software für die gesamte Software gelten müssen.
4. Open-Source-Lizenzbedingungen sind untereinander teilweise inkompatibel, d.h., bestimmte Open-Source-Komponenten können nicht miteinander kombiniert werden, ohne gegen eine der Lizenzbedingungen zu verstoßen.



Maßnahmen im DLR

Informationen und Schulungen

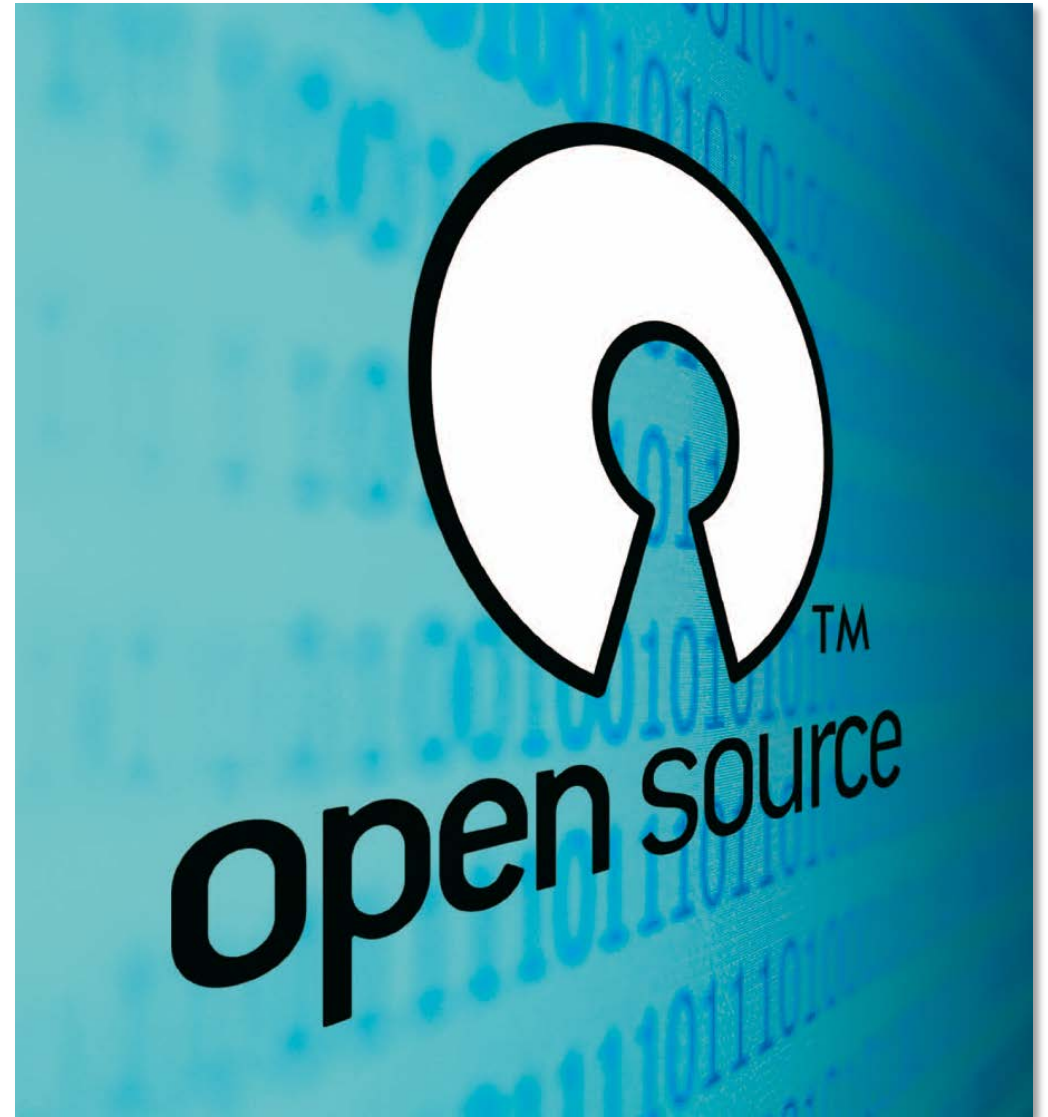
- Lizenzrechtsschulung
- Broschüre

Wissensaustausch

- Wikis
- Wissens-Austausch-Workshops

Beratung und Unterstützung

- Empfohlene Lizenzen
- Support



Informationen und Schulungen



Wissen für Morgen



Open-Source-Lizenzrechtsschulung

„Rechtliche Aspekte der Open-Source-Nutzung“

- Reguläre und regelmäßige Schulung im DLR-Bildungsprogramm
- Jeder DLR-Mitarbeiter kann teilnehmen
- Kurze Schulung (derzeit ca. 4 Stunden)
- Durchgeführt von zwei Personen
 - Jurist vom DLR-Technologiemarketing (Abt. Lizenzen und Patente)
 - Softwaretechniker von DLR-Einrichtung Simulations- und Softwaretechnik



