
OPEN STANDARDS & OPEN SOURCE
ACCELERATING INNOVATION BY OPENNESS
Examples from Industry and Academia
Modelon

OVERVIEW

- Open source platforms in engineering
- The Importance of Open Standards
  - Modelica, FMI, XML, ...
- FMI, the Functional Mock-up interface
- Examples
  - Trajectory optimization
  - Open source CFD
- Open Source business & development models
  - What is needed to make it work?
  - Industry view: Why use open source?
  - Academic view: Why engage with open source?
  - Tool vendor view: Why open source your code?

OPEN SOURCE PLATFORMS IN ENGINEERING

- Programming languages & Tools
  - Python, R, Julia, Modelica, Eclipse, ...
- Matlab replacements
  - Python + SciPy/NumPy, Octave, Scicos
- CFD codes
  - OpenFOAM
- FEM codes
  - Fenics project, OpenFOAM
- Modeling and Simulation Tools
  - SciCos/Scilab, JModelica.org, Openmodelica
- Computer Algebra Tools
  - CasADi, Sympy, Sage, Maxima, ...
- Toolkits for scientific visualization
  - VTK, ITK, ParaView
- Many other classes of tools

  ...

MOTIVATION

Capturing the Real Value of Innovation Tools
(MIT Sloan Management Review)

What is important to succeed with innovation and development?

- People,
- Processes
- Tools

Are jointly responsible!

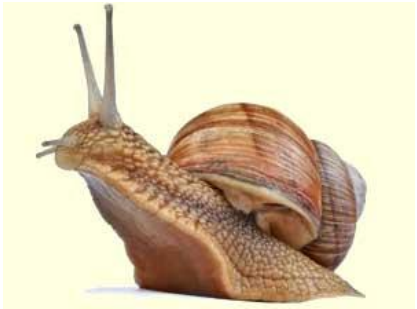
Open source is more flexible than commercial tools, thus often easier to adapt to specialized processes.

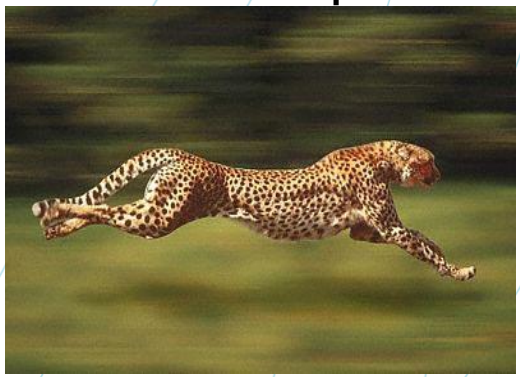
OPEN SOURCE PLATFORMS

- Observations:
  - Many tools use Python as integration platform. Even commercial tools use Python interfaces for scripting & automation
  - The speed of innovation & tool creation seems higher in some open-source communities than in commercial tools (Python-related & Eclipse-based tools)
  - Mixed open-source/commercial solutions very common in Science and Engineering

HOW FAST ARE STANDARDS ADOPTED?

- Some standards become obsolete before they are adopted!
  - STEP: started in 1984, adoption still weak
- Some standards are too complex to ever be fully adopted
  - SGML, now only one profile used, XML
- How to drive fast adoption of a standard?
  - Make implementations available in open source!





MODELICA

- Standard by non-profit Modelica Association
- Developed since 1996, active development
- Strongest contender to become a truly vendor-independent modeling format, supported by many vendors
  - At the start, there was only one viable tool, now there are many, and the number is growing
- Development process in Modelica Association is maturing

THE FUNCTIONAL MOCK UP INTERFACE

- Version 1 released Jan 26 2010 / Oct 12 2010
- Well-visited track at Modelica conference 2011
- *30 Tools* with support listed May 14th 2012
- Many in-house uses at companies

- Why so fast?
  - BSD-licensed SDK released immediately with standard → low initial threshold
- Open source (with right license) accelerates

FMI OUTLOOK

- Has been an idea waiting for someone to come and propose
- Likely to be adopted broadly in automotive and aerospace industries, possibly more
- Has a chance to greatly simplify model exchange and tool interoperability in modeling and simulation
- Has a chance to break up Simulink's position as single platform for integrating simulation tools

OPEN STANDARDS

- Lower threshold of entry for new commercial players
- Potential for large cost savings for both vendors and users
- Can open up commercial space even in near monopoly situations.

WHAT ARE STANDARDS GOOD

- Standards drive down prices!
- They speed up & simplify the tedious part of development (in software)
- Dominating monopolies usually don’t occur in domains with a good standards culture.

- No distinction made between formal standards and de-facto standards!

