



Open Source Tools for Embedded Systems

# **PolarSys**

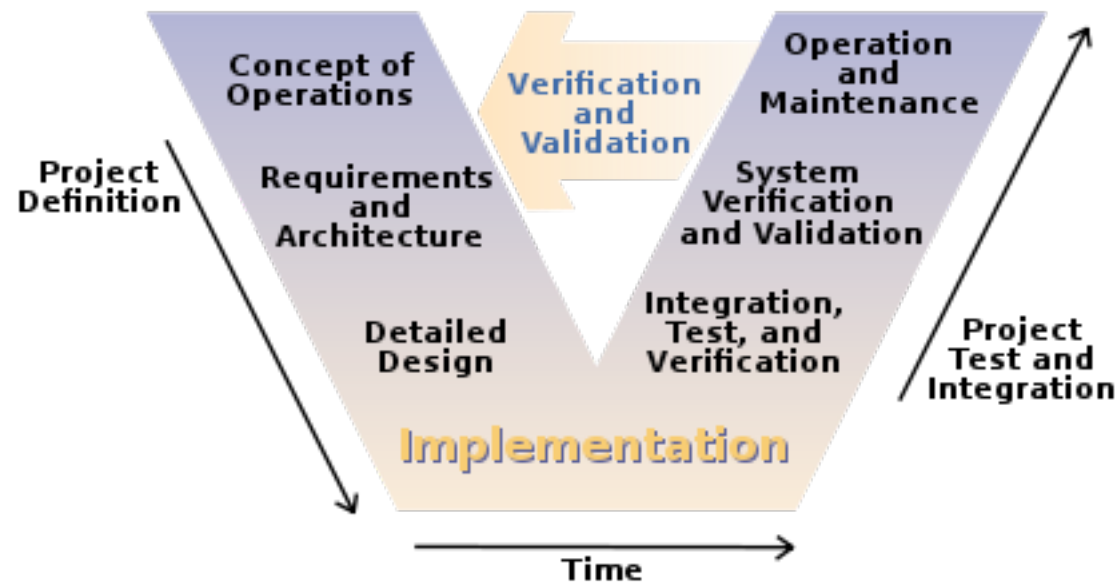
## **An example of open source community with governance**

September 28<sup>th</sup>, 2014



# PolarSys goals

- Mature Open Source tools for Embedded Systems
- Long Term Support (up to 30 years)
- Support System and Software Engineering activities





[GETTING STARTED](#)[MEMBERS](#)[PROJECTS](#)[MORE ▾](#)[HOME / ABOUT US](#)

## About Us

- » [Annual Report](#)
- » [Foundation](#)
- » [Governance](#)
- » [Legal Resources](#)
- » [Contact Us](#)