Agenda der DLR Open-Source-Lizenzrechtsschulung

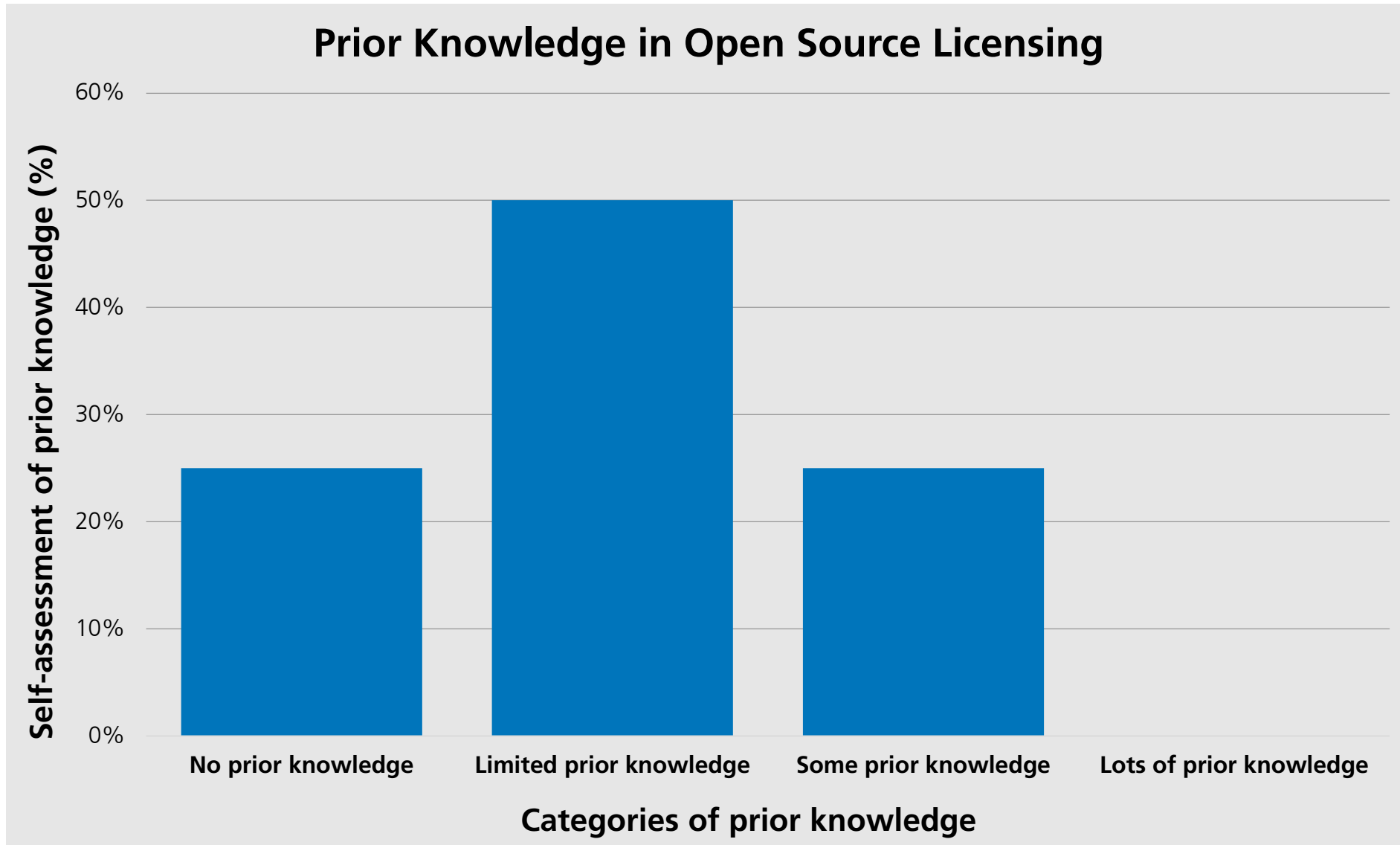
Einführung in Open Source	Rechtliche Grundlagen	Open Source Software	Ausgewählte Open-Source-Lizenzen	Nutzung von Open-Source-Software
<ul style="list-style-type: none"> • Open Source und Free Software • Open-Source-Philosophie • Open Source im DLR 	<ul style="list-style-type: none"> • Copyright • Urheberrecht • Vertragsrecht • Rechtewegfall • Rechtsfolgen • Eigentum 	<ul style="list-style-type: none"> • Lizenzarten von Software • Definition von „Open Source“ • Open-Source-Lizenzmodelle • Copyleft 	<ul style="list-style-type: none"> • GPL, AGPL, LGPL • Eclipse Public License • BSD License • Apache License • Artistic License 	<ul style="list-style-type: none"> • Nutzungsformen von Software • Open Source in eigenen Projekten • Lizenzkompatibilität • Entwicklung und Verteilung von Open Source



Schulungsteilnehmer

Year	Participants	DLR Site
2012	32	Oberpfaffenhofen
2013	23	Berlin, Köln
2014	12	Stuttgart
2015	9	Berlin
2016	14	Köln