EXAMPLES

GOING TO THE MOON

- **Objective** Minimize the fuel consumption required for an insertion into an elliptical moon orbit from a halo orbit around libration point *L1.*
- **Dynamics** The dynamics are described by the planar, circular, restricted three-body problem (PCR3B

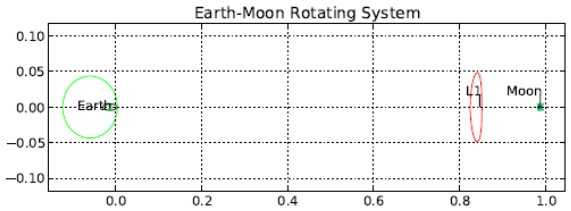
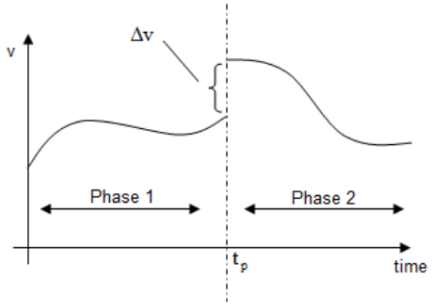
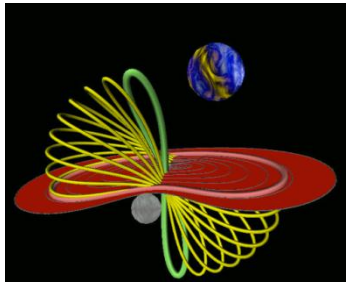


Figure: Earth-Moon system together with a Halo orbit around the libration point L_1.



- Divide the problem into phases.
- Allow the velocities to be changed over the phases.
- Allow the time *tp to be free.*
- Introduce a *linkage* constraint

Visualization of Halo orbits

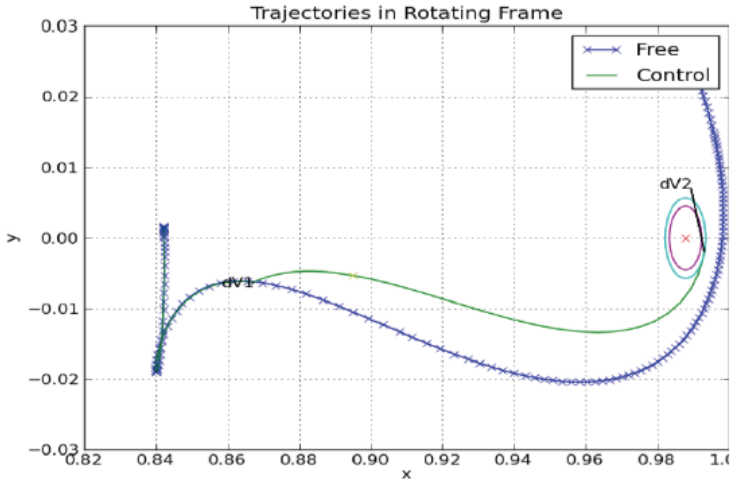


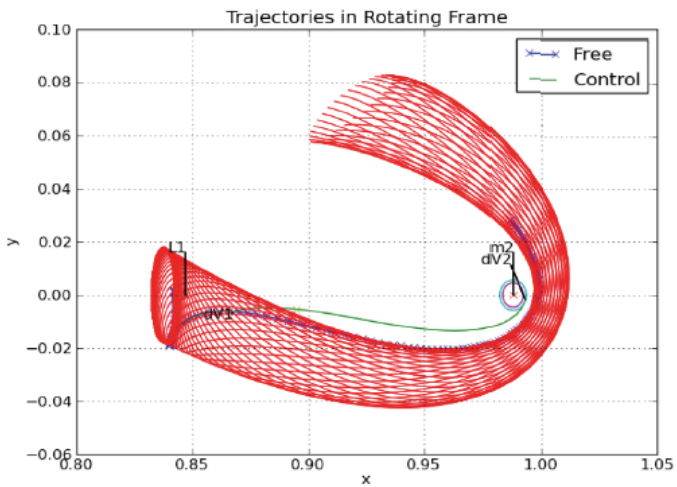
MITSUBISHI ELECTRIC RESEARCH LABORATORIES
Cambridge, Massachusetts

- **Missing Pieces**: availability of Gauss Pseudospectral Method and a formulation that can handle phased optimization methods

SOLUTION

Results for a fixed initial point with a small perturbation on the unstable manifold. Solution obtained using the Gauss Pseudospectral Method in JModelica.org.





The blue line represents the trajectory where no control is applied. The green line is the optimized trajectory with dV1 and dV2 as control.

- From idea via algorithm implementation to solution (used in at least one patent) in less than 3 months.

