Titan

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 (JUnit, 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
- As an open source product licensed under the Eclipse Public License (EPL), it avoids vendor lock-in
- 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
- 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 adaptors and protocol support are available in open source

Titan architecture
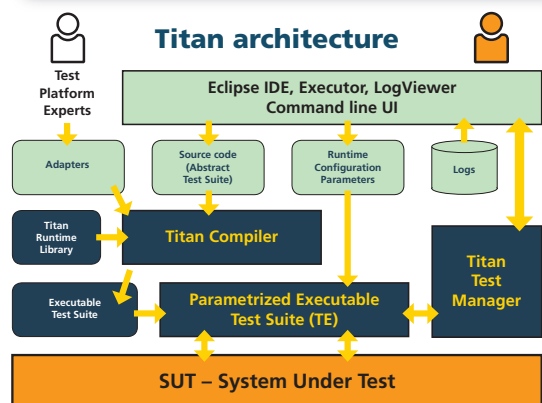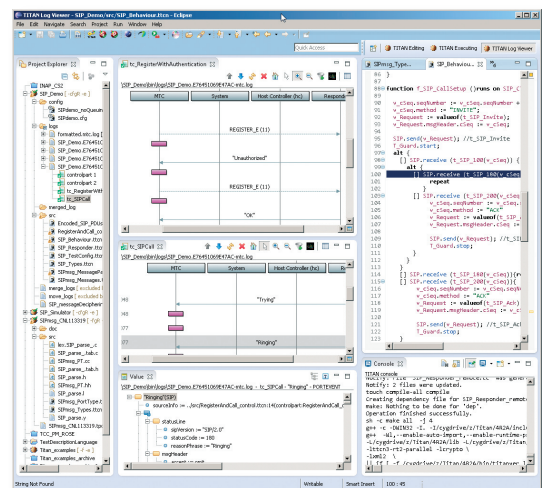
TTCN-3 a universal standard test language

One of Titan's key features is its implementation of the Testing and Test Control Notation version 3 (TTCN 3) language. TTCN-3 is an international standard, specified and published by ETSI and endorsed by ITU-T. It has been applied to a variety of domains such as Automotive, Avionics, Finance, Healthcare, IT, Energy, Space and Telecom. It is also used in research projects and by many universities. TTCN-3 is used by standardization bodies and consortia such as 3GPP, AUTOSAR consortium, ETSI, MOST cooperation, OMA and the Wimax Forum to produce conformance and interoperability test standards.

TTCN-3 benefits

- Precise syntax and semantics which maximize compile time checking capabilities
- High level language and abstraction of the platform and test environment
- Rich type system, allowing the support of a wide range of protocols and API definitions
- Test data can be based directly on XSD, ASN.1, IDL and JSON (under development) specifications
- Programming language features for efficient behavior design
- Rich language features to meet testing needs
- High precision for controlling test verdict assignment and logging
- Maximum reusability due to separation of the test logic from the details of the environment: test cases are reusable in different software development phases, platforms and test environments
- Well suited for all forms of grey-box and black-box types of testing

About PolarSys

PolarSys is an industrial working group dedicated to open source tools for embedded systems development. It addresses the full range of System Engineering activities including Modeling, Requirement Engineering, Simulation, Coding and Debugging, Testing and Verification.

It was created by large organizations such as Airbus, Atomic Energy and Alternative Energies Commission (CEA), Ericsson, and Thales. It is an ecosystem of tool users, vendors, service providers and researchers, which has several benefits
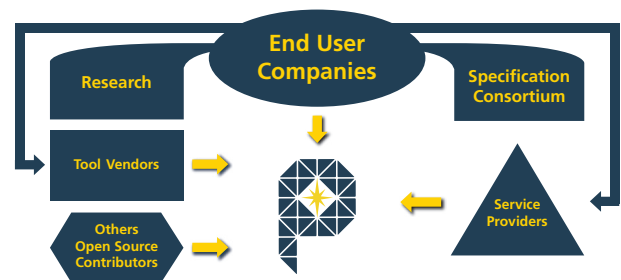
PolarSys: a user driven ecosystem

- Open Innovation and advanced features
- No Lock-in: you or third parties can add features
- Open source with commercial support
- No license fees
- Industrial user community driven
- World class intellectual property management for open source

- Very long term support
- Systematic maturity assessment
- Interoperability
- Accelerates product development
- Technology platform
- Designed for extensibility and adaptation to your context

Resources

PolarSys: **http://www.polarsys.org**
TTCN-3: **http://www.ttcn-3.org**
Titan Eclipse project: **http://projects.eclipse.org/projects/tools.titan**

Eclipse Foundation