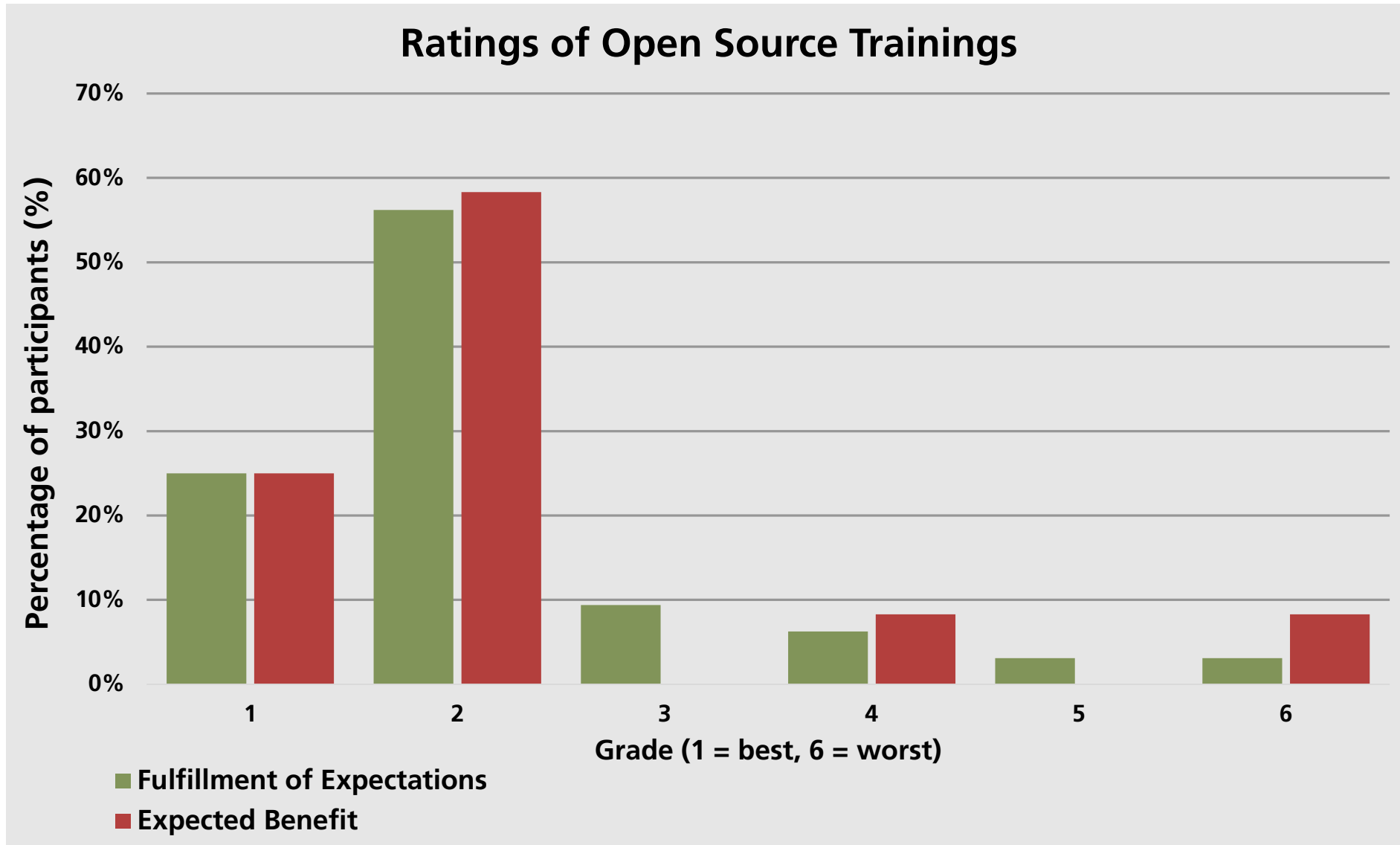


Erwartungen der Teilnehmer

Erwartungsabfrage vor den Schulungen (*Auszug*)

- Lernen, welche Aspekte in Projekte zu berücksichtigen sind
- Einen Überblick über Open-Source-Lizenzmodelle bekommen
- Lernen, wie man Open-Source-Software in Closed-Source-Projekten nutzen kann
- Lernen, wie man Software unter Nutzung von Open Source nutzt
- Beispiele aus dem DLR kennenlernen
- Rechtliche Grundlagen kennenlernen
- Lernen und verstehen der persönlichen Konsequenzen bei Lizenzvergehen





Open-Source-Broschüre

Lizenzrecht

- Grundlagenwissen über Open-Source-Lizenzrecht
- Entwickelt durch Anwaltskanzlei
- Finanziert und Herausgegeben vom DLR-Technologiemarketing



Open-Source-Broschüre

Struktur des Inhalts

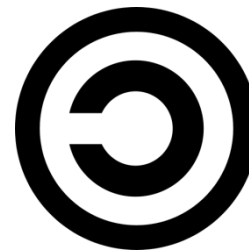
Zwei große Kapitel

- Verteilung von *unmodifiziertem* Code
- Verteilung von *modifiziertem* Code

In beiden Kapiteln: Auflistung aller Verpflichtungen bei den Lizenzen, die im DLR am weitesten verbreitet sind

