What is CANoe4SW?
CANoe4SW is the comprehensive tool for development, test, and analysis of software in cyber physical systems — individual software components as well as subsystems and entire distributed systems. CANoe4SW supports software developers and testers in all markets (medical, railway, automotive, ...) throughout the entire development process of distributed systems and IoT devices. Virtual execution environments on the computer, in virtual machines, or in the cloud enable software to be tested for any targeted OS based on Windows or Linux, e.g., Ubuntu, CentOS, SUSE. The System Under Test is accessed by its functional system interfaces at a pure software level. This enables early “black-box” SIL testing independently of hardware availability. CANoe4SW integrates seamlessly in CI/CT environments. In addition, the support of connectivity protocols such as MQTT provides access to IoT devices and back end software running in the cloud.

Overview of Advantages
> Interactively develop and test distributed systems
> Easily automate tests to ensure quality at a system-level
> Divide and conquer the system by isolating single components using models for both physical and software environments
> Test early in the development process by utilizing virtual execution environments
> Cover dynamic aspects of the software under test by stimulating and monitoring value curves over time
> Simulate failure scenarios not easy to be covered in real environments, e.g. downtime of cloud services
> Utilize CI/CT environments right from the beginning by being independent of hardware and other software components
**Application Areas**

> **Monitoring over Time**
Monitoring over time fully integrated software components: Run your software under test asynchronously in virtual execution environments. Observe dynamic aspects while stimulating the application via its functional system interfaces at a software level.

> **Environment Simulation**
Environment simulation by models: Build a “natural ecosystem” for your software under test. Simulate both physical and software environments. Test single software components in isolation before integrating into subsystems.

> **Interactive Development and Test**
Interactive development and test in an exploratory way: Stimulate the software under test using panels, waveform generators, and scripts. Analyze the reaction of your application and value relations via graphical and text-based analysis windows.

> **Automated Testing**
Automated testing with the test design tool vTEST-studio: Leverage the flexibility of various test design methods such as graphical diagrams and tabular sequences. Benefit from flexible parametrization concepts and universal variant support. Ensure traceability from requirements to test cases and results.

> **Debugging**
Debugging on the host: Profit from the power to debug in the development environment rather than on the target. Observe application behavior, view and modify internal variables, watch call stacks, etc. while stimulating the application via interactive or automated tests.

*More information: [www.vector.com/CANoe4SW](http://www.vector.com/CANoe4SW)*

---

**CANoe4SW User Interface**