# About the Eclipse Foundation

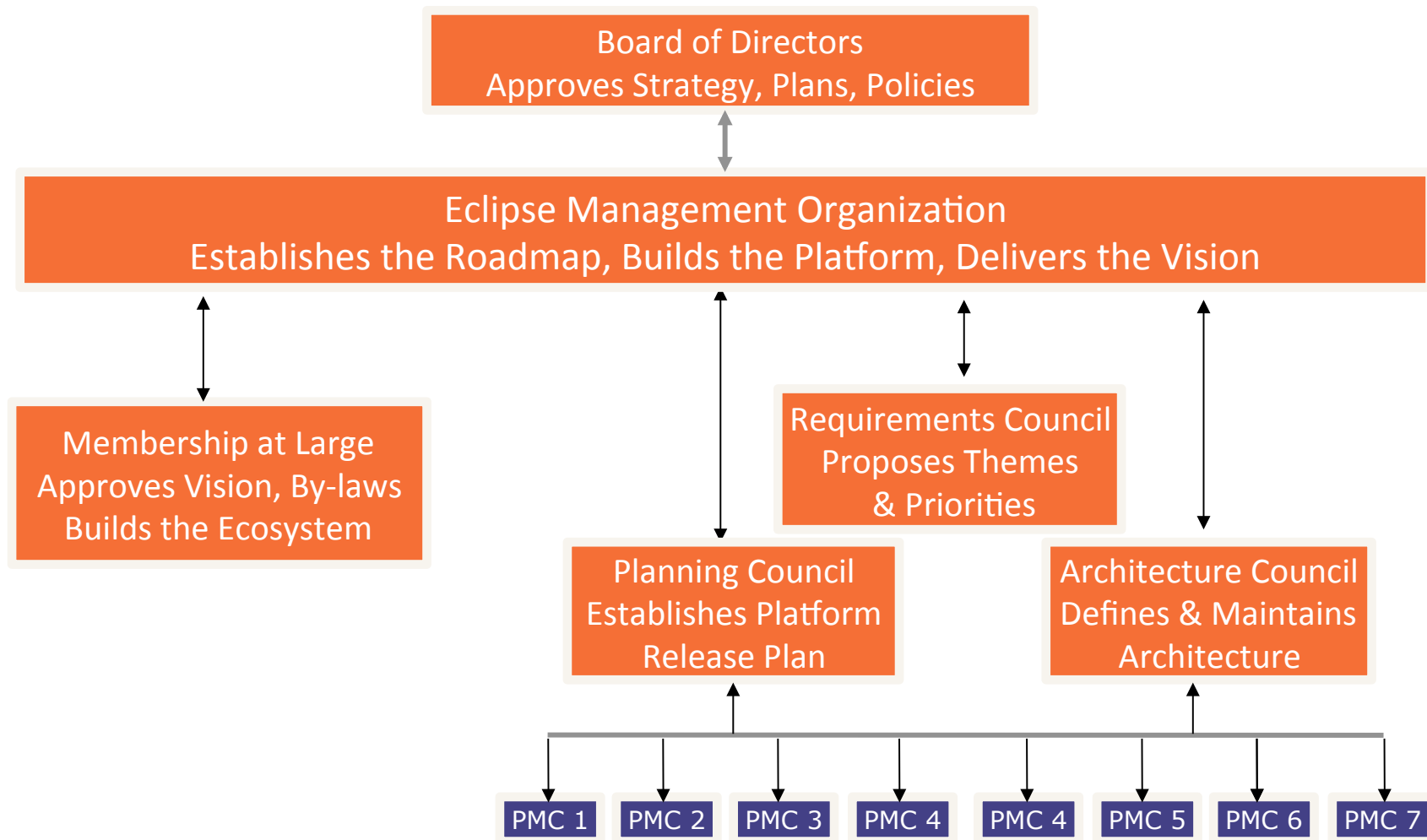
- **What is Eclipse and the Eclipse Foundation?**
- **Services of the Foundation**
  1. **IT Infrastructure**
  2. **Intellectual Property (IP) Management**
  3. **Development Community Support**
  4. **Ecosystem Development**
- **A Unique Model for Open Source Development**
- **What is the history of Eclipse?**

## What is Eclipse and the Eclipse Foundation?

Eclipse is a community for individuals and organizations who wish to collaborate on commercially-friendly open source software. Its projects are focused on building an open development platform comprised of extensible frameworks, tools and runtimes for building, deploying and managing software across the lifecycle. The Eclipse Foundation is a not-for-profit, member supported corporation that hosts the **Eclipse projects** and helps cultivate both an open source community and an ecosystem of complementary products and services.