Lizenzen sind gruppiert in Lizenzen

- mit starkem Copyleft
- mit schwachem Copyleft
- ohne Copyleft
- mit speziellen Rechten



Open-Source-Broschüre

Praktische Informationen

Checklisten

Infoboxen

Weitergabe veränderter Software

6 Lizenzen mit strengem Copyleft

GNU General Public License Version 2 (GPLv2)

- Mitliefern des Lizenztexts (siehe Merkbox 1, S. 10)
- Zugänglichmachung des Quellcodes (s.o. Merkbox 2, S. 10)
- Vollständiger korrespondierender Quellcode (s.o. Merkbox 3, S. 11)
- Urhebervermerk (s.o. Merkbox 4, S. 11)

Beibehalten der vorbestehenden Urhebervermerke (s.o. Merkbox 4, S. 11)
Anbringen neuer Urhebervermerke

Merkbox 11

Bei der Veränderung der Software sind vorbestehende Urhebervermerke beizubehalten (s.o. Merkbox 4, S. 11) und neue Urhebervermerke in den hinzugefügten Dateien anzubringen.

Wie gestalte ich einen neuen Urhebervermerk?

Im Header der Quellcode-Dateien ist der Vermerk „© [Jahreszahl], Deutsches Zentrum für Luft- und Raumfahrt e.V., author: [Name]“ anzubringen.

- Disclaimer (s.o. Merkbox 5, S. 11)
- Änderungsvermerk

in den geänderten Quellcode-Dateien.


Merkbox 12

Wie gestalte ich den Änderungsvermerk?

Der vorbestehende Urhebervermerk ist beizubehalten und ein kurzer Hinweis auf die hinzugefügte/geänderte Funktion mit Datumsangabe und Namensnennung des Deutschen Zentrums für Luft- und Raumfahrt e.V. sowie des Autors der Änderung.

Der Urhebervermerk kann dabei wie in Merkbox 11, S. 24 aussehen.

- Hinweispflichten bei interaktiven Kommandos

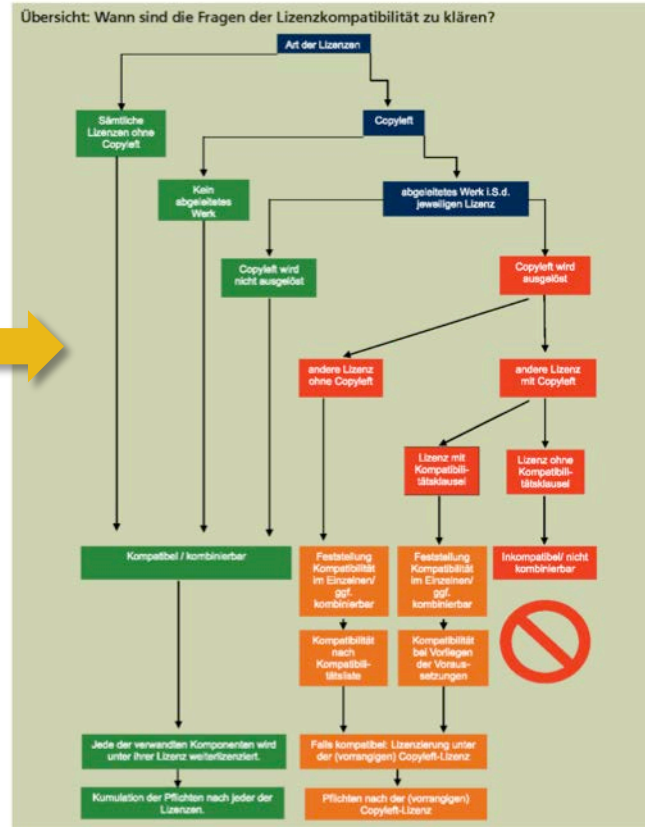
24 

Open-Source-Broschüre

Entscheidungshilfen

Entscheidungsbaum

11 Anhang



Wissensaustausch



Wissen für Morgen



Wikis

- Ursprünglich viele Wikis im DLR
- Seit 2013 zentrales DLR.**Wiki** auf Basis Atlassian Confluence
- Wiki-Bereiche für
 - Software Engineering
 - Veranstaltungen
 - ...

The screenshot shows the DLR.Wiki interface for the 'Software Engineering' section. The top navigation bar includes 'DLR.Wiki', 'Bereiche', 'Frage ans DLR', and 'Erstellen'. The main content area features a welcome message: 'Welcome to the SoftwareEngineering.Wiki! The SoftwareEngineering.Wiki is the place to create, share and discuss software engineering content with colleagues on a working-level! We aim for an open and constructive exchange of ideas. Therefore, feel free to share your knowledge and encourage others to do so as well! Before you start: Please visit the Get Involved! section and subscribe to our Blog! (If you are looking for further software engineering resources, please see our Software Engineering Intranet page or contact the Software Engineering Contact of your Institute!)'. Below this are four featured sections: 'Get Involved!' with a photo of people, 'Ask a Question' with a thinking emoji, 'Topics' with a grid of images, and 'Literature' with a stack of books. The right sidebar contains 'News-Einträge' and 'Latest Questions'.



