1 System for vedlikehold av bilsoftware - SoCam

The tool is used to maintain software used in the control units for a modern car. The following section is a short summary of how this is done for a real car. The system presented in section 2 is a simplified version of the real thing.

The description that follows is taken from the paper "Software handling during the vehicle lifecycle" by H. Alminger and O. Josefsson from Volvo Car Corporation. The description is inserted here in order to give the students a good understanding of the real system and how it works.

When new vehicle software is developed it is handled as follows:

- The company has a configuration management (CM) team that is responsible for
    - Receiving each software component from the software development department and performing initial tests.
    - Keeping track of the status for each software component.
    - Coordinating the release activities.
- When the CM team has accepted the software component:
    - It is moved to the factory's software repository and is available for all factory systems.
    - Receives it own, unique serial number. In this way, each authorized dealer can order software in the same way as he orders any other vehicle part.
- The software is downloaded to each vehicle under production in the factory together with the configuration info. When the vehicle gets the status "factory complete" the databases are updated with all info for this particular vehicle.

The system uses four databases:

- Software archive: the repository for all software components. The archive is searchable by product serial number.
- Action database: contains descriptions of which software goes with each electric control unit (ECU). The content of this database will depend on the vehicle's service history.
- Key database. Contains all necessary PIN codes and keys that are needed in order to update a specific vehicle. We
- Vehicle database: contains information for each vehicle – software configuration hardware configuration (ECU), parameters and vehicle identity.

When the vehicle enters a registered garage for a service action, the following happens:

- The vehicle's current configuration (hardware and software) is read from the vehicle's main computer by a local instance of the maintenance system. The vehicle information, together with required actions, is sent to the factory database.
- A safe client fetches the actions from the action database, compares the content with the vehicle database and performs the following actions:
    - Decides which software components that are needed.

    - Decides which electrical and electronic components that are needed.
    - Collects PIN codes and keys needed for the intended service actions.
- The information fetched by the safe client is packed and sent to the computer at the service garage. This computer now performs the updates and the updates are downloaded to the vehicle.
- After the update, the vehicle’s configuration info is updated and stored in the vehicle and in the vehicle database.

If the replacement of an ECU or an update or a new software component has safety critical implications, the manufacturer does a recall. This is done by:

- Sending the necessary updates and components to each registered garage.
- Sending a message to the owner of each vehicle where he is ordered to contact the garage where he goes for service.

2 Functional requirements

This chapter describes the system requirements. The functional requirements for the system are described in section 2.2. All requirements are number sequentially.

2.1 High-level functionality

SoCam shall support the following services:

- Insert new software and new versions of existing software into a software database.
- Maintain a database that shows which control units (ECU) that needs which software
- Send an alarm (”recall”) if we discover critical defects.
- Assist the garage in updating care software

2.2 Requirements specification

The SOCAM requirements are inspired by, but are not in strict conformance to, the real system used by the manufacturer.

2.2.1 Manufacturer system

1. The CM can enter new software and new versions of existing software into the software archive. Each software component has a unique part number. Versioning is done by sub-numbering. E.g. if a software component has part number 123456, then the first version has part number 123456.0, the next version has part number 123456.1 and so on.
2. The CM can enter an action script into the action database and update an existing one. Each action script gives info on which electronic or electric control unit (ECU) needs which software component. The action script connects each ECU to the software component that is used to control it.
3. The CM can define a vehicle database entry for a new production series. The personnel at the factory site insert the vehicle’s serial number into a copy of this entry and insert it into the vehicle database for each finished vehicle when it leaves the production line. Each vehicle entry has information on
   - Hardware configuration – ECU part numbers
   - Software configuration – software component part numbers
   - Vehicle history log – changes and updates that have been performed.
   - Vehicle serial number
4. The CM can initiate a recall. This is done by:
   - Identifying all vehicles that have the faulty components. In our case, we will only consider software faults.
   - Sending this info plus the new version of all components to all registered garages.

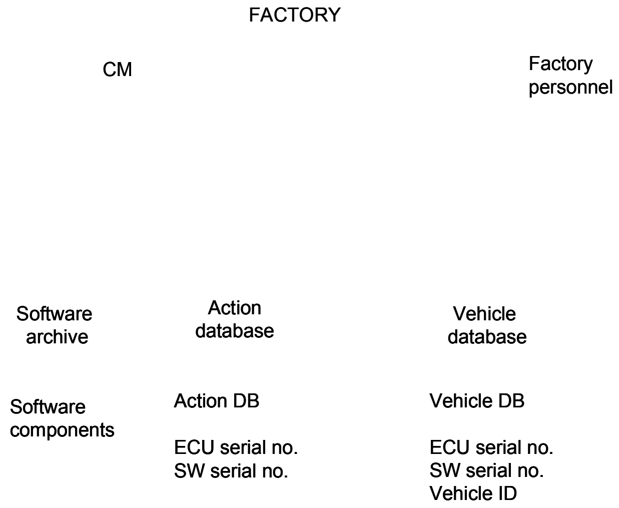

Fig 1: Factory system

2.2.2 Registered garage system

5. Keeps a customer database. Each vehicle sold has an entry with customer info plus the vehicle's serial number.
6. Provide maintenance. This consists of the following actions:
   - Download the vehicle's configuration info from the vehicle's main computer and from the vehicle database by using the vehicle's serial number as a key.
   - Use the two sets of configuration info to identify the latest version of each software component for each ECU in this vehicle.
   - Download latest version of all components that are needed for this vehicle but not yet installed.
7. Service a vehicle. This consists of the following actions:
   - Install all new software components
   - Update the history log and send it to the action database
   - Update the vehicle's configuration info
   - Update the vehicle's configuration info in the vehicle database.
8. Use the customer database to identify the vehicle's serial number based on customer name and vice versa.