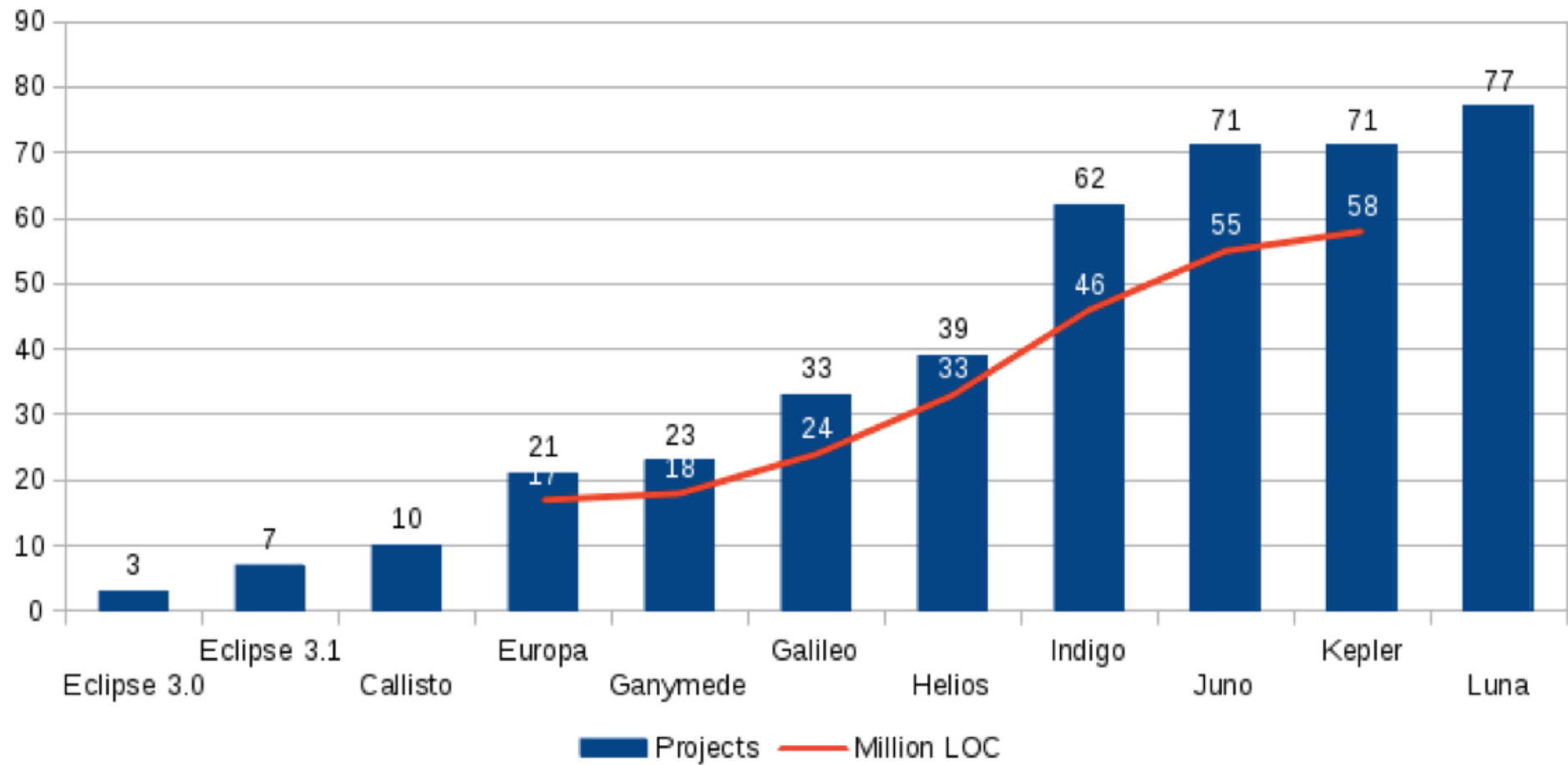


# Governance and meritocracy





# Predictability





# Diversity





# Eclipse

Millions of users

Thousands of products

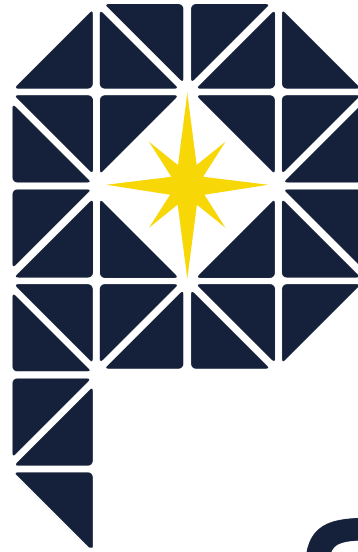
Thousand+ committers

Hundreds of companies and projects

Twenty employees

Zero product managers