Wiki für Open Source

Bereich im DLR.Open-Wiki mit

- Tools
- Literaturhinweisen
- Fragen und Antworten
- Veranstaltungen

Alles Informationen, die man auch durch (Bing|DuckDuckGo|Google)-Suche findet

Single-Point-of-Information im DLR

The screenshot shows the DLR Wiki page for 'Open Source'. The page title is 'Open Source' and it was created by Schlauch, Tobias, and last edited by Seth, Daniel on January 13, 2017. The main content area contains an introductory paragraph about Open Source Software (OSS) and a list of topics: 'Legal aspects of OSS licenses', 'Finding suitable OSS libraries', and 'Publication of OSS'. Below this, there are sections for 'Subtopics', 'Internal Discussions / Workshops', and 'Further Readings'. The 'Subtopics' section lists 'Open Source Hosting' and 'Open Source Licenses'. The 'Internal Discussions / Workshops' section lists a group discussion on OSS and Inner Source usage at DLR. The 'Further Readings' section lists a DLR-specific handbook, an internal course on legal aspects of OSS usage, and general questions about OSS. On the right side, there is a blue graphic with the 'open source' logo and a list of contributors including Haupt, Carina, Scholz, Michael (TS-DAT), Schreiber, Andreas, Mühlbauer, Martin, Knodt, Uwe, Schlauch, Tobias, Seth, Daniel, Bender, Florian, Bachmann, Arne, Brommund, Katharina, Brunner, Sebastian, Naß, Andrea, Riener, and Heinz. Below the contributors list is a 'Related Information' section with links to 'News...', 'Questions...', 'Literature...', and 'Events...'. At the bottom of the page, there is a comment box and a footer indicating the page is powered by Atlassian Confluence 5.10.6.



WissensAustauschWorkshops

Workshops zum Wissensaustausch zwischen DLR-Mitarbeitern (Peer-to-peer)

Es gibt WissensAustauschWorkshops (WAWs) zu unterschiedlichen Themen

- Software Engineering
- Visualisierung großer Datensätze
- Autonomes Fliegen
- Photonische Systeme
- DLR.Open (inkl. Open Source)

WissensAustauschWorkshop

DLR.open

18. - 19. Oktober 2016 in Köln-Porz, Casino

Jetzt kostenlos
anmelden

bis 21. September 2016
über das Bildungsprogramm
im Intranet

Kontakt:
Philipp Bergeron
☎ 4002

Mehr Details unter:
wissen.DLR.de/waw

Eine Initiative des
Wissensmanagements im DLR



WissensAustauschWorkshops

WAWs sind jeweils offen für DLR-Mitarbeiter aller Standorte und Institute

- Bis zu 60 Teilnehmer pro WAW

WAWs sind interaktiv

- Kurze Impulsvorträge im Plenum
- Persönliches Vorstellen
- Lightning Talks
- Arbeit/-Diskussion in Kleingruppen





Carina Haupt
BIETE

Software Engineering

ROSS

Python

Open Source

Suche

Tage

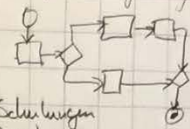
Michael Meinel, Simulations- und Softwaretechnik Software-Engineering

Biete

Software-Engineering

Prozesse

Werkzeuge



Schulungen
Beratung

● ● ● Python ... auch mit anderen Sprachen (F#)

↳ Fortran ... github.com/DLR-SC/F#

● Continuous Integration

Software-Tests — Werkzeuge und Methoden

Werkzeuge -f. Workflow-Überwachung und
-kontrolle (Wo läuft es?)

Stufe

- Bewertung (Was hilft?)