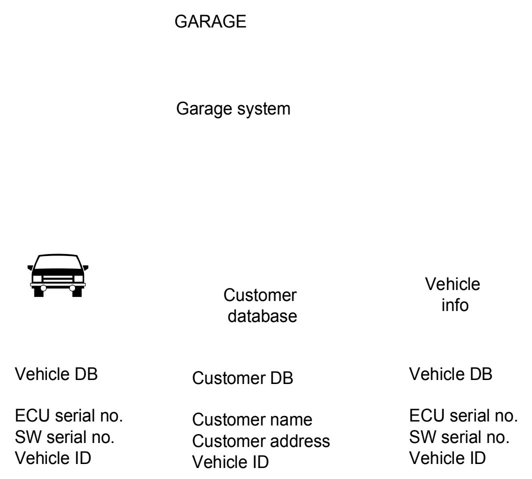

Fig 2: Garage system

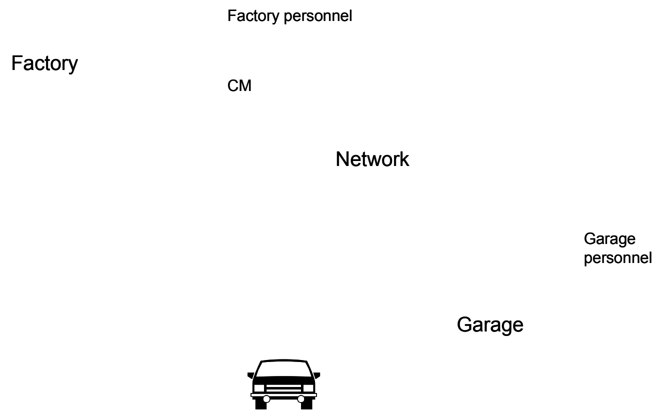

Fig 3: System's overview

2.3 Scenarier

The following scenarios are not part of the system's requirements. They are presented here to assist in understanding and interpreting the system's requirements. See fig. 3 above for a system overview.

When we in a scenario refer to a requirement, e.g. "Requirement 3", we refer to a requirement in the requirement list presented in the previous section.

2.3.1 Scenario 1 – a new version of a software component

The software development department has made a new version of software component 123 which is used to control ECU-48.

The CM responsible logs on to the system and enters the new component into the software archive with part number 123.4 – Requirement 1.

The CM-responsible then updates the action database with the info that the latest version of the control software for ECU-48 is now component 123.4 – Requirement 2.

The CM-responsible logs out.

2.3.2 Scenario 2 – a safety critical software defect is discovered

The software development department has made a new version of software component 176, which is used to control ECU-32. The new version corrects a problem that is considered to be critical for the safe operation of the vehicle.

The CM responsible logs on to the system and enters the new component into the software archive with part number 176.3 – Requirement 1.

The CM-responsible updates the action database with the info that the latest version of the control software for ECU-32 is now component 176.3 – Requirement 2.

The CM-responsible uses the vehicle database to identify all vehicles which uses ECU-32. The list of vehicles obtained is sent to all registered garages – Requirement 4.

The CM-responsible logs out.

Each garage receives the vehicle list and uses their local database system to identify all registered owners in their database that owns a vehicle in the received vehicle list – Requirement 5. All registered owners will receive an email which tells them to schedule a maintenance action as soon as possible – Requirement 8.

2.3.3 Scenario 3 – a vehicle is scheduled for maintenance

A registered owner of a vehicle arrives at his garage for maintenance. The garage personnel use their local system to read the current configuration info from the vehicle's main computer.

The garage personnel logs on to their local system and uses the owner's name to identify the vehicle's serial number – Requirement 8.

They then send a request to the central system at the manufacturer's site to fetch the vehicle's configuration info – hardware and software – and the history log from the vehicle database – Requirement 6.

The history log and configuration info is used to identify the latest versions of all software that has been changed but not installed. These components are downloaded to the local system – Requirement 6.

The new software versions are downloaded to the vehicle's master computer and the configuration info is updated in the vehicle – Requirement 7. In addition, this vehicle's configuration info is sent to the central computer to update the configuration info in the vehicle database.

2.3.4 Scenario 4 – a new vehicle is finished from the factory

A new vehicle series is accepted for production. The CM enters the action scripts into the action file – Requirement 2. This vehicle uses only standard ECUs and the software control components are already available in the software archive.

The CM builds a vehicle database item for the new production series – Requirement 3.

A new copy of this item is instantiated for each vehicle and the vehicle's serial number is inserted when the vehicle leaves the production line. The item is inserted into the vehicle data base when the car has been inspected by a QA engineer – Requirement 3.