# Software from the Factory

Continuous integration shortens development cycles

IAV is using a new software development method to meet its customers' needs faster and in even better quality. New program versions are now tested sooner and more frequently. Although we are mainly using 'continuous integration' in software development for engine control units, it is also suitable for other fields.

The software development process in most cases begins with an engineer wanting to implement a specific function in the vehicle, for example to adjust the speed of the coolant pump as soon as there is a change in load. The subsequent process can be compared with a loop: program code is written for the required function, integrated into the system, tested in the vehicle or on test benches, with any faults occurring being eliminated by new program code which is then re-integrated into the system. Often, several runs are necessary before everything works exactly as it should.

In the past, new software versions were gathered over a period of several days to weeks, integrated into the overall system and then either investigated on a test bench or in a test vehicle. Instead of this, IAV uses 'continuous integration'. As soon as a new version of a sub-program is ready, it is immediately integrated into the overall system. Testing can then begin without any delay. This also applies to calibrating the function, such as for the parameters governing the speed of the coolant pump in relation to the driving situation.

Mobility needs and the automobile market are rapidly changing. This is having a huge impact on software development, for example through the growing convergence of vehicles and consumer electronics. IAV covers the entire spectrum in this field – from embedded software and back-end solutions to developing apps. With its proprietary solutions and as a competent partner, IAV is actively involved in shaping digitalization.
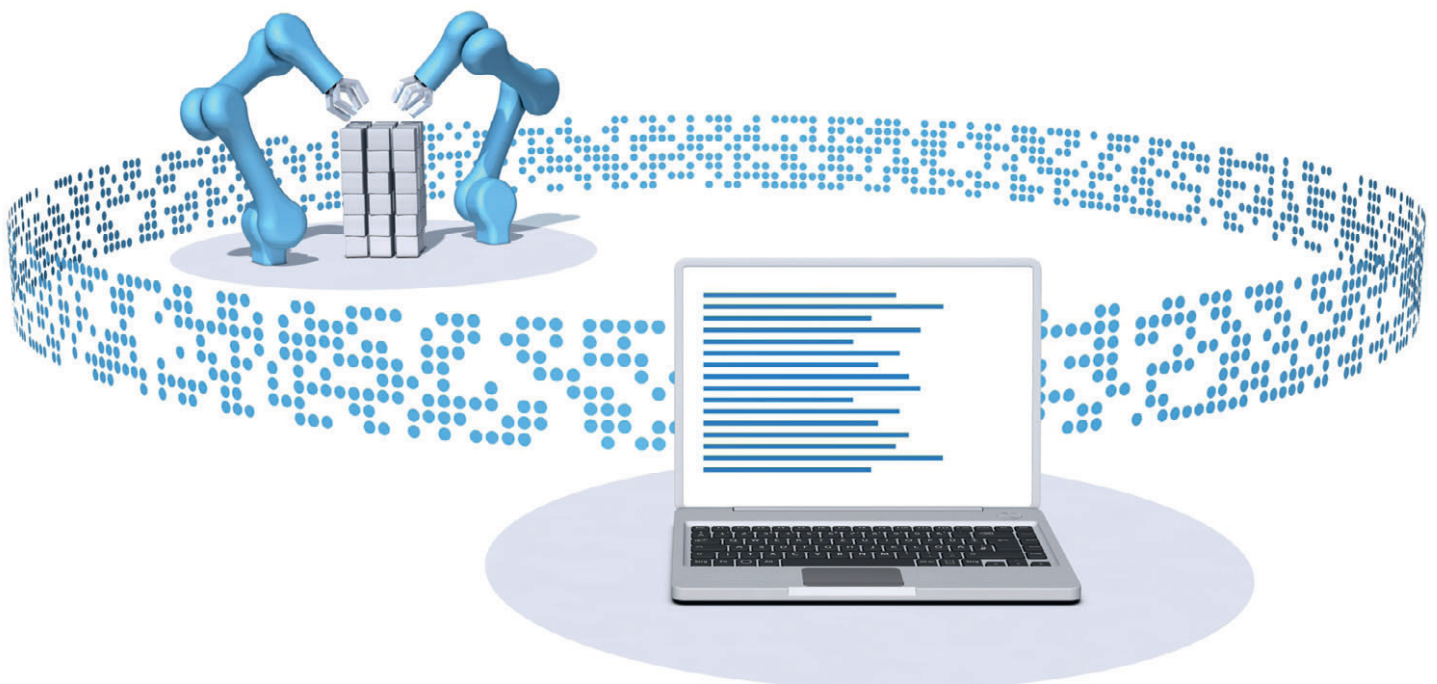
Modules on the Integration Assembly Line

Conventional procedure
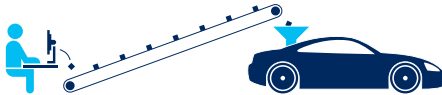
Continuous Integration

Changes are being implemented all the time

A software is responsible for 'continuous integration' that starts all of the necessary steps automatically. Our developers deliberately speak of a "robot" because the new approach has similarities with a factory. Instead of collecting changes in the program code and only integrating them collectively every few days or weeks into the system, we now implement them on a continuous basis – in a similar way to an assembly line that continuously delivers new parts to a machine.

But 'continuous integration' is just the beginning: At the moment, our experts are also working on automating the process of testing software on test benches and transferring it to the control units in test vehicles. This would then be 'continuous testing' and 'continuous deployment'.

Simplified troubleshooting and faster testing

The major benefit of the continuous working method is that malfunctions can be identified more quickly because each software version differs only slightly from the previous one. In addition, our algorithm developers can test the changes they wish to make after a short time.

In future, the results of measurement drives are to be evaluated using artificial intelligence. An algorithm for machine learning is to analyze the data and reveal any malfunctioning, for example engine overheating. It will be able to localize the cause, such as a problem in thermal management, and provide the engineer with relevant information or even autonomously and provide software versions with debugging capability. This completes the circle, and the next loop in software development can start.

Sooner and more frequently on the test bench

Thanks to 'continuous integration', new program versions are now reaching the test bench sooner and more frequently. Although this will not make faults vanish, they will be found more quickly in future. As a result, the continuous working method will let us meet our customers' demands with an even higher level of quality. This approach is suitable for all forms of embedded software and, apart from the automotive sector, can also be used for industrial control systems.