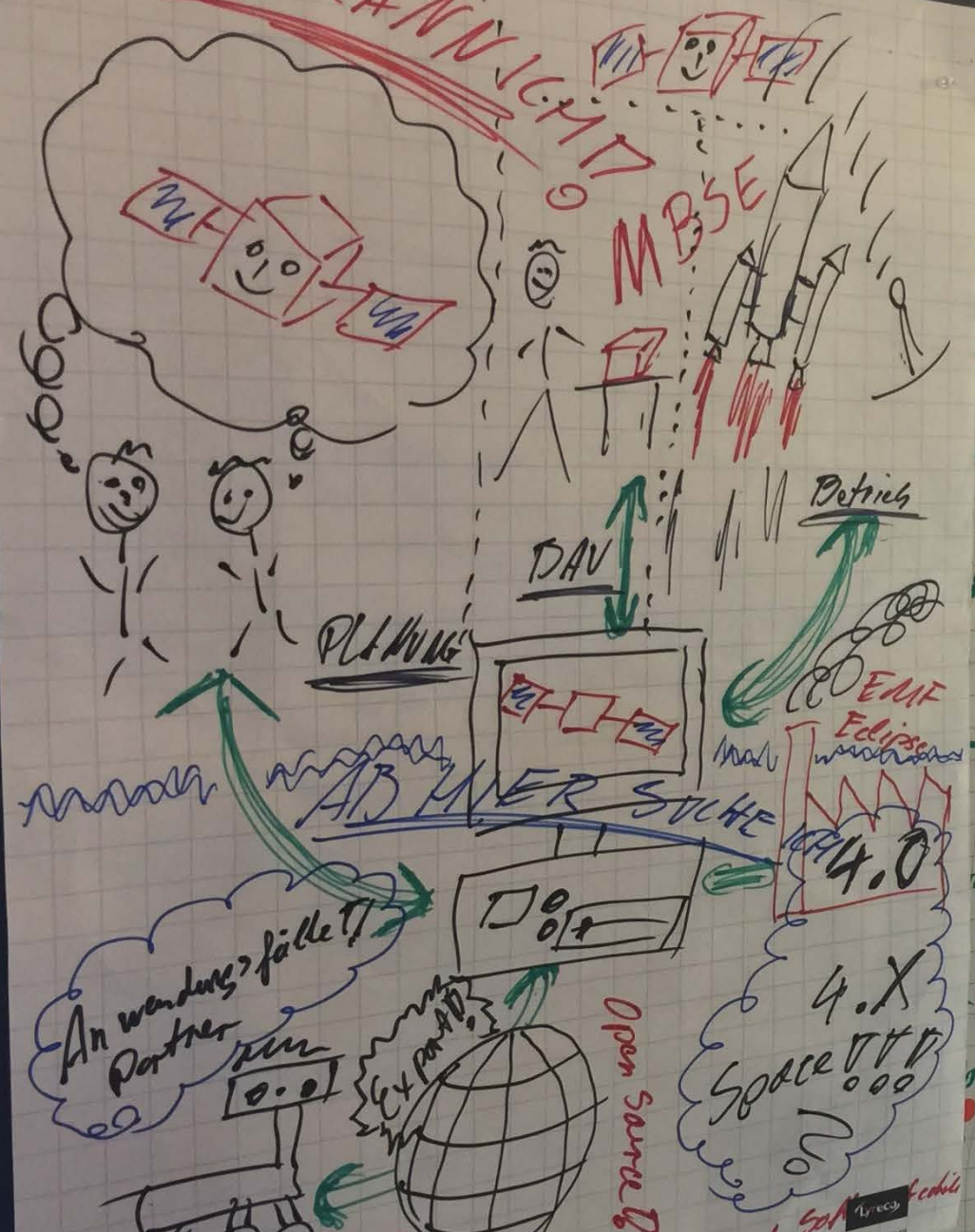
Sicherheit

Welche Gefahren gibt es? ● ●
Wie unterschiedlich ist sichere StD? ● ●

Projekte

f. Fortran - Anbindung an ... ● ●
Python, Java, C#, ...

DAS KANN NICHT

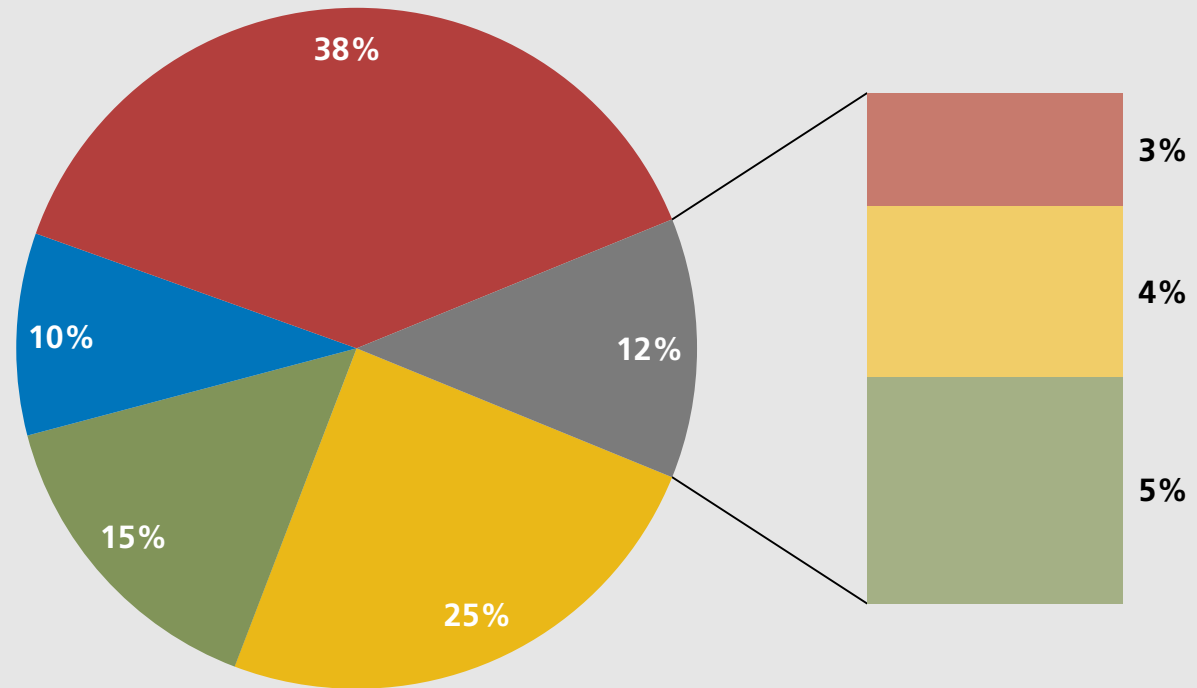


Leistung
ingen)
entwickl
in
nabricator"