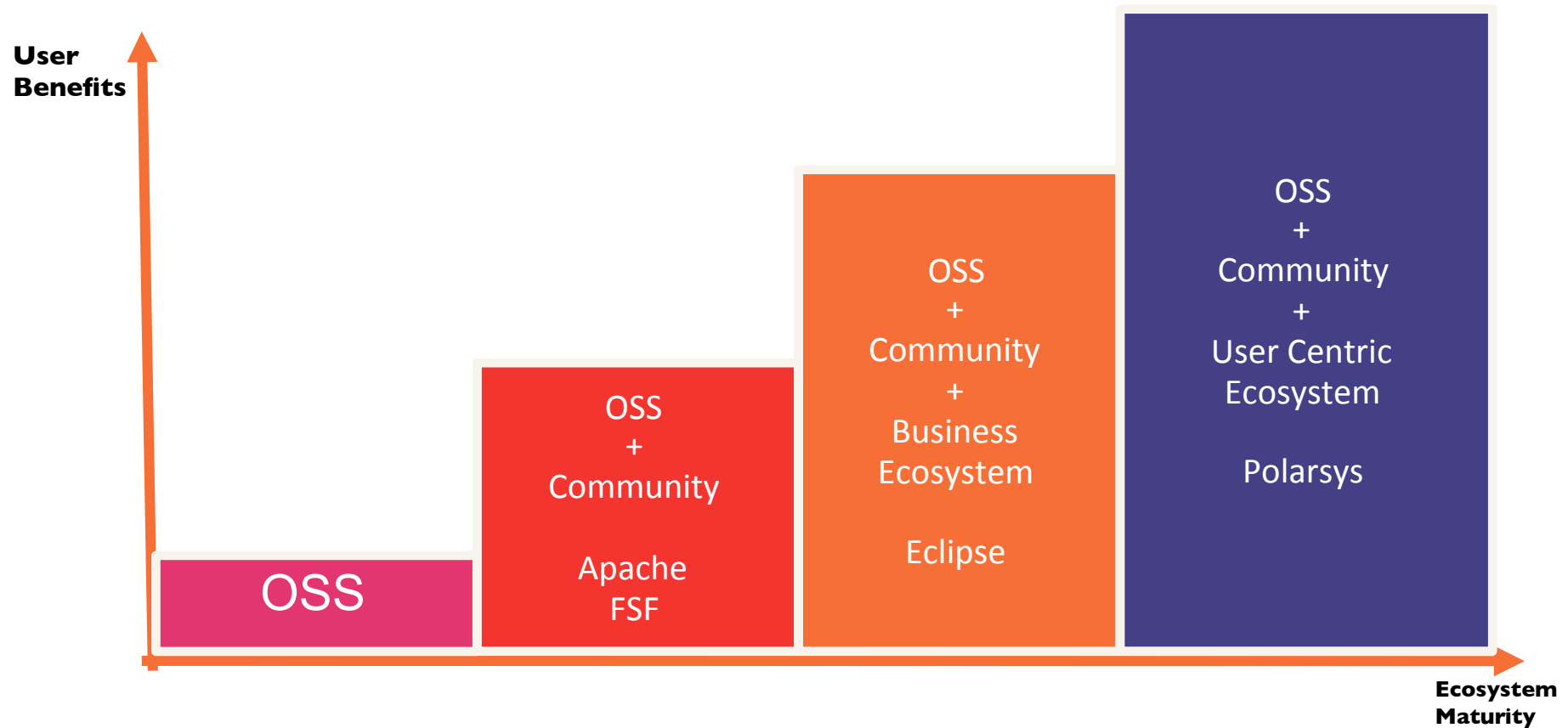
# POLARSys

**Extending the Eclipse  
Ecosystem**



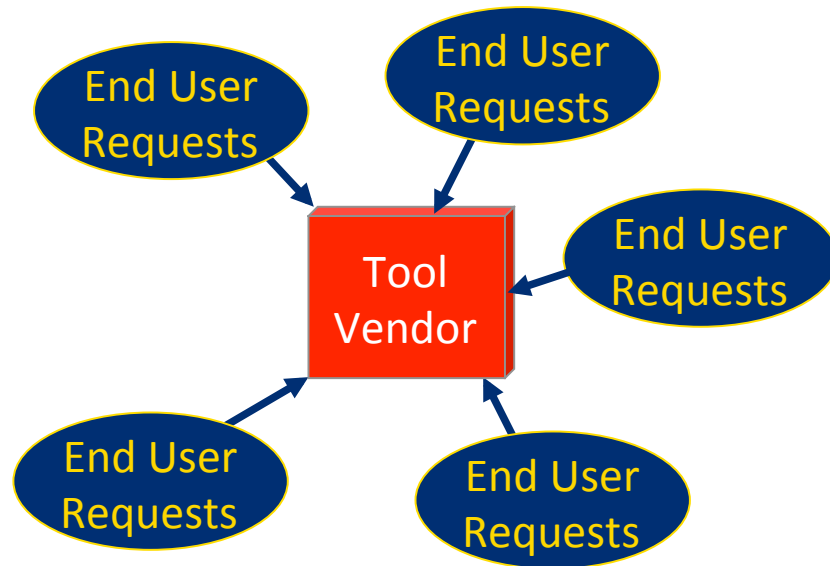


# A new kind of OSS ecosystems



# An ambitious objective

## Change the tools ecosystem



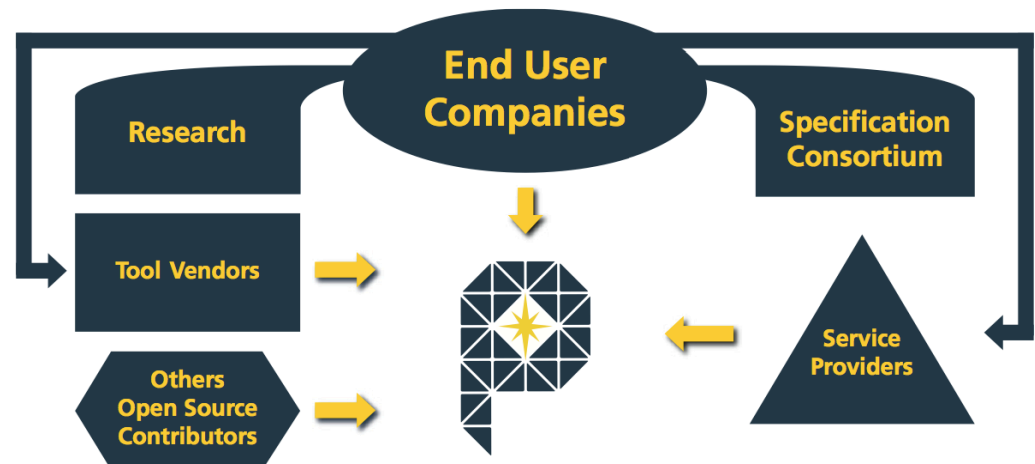
Typically less than **20%** of Requests For Enhancement (RFE) get into the tool