OS SOFTWARE INVOLVED

1. JModelica.org
   - Formulate dynamical system in Modelica
   - Formulate optimization problem in Optimica
2. Python
   - Scripting, integration, plotting
3. CasADi: AD-techniques for algorithm development
   - Efficient symbolic computation of derivatives
4. IPOPT
   - Efficient numerical solution of resulting NLP

⇒ **Prime example of technology integration**

TALENT INVOLVED

1. Controls engineer with good understanding of problem
2. Numerical analysis student with good understanding of tool chain

→ Innovation needs the right mix of people to get the job done!

AERODYNAMIC PROCESS AT AUDI

- Process based on open source CFD code OpenFOAM.
  - Wind tunnel booked at 105% capacity!
  - Cost-effective replacement of some wind tunnel experiments needed
- Challenges of Open-Source Software in an Industrial Environment!

From Presentation on OpenCFD 2009, by Dr. Moni Islam, Wind Tunnel Centre of AUDI

WHY USE OPEN SOURCE?

- **Quasi-monopolistic environment**
  - Very small number of commercial codes truly viable for productive use
  - Proprietary technology offering limited insight or black-box approach
  - License fees increase with increasing use
  - Code development driven primarily by vendor's interest
  - Very high overhead associated with switching to alternative product

- New approach was needed!

WHY OPEN SOURCE CFD?

- Solution to many current problems provided by open-source model for CFD code
  - High process integration
  - Robustness, ease of use and application speed achieved by application-specific customization
- High accuracy
  - **Full transparency of technology** (v. black-box approach) permits complete analysis and solution of problems
  - New / alternative technology can be implemented rapidly on demand
- Initial cost not an issue: cost scales better with increased use!
  - Fixed with increased use
  - Limited and predictable: User pays for only what she needs
- Excellent long-term potential for technological development and process integration due to high customizability
- User has free choice of technology provider

SOLUTION ELEMENTS

- Multi-year project to introduce open source based solution into development at AUDI, Volkswagen & Seat
- Training & Core development by OpenCFD Ltd.§
- Custom application development by ICON Ltd
  - Customization to AUDI specification & process
- Custom applications NOT open source!
  - Some important features only in customized version
- Process & Workflow design important
- 3-10 jobs/day on massively parallel cluster

§ About 500.000 Lines of Code added to OpenFOAM core by developers in a period slightly longer than the introduction at AUDI

CHALLENGES

- “CFD for Dummies” not a viable working model
- High flexibility offered by OpenFOAM toolbox requires in-depth knowledge of end user:
- Open source has cost that must be borne by the user

- Understanding of intellectual property required
  - Reluctance toward open-source software due to view as risk to know-how and investment
  - Need for clear boundaries between IP of public domain, technology provider and end user
  - Clear conceptual understanding by management and technical staff essential

CASADI

- Symbolic framework for implementing derivative-based algorithms for dynamic optimization
- Open-source (LGPL) tool, use from C++, Python or Octave
- Developed at the Optimization in Engineering Center (OPTEC) at the Katholieke Universiteit Leuven (Belgium)

CASADI

- Automates tedious and involved tasks such as derivative calculations
- Allows users to work in a high-productive environment such as Python instead of a low-level language such as C without compromising numerical efficiency
- Example: Collocation algorithm & Gauss pseudo-spectral algorithms in JModelica.org

→ improves flexibility and development efficiency for new algorithms in research tremendously