Gesamtergebnis Erwartungen in Prozenten

- Netzwerken
- Open Source Prozess
- Rechtliche Informationen
- Informationsaustausch
- Maßnahmen ergreifen
- Sichtbarkeit
- Ersatz proprietärer Software



Erkenntnisse aus den WissensAustauschWorkshops

Generelle Erkenntnisse

- Open-Source-Software ist bereits weit verbreitet im DLR
- Open-Source-Software wird vorwiegend in Form freier Tools und Libraries verwendet
- Eigene Forschungssoftware ist bisher noch vorwiegend Closed Source
- Großes Interesse an Veröffentlichung der eigenen Software

Kritik gegenüber Open Source

- Man verliert Möglichkeiten, Gelder zu erwirtschaften
- Der Aufbau einer Open-Source-Community ist zu aufwendig
- Zusätzlicher Zeitaufwand ist zu hoch und dadurch demotivierend

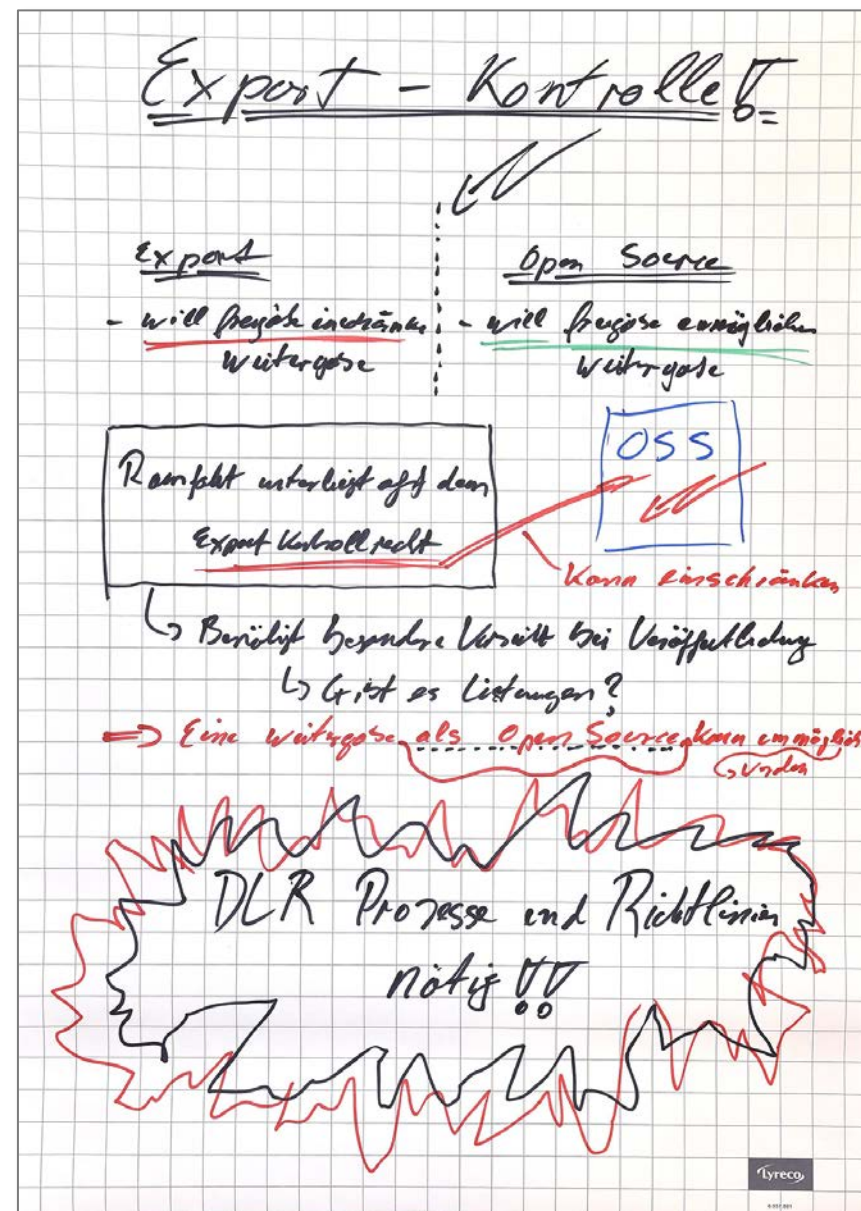


Erkenntnisse aus den WAWs

Ein wesentliches Resultat aller Workshops war, dass ein klar definierter Prozess zum Umgang mit Open Source fehlt

- Entwicklung eines Prozesses nun in Arbeit
- Bestandteil der Qualitätsmanagementsysteme in den Instituten
- Abgesegnet von "ganz oben"