Proprietary Tool Vendor Ecosystem

### No lock-in!

You or many third party can add features



### PolarSys: a user driven ecosystem

**100 %** of RFE are able to get into the tool  
e.g. **80%** generic, **20%** as user extensions



PolarSys Open Source Ecosystem

# PolarSys Members (2014/10)



## Steering Committee Members



## Participating members



## Academics / Universities



... Other universities coming



# PolarSys additional services

Maturity assessment

Long Term Support

Project Planning Committee

Custom builds

Solutions marketing

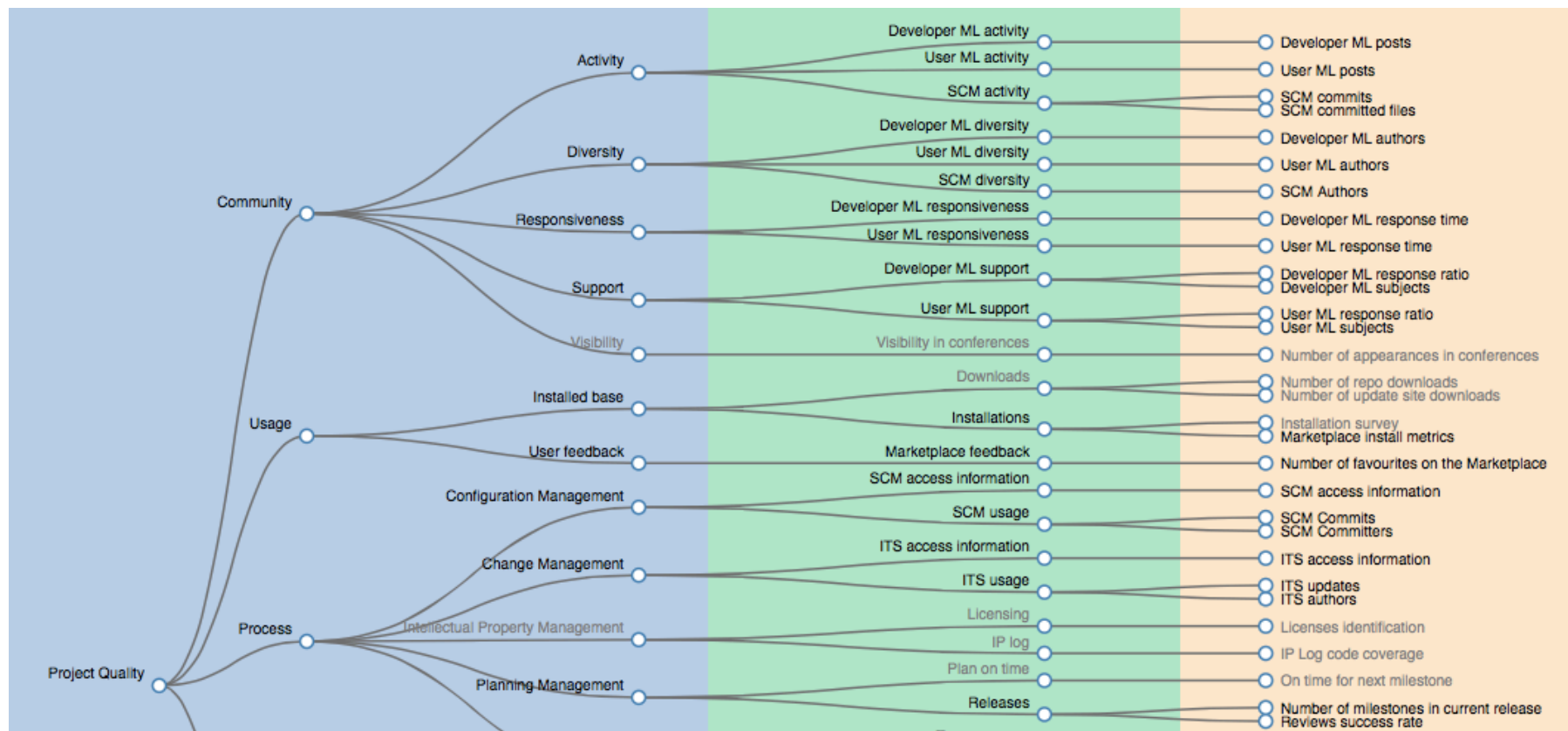
- Data sheets
- White papers





# PolarSys additional services

## Maturity assessment





# PolarSys additional services

Maturity assessment

Long Term Support

Project Planning Committee

Custom builds

Solutions marketing

- Data sheets
- White papers





# PolarSys additional services

Maturity assessment

Long Term Support

Project Planning Committee

Custom builds

Solutions marketing

- Data sheets
- White papers



Titan is a test automation toolset supporting the complete chain of test development, test execution and result analysis. It is based on the standard test language TTCN-3 and thus is extremely well suited for grey-box and black-box testing such as component, function, integration, system, acceptance, conformance and model based testing. In addition, Titan is being used as an engine to create additional tools for performance testing and security testing.

Titan provides a full-featured TTCN-3 open source environment with compiler, runtime, controller, Eclipse-based and command line interfaces, and System Under Test (SUT) adapters for several protocols. TTCN-3 has a proven track record for testing complex systems with a large code base and is applicable in multiple domains.

## Eclipse-based and command line environments

Titan provides a complete Eclipse IDE and a command line; both have been architected to work in mixed mode.

Titan's Eclipse interface allows the development of test cases, the launch and monitoring of test execution and analyzing test results. The editor provides sophisticated editing, search, code completion, analysis and quality assurance features. Test execution events and status are shown in real-time and are logged for post-execution analysis. Test logs can be viewed both graphically and in a textual/tabular format. When selecting a log event, the source code line producing the event is automatically shown.