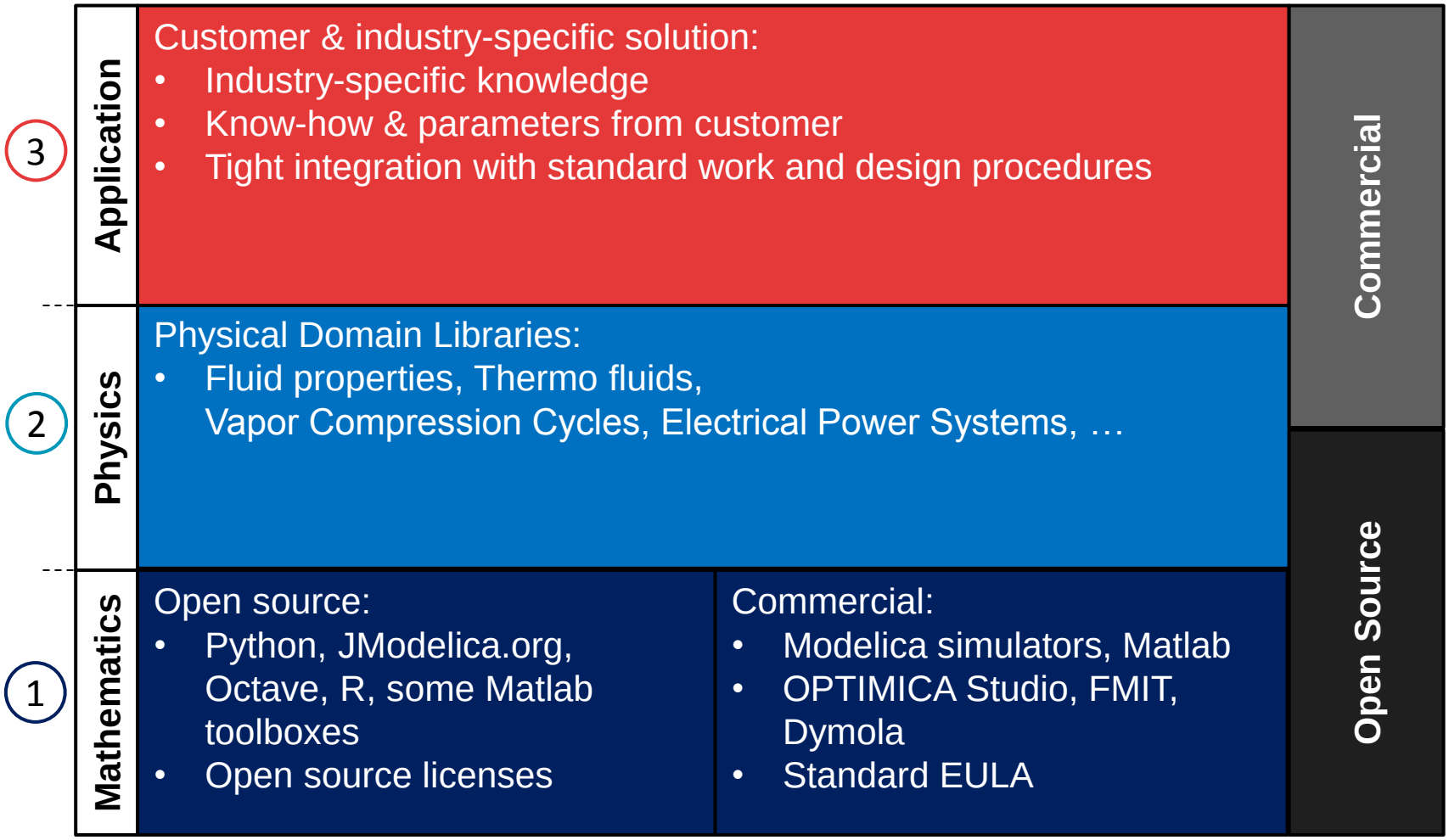
OPEN SOURCE BUSINESS MODELS

- “Commoditize your complement”
  - Make Android free to make money on mobile advertisements
- Open source + X
  - X = consulting (IBM)
  - X = support & subscriptions (Redhat)
  - X = commercial complements (Oracle)
  - X = training & documentation (OpenCFD Inc.)

INDUSTRY: WHY OPEN SOURCE

- ✓ Problem: how can we keep a competitive advantage if our tools are based on open source?
- ✓ Often a need to have a proprietary layer on top of the open source stack:
  - Proprietary application on top of OS software
  - Proprietary process
  - Device using the software

IP LAYERS IN MODELICA SOLUTIONS



OPEN SOURCE PROCESSES

- Development completely open
  - Public tracking of activity
  - Public code review
  - Public testing
- Open process creates trust!
- Open development and testing process a competitive advantage of open source

ACADEMIA: WHY OPEN SOURCE ?

- Scientific American (12/4 2012)§: Secret Source Code is Bad for Science

  "Now, a group of scientists is arguing for new standards that require newly published studies to make their source codes available. Otherwise, they say, the scientific method of peer review and reproducing experiments to verify results is basically broken."

- The “Science” Journal requires source code publication!
- Peer review is fundamentally broken without access to source code!

§ http://www.scientificamerican.com/article.cfm?id=secret-computer-code-threatens-science

ACADEMIA: WHY WORK WITH OPEN SOURCE 2

- Our experience:
  - Academic open source projects that are used in commercially supported open source improve their process discipline:
    - Testing
    - Reliability
    - Dependable schedule

➢ Increased quality of software for academic users!

VENDOR: WHY OPEN SOURCE

- Realize that real business is complement to a current product
  - Open source the product to “commoditize your complement”
- Use open source to seed the market for “open source + X”
- Increase development workforce through community
  - Important and nontrivial to set up and maintain active community!
  - Requires time, and people with the right skills to build up and nurture the community
- Lower entry threshold for technologies into new markets/industries
- Create academic/industrial collaboration to jump ahead/stay ahead of the competition

SUMMARY

- Open Source and open standards can both increase the speed innovation
- Cost is not the main driving reason for the adoption of open source, it is other positive effects of openness!
- Open source business models are viable for companies of all sizes

Thank you!