Life Cycle Support and Security Monitoring

Embedded open source systems profit from the enormous development capacity and the continual innovation of the open source community. In order to make optimal use of these advantages, however, a suitable maintenance and further development concept is needed at the level of the device software.

The devices and systems from the wide diversity of target industries usually have a life cycle which exceeds five years – in some cases by a wide margin. It is ever more commonplace that they are integrated into wide-area networks and must meet the certification requirements usual in the industry and the IT compliance requirements of the operator.

Within the framework of maintenance agreements, emlix not only makes available the capacity of developers who are familiar with the specific software system in question, the service also includes the continual monitoring of the relevant communities, mailing lists and other information sources for all the open source components contained in the software.

Security and maintenance report

Once a month, every maintenance customer receives a security and maintenance report in which an overview of the updates, bugs and bug fixes relevant to his system is presented.

They each contain a recommendation for resulting measures: a security bug might not affect the particular application context of a device, or it might make a bug fix in the next scheduled release seem appropriate. It might even necessitate the immediate roll-out of a fixed software version.

Continual security monitoring

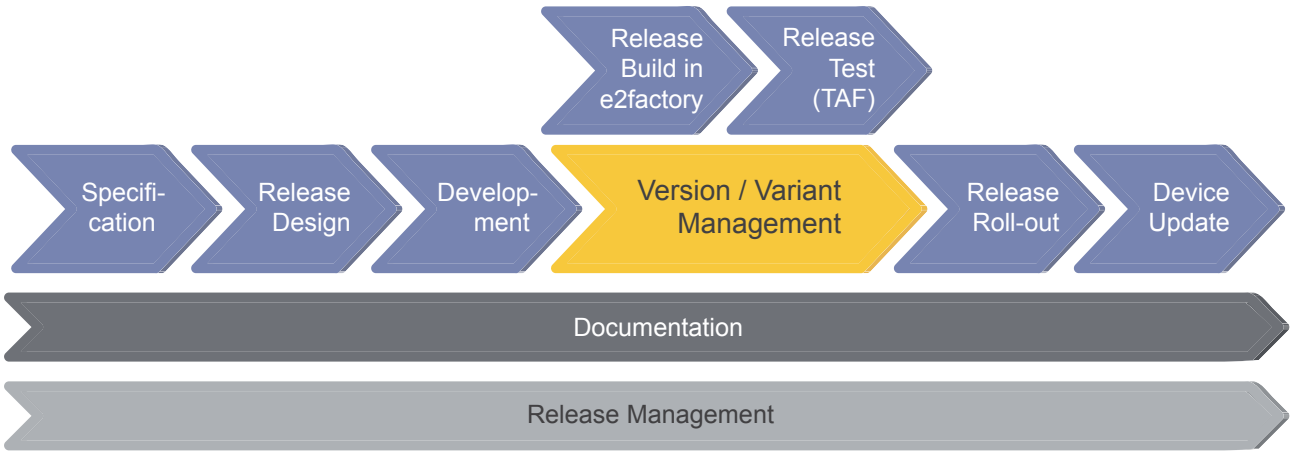
As a matter of principle, emlix only becomes active after consultation with the customer and never makes modifications to the software that have not been agreed. Once a month, the integrity of the software and the build process is checked.

Traceability of changes

In addition to changes emanating from the community, adaptations to new hardware revisions because of discontinued components, the integration of new features or end-customer-specific extensions can be performed within the framework of a maintenance contract.

All the changes to the software components made during the maintenance phase are reproducible and are versioned just like the build process itself. The typical requirements for traceability are thereby fulfilled.

At the end of a maintenance agreement, all the necessary technical and process infrastructure for this can be passed



With network-integrated products, regular security updates are essential. In order to avoid expensive recertification, accurate software management is obligatory, especially in the field of medical technology.

to the customer so that he can continue the process using his own capacities.

An additional component of a maintenance contract, if desired, is a subscription to the emlix Test Application Framework (TAF) with the project-relevant test set. This makes it possible for project engineers to run regular automated test cycles on their customers' systems independently of emlix and to integrate further tests into the framework.

Based on this service portfolio, maintenance agreements can be very individually designed. Further parameters are the number of developer hours included and the reaction time. The typical requirements of certification can be a decisive factor if these prescribe a documented maintenance process.

A maintenance agreement also ensures that the know-how transfer between emlix and the customer's developer team does not come to an end. During field tests or the early phases of market introduction any problems that arise can be immediately evaluated and jointly solved within the framework of the maintenance agreement.

emlix GmbH

solutions @ emlix.com
http://www.emlix.com

Phone +49 (0) 551 / 30664-0
Fax +49 (0) 551 / 30664-11

Consolidation of Linux systems

At companies that have already been using embedded Linux systems for a long time there are often multiple distributions from different manufacturers and from the community as well as the company's own compilations in use. If this is the case, it can be worthwhile to consolidate these Linux systems and to unify build and software management processes. The result is generally a „core Linux system" that can be centrally maintained and, for example, provided with bug fixes, as well as the BSPs resulting from it for specific devices.

The typical aims of such consolidation are:

- Minimization of documentation, maintenance and further development costs
- Reduction of the person-dependency through transparency and good documentation
- „Central" bug fixes and package updates for a complete product family
- Minimization of the cost of adapting the embedded Linux platform for a new product
- Where appropriate, reduction of certification and recertification costs
- Comprehensive hardware abstractions and thereby independence from the hardware manufacturers
- The greatest possible transparency in relation to the embedded Linux systems used by the company

The target is to be able to adapt the software platform to the requirements of newly developed products at minimal cost over its entire life cycle. Additional costs and risks resulting from repeated adaptations, the common lack of documentation and the high complexity of standard distributions and manufacturer BSPs are ruled out.