Command line components allow the building of executable test suites and launch test runs on demand, or automatically on an event-based or scheduled (nightly) manner for continuous integration. They also allow test projects, developed in the Eclipse environment, to be built and executed from the command line.

## Industrial Toolset

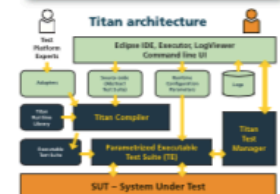
The Titan toolset enables teamwork in distributed teams worldwide and can be used in workflows where tests are either manually developed, generated from Models, or developed for continuous integration. Titan provides a scalable toolset:

- Supports international specifications ASN.1, XSD, IDL and JSON
- Fast compilation, fast incremental re-compilation
- High performance runtime
- Detailed, configurable logging
- Logger plugin API and built-in loggers (Unit, textual)
- APIs for external C/C++ functions and specific codecs
- API to interworking with other languages like Java, Python etc.
- Built-in codec generators for XML, ASN.1, bit-oriented and textual protocols
- Distributed, multi-platform test execution on Linux, Solaris and Windows platforms



## Titan Benefits

- All-in-one solution for your testing needs
- More than ten years of development, thousands of active users
- Secure investment: based on a standard test language
- Quick development of new protocol support and quick adaptation to protocol changes
- Well suited to both traditional V process and agile development processes
- Allows testing at an early phase of development, decreased cost, improved product quality
- Faster development of tests with an easy-to-learn intuitive test tool and language
- Highly efficient in V-model, Lean, Agile and key to enable Continuous Integration
- Increased testing productivity and test coverage providing quicker time to market
- Multi-purpose: functional and non-functional testing (performance, security)
- High degree of test re-usability
- A set of SUT adapters and protocol support are available in open source