Allgemein viel Feedback für weitere Aktivitäten



Beratung und Unterstützung



Wissen für Morgen



Hilfe bei Lizenzfragen im DLR

Hilfe und Unterstützung je nach Fragestellung

- Allgemeine Lizenzfragen, Schutzrechte
→ DLR-Technologiemarketing
- Rechtliche Unterstützung für Copyright- und sonstige Rechtsfragen
→ DLR-Rechtsabteilung
- Lizenzkompatibilität, Lizenzauswahl, Entwicklung
→ DLR-Simulations- und Softwaretechnik

Email als zentrale Anlaufstelle

- opensource@dlr.de



Empfohlene Lizenzen im DLR

Default-Lizenzen, geprüft durch die DLR-Rechtsabteilung

- Simplified BSD License
- Apache License 2.0
- Eclipse Public License 1.0



Support im DLR

Typische Themen für individuelle Beratung

- Kriterien für Auswahl von Open-Source-Software
- Best-Practices für eigene Open-Source-Projekte
- Migration nach Open Source



Q&A im Wiki

DLR Wiki Bereiche - Frage ans DLR Erstellen

Seiten / SVN / Software Engineering

Ask a Question

Angelegt von Schlauch, Tobias, zuletzt geändert am 10. Februar 2017

Do you have a programming problem? You heard about a cool tool but you do not know whether it useful in your situation? We might be able to help you!
On this page you can easily ask your programming or software development related question!

- If you want to ask a Question...
- If you want to answer a Question...
- If you - as the questioner - want to mark a Question as answered...

Ask a Question

Question	Status	Answers
I want to upstream a (small) patch, what form of signoff do I need from whom?	ANSWERED	3
Was sollte an Mindestvorgaben für das Software-Testen etabliert werden?	ANSWERED	5
Perl Distribution für Windows im DLR	ANSWERED	5
Abfrage der SVN-Revisionsnummer eines Python-Skripts	ANSWERED	5
Einführung von Versionsverwaltung	OFFEN	3
Simpack User Routines	ANSWERED	4
Zugriff auf Python Modul	ANSWERED	6
Jenkins-Mantis	OFFEN	1
Zugangsdaten Verwaltung in Jenkins	ANSWERED	3
3D Animation mit Python	ANSWERED	5
Java IRC Chat	ANSWERED	8
Schulung Jasper Reports und iReport Designer	OFFEN	2
Tools for Software Development Project Management (Requirements, Specification, Monitoring)	ANSWERED	2
Welche Python-Distribution?	ANSWERED	10
Tools zum Testen von GUIs / HMIs	OFFEN	3
self Objekt in Python ersetzen	ANSWERED	2
Wie kann der Fehler bei der Auto Completion in Eclipse Luna gelöst werden "Compilation unit is not on the build path of a Java project"??	ANSWERED	2

Gefällt mir 6 Personen gefällt das.

Schreiben Sie einen Kommentar...

Contributors: Schlauch, Tobias, Gatter, Alexander, Posovezky, Philipp, Ricking, Andreas, Calaminus, Bastian, Goormann, Lukas, Meinel, Michael, Rooney, Daniel, Bachmann, Arne, Göttfert, Tobias, Munteanu, Robert, Schulz, Susanne, Unbekannter Benutzer (fwe_ru), Amore, Mario d', Haupt, Carina Handrick, Björn, Pilewiczki, Andre, RShrig-Zöllner, Melven, Schreiber, Andreas, Unbekannter Benutzer (jaco_da), Unbekannter Benutzer (sire_ed), Zeidler, Petra, Behnecke, Danny, Brandt, Hartmut, Fuchs, Benjamin, Graser, Maximilian, Heinrich, Lars, Köhler, Claas, Henning, Künemund, Maren, Malbaum, Olaf, Seth, Daniel, Steinmetz, Franz, Urlings, Thomas

Latest Questions: Experience with Django framework, I want to upstream a (small) patch, what form of signoff do I need from whom? — I have a patch for an Open Source tool that makes the tool run better on Solaris 11. I want to contribute that patch back to the project that makes the tool. Who do I need to ask for sign-off to be allowed to contribute the patch? Should I submit it as DLR employee or privately?, Perl Distribution für Windows im DLR — I need a Perl distribution for windows, is there a recommendation regarding Perl at DLR?, Einführung von Versionsverwaltung, Was sollte an Mindestvorgaben für das Software-Testen etabliert werden?

Powered by Atlassian Confluence 5.10.3 · Fehler/Anregungen · Atlassian-News

Atlassian



Key Message

Unser Vorgehen im DLR basiert auf

1. Zuerst zielgerichtete Informationsangebote für Mitarbeiter
2. Danach Raum und Gelegenheit zur „Peer-to-Peer“-Diskussion und Wissensaustausch
3. Erst am Ende formale Prozesse und Vorgaben „von oben“

Feedback der Kollegen

- Positive Bewertungen für die bisherigen Maßnahmen
- Tipps, Hinweise und konkrete *Action Items* für weitere Maßnahmen

Unser Vorgehen wird von anderen übernommen (Helmholtz-Gemeinschaft etc)



Vielen Dank!

Fragen?

Andreas.Schreiber@dlr.de
www.DLR.de/sc | [@onyame](https://twitter.com/onyame)