How Software Is Changing the Automotive Landscape

Hans Aerts and Han Schaminée

From the Editors
In the Impact department article in the July/August 2011 IEEE Software, Hans Aerts and Han Schaminée described how TomTom was a volume leader in consumer navigation systems and how TomTom applied that volume leadership when it entered the automotive market. Here, Aerts and Schaminée present what has occurred over the past six years and predict what will happen in the automotive industry. —Michiel van Genuchten and Les Hatton

VOLUME HAS BEEN a key driver in the automotive and navigation domains. TomTom started as a high-volume supplier in the navigation aftermarket. Because car buyers are no longer prepared to pay high prices for custom in-car navigation solutions and because of widely available smartphone navigation apps, an in-car navigation system is no longer a strong selling point for a car. This means ownership costs for navigation systems must decrease.

TomTom had been able to leverage these costs with its aftermarket solutions, but that market has matured rapidly and is in decline. Thus, the company had to find another way to secure sales volume. Clearly, that volume couldn’t come from the custom-made, fully integrated head units (receivers) that TomTom had been successfully delivering from 2009 onward. The company couldn’t get sufficient volume from projects in which it had to provide an integrated hardware–software solution. There was too little competitive advantage in the hardware supply.

So, TomTom changed from supplying head units to providing navigation software to existing head unit suppliers. Previous competitors became partners who provided hardware and nonnavigation functionality; TomTom provided the navigation application services and maps. This created greater demand for the navigation stack, resulting in a substantial increase of sold licenses (see Figure 1) and a higher market share. The data after mid 2017 in Figure 1 reflects the expected number of licenses from confirmed deals. The figure is indexed such that 2016 is the reference index 100 (that is, the 2016 statistic constitutes the baseline for comparison).

Exploiting Synergies
Just increasing the demand wasn’t enough to provide benefits from the higher volume. If automotive-navigation projects still continued to have a high level of bespoke engineering, there would still be no way to lower costs, except from the fact that the market as a whole was growing. The need existed to exploit many more synergies between...
the various automotive projects that employed the navigation stack. This automatically brought requirements such as modularity and configurability to the forefront. But the need for built-in quality was also growing. The bespoke-engineering projects often spent much time on manual testing. The belief was that more modularity and better quality would drastically reduce cycle time and project costs. However, these things are easier said than done.

Nevertheless, the investment in modularity and quality was worthwhile. Project costs and cycle times decreased substantially; see Figure 2. TomTom achieved this without decreasing the diversity of available navigation solutions—they all looked different and ran on a variety of platforms.

In this context, a project is a total work package as agreed upon with an automotive customer. A project starts when the contract is confirmed and ends with the product’s release.

**Modularity**

As with many big software stacks, the navigation stack had grown over the past 10 years into a big, monolithic software beast that was difficult to maintain, customize, and integrate on different platforms. TomTom had designed it primarily for aftermarket products, which had much lower requirements for customization and integration. The stack now had to be made ready to enter the automotive market with a high-volume strategy.

First, TomTom split the UI from the core functionality with the business logic, because the UI often must be customized. The company offers the navigation core as a configurable client library with well-documented interfaces.

Another problem was that there were too many code branches, leading to high maintenance. The new setup immediately introduced a strict one-code-branch-only strategy, which quickly reduced maintenance costs substantially.

**Quality**

The navigation core needed to serve many automotive applications. So, it had to meet the highest automotive quality requirements, regarding not only the number of failures but also regression. This required all the teams to focus on quality.

Many teams found out that defining a threshold value for open defects and then solving defects when that threshold is reached (instead

---

**FIGURE 1.** Growth in the number of TomTom software licenses sold. Changing from supplying both hardware and software to just supplying software resulted in a substantial increase of sold licenses and a higher market share.