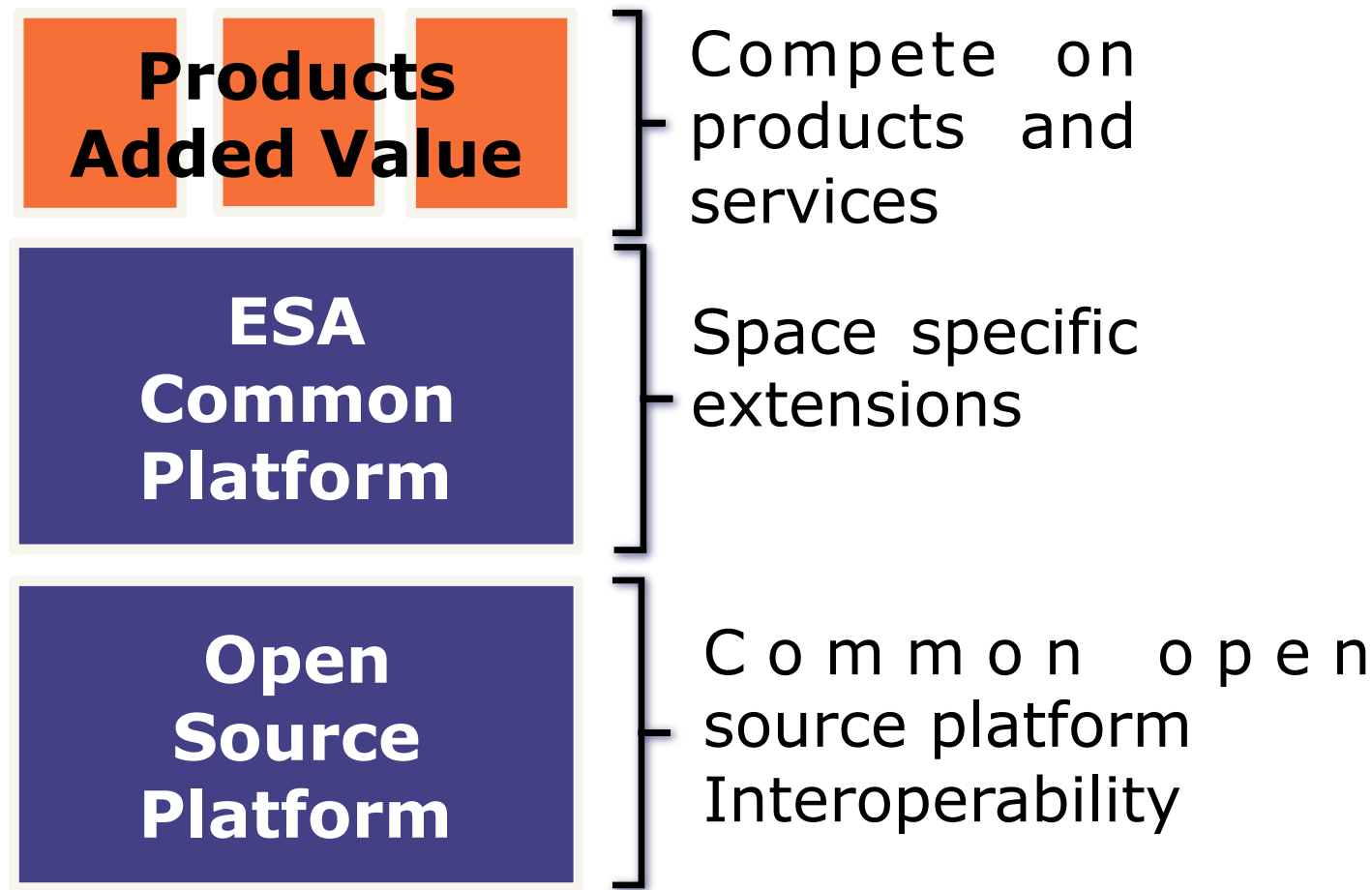
# An ecosystem relying on an extensible platform



- Identify precisely what your competitive differentiators are for your customers
- Focus all possible energies there, and acquire everything else from OSS, or help build it in OSS

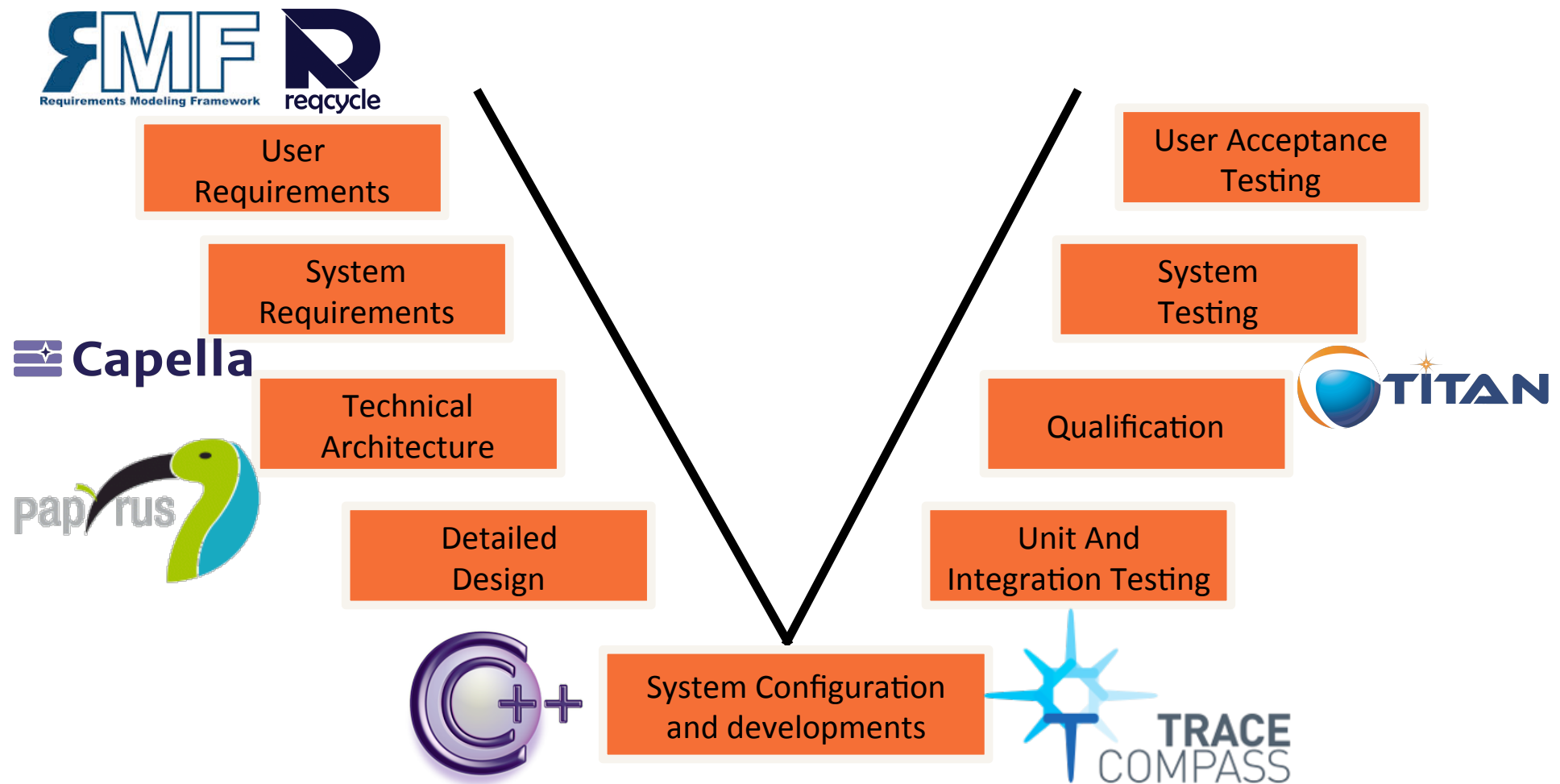


# A space specific implementation?





# The tools ecosystem is growing





# Underlying components

Solutions leverage a rich ecosystem of Open Source Projects

- Graphical frameworks (Sirius, GMF, ...)
- Code generation/transformation (Acceleo, ATL, ...)
- Doc generation (Gendoc)
- SVN / Git Support (eGit, eGerrit)
- Model validation (OCL)
- Collaborative repository (CDO)
- Compare and merge models (EMF Compare)





# New and growing topics

- Functional Simulation
- Requirements engineering and traceability
- Safety Analysis
- Middleware/Runtime

*Ideal place for  
collaboration*





# The future is Open!







# Photos credits

- <http://www.freeimages.com/photo/1241753>  
by Jason Weeks
- <http://www.freeimages.com/photo/1440526>  
by Eduardo Siqueira Filho
- <http://www.freeimages.com/photo/1356931>  
by Guglielmo Losio