**FIGURE 2.** With increased modularity and quality, projects became shorter and cheaper. The number in each rectangle indicates the year and month a project started.
of implementing new features) results in larger throughput. Fixing defects early is cheaper and creates a more stable basis for further development. So, agile methods support defect management in the same way that they promote the benefits of limiting work in progress. Thus, TomTom defined limits for the maximum number of open defects for each team. When the number of defects exceeded the limit, defect fixing had priority over feature development, and the reserved weekly capacity for defect resolution increased. Figure 3 shows the decrease in the weighted number of open defects.

Another measure was heavy investment in test automation as part of every build. Regression could be detected early, resulting in higher quality. This supported continuous integration that allowed for early failure detection and a stable platform for further development.

Finally, TomTom introduced advanced static-code-analysis tools. It measured code quality using TICS (TIOBE Software Quality Framework; www.tiobe.com/tics/tics-framework), which supports the software qualities defined in the ISO 25010 standard. TICS covers metrics for code coverage, abstract interpretation, cyclomatic complexity, compiler warnings, coding standards, code duplication, fan-out, and dead code. It analyzes hundreds of projects and keeps track of the top three for small, medium, and large projects. Much of the drive for quality was supported by the teams’ desire to be in the top three of one of the categories. This definitely helped boost the navigation stack’s quality.

Figure 4 shows the results of the static code analysis from the third quarter of 2014 to the fourth quarter of 2016. The dip in 2015 was due to a substantial refactoring of the code base, which created some temporary instability but enabled further improvement of the code quality.

**FIGURE 3.** The weighted number of open defects. A strong focus on quality resulted in a substantial decrease of open defects.

**FIGURE 4.** Improved code quality. The dip in 2015 was due to a substantial refactoring of the code base, which created some temporary instability but enabled further improvement of the code quality.

**Productivity**

To cope with the increase in navigation stack customers, TomTom had to shorten the cycle time of new features. As the speed of innovation in the automotive industry increases, innovation cycle time must decrease. Increased innovation also leads to more uncertainty, and again, the best way to address that uncertainty is to implement shorter cycle times. This allows for fast feedback on the developed features.

To achieve shorter cycle times and thus increased productivity, TomTom implemented SAFe (Scaled Agile Framework; www.scaledagileframework.com). Stable, colocated, self-managed teams, rather than constantly changing project teams, delivered a flow of
features. Initially, the product increment frequency was three months. However, when TomTom’s aftermarket products turned out to require even shorter times to respond to market feedback, TomTom lowered the frequency to two months.

Figure 5 shows the improved productivity in terms of story points per person-week. Story points serve as a unified measure for effort estimation. Because of some organizational-structure changes in 2016, the 2016 statistics aren’t comparable and therefore aren’t in the figure.

Many people have written about agile development’s advantages. It’s often suggested that agility doesn’t work in environments with strong customer commitments. We learned that the opposite is true. Agility provides much more transparency regarding the feasibility of meeting the commitment than do more traditional project management methods. Accurate monitoring of the velocity gives you a good feel for the need for corrective measures.

But agile development’s promises can become reality only if the leadership style changes. Managing by delegating commitment to the team level disastrously affects transparency, quality, and productivity. Teams have no way to deal with the commitment and uncertainty other than by adding window-dressing, lowering quality, or padding their estimates. In contrast, when teams see that the leadership accepts that uncertainty exists and mistakes will be made, they’ll become more engaged. They’ll learn fast, dare to take risks, and be transparent about them, so that commitments aren’t put at risk.

**On the Brink of Big Changes**

Some people say the automotive industry will see more innovation in the next 10 years than in the past 100 years; we agree. (An overview of the changes’ impact on software appeared in the July/August 2017 *IEEE Software*.1) For instance, for many years, the functionality of automotive infotainment systems has been a decade behind that of comparable consumer products. (However, automotive systems’ quality is much better, although consumer systems’ quality is improving rapidly.) It’s questionable whether users still accept the wide gaps in functionality. When big software companies enter the automotive industry, they’ll bring unprecedented innovation.2 Connectivity and over-the-air downloadable software will be important enablers for that.

One important trend in the automotive industry is increased automation. Cars will be able to drive more autonomously, primarily to substantially improve safety and reduce fatal accidents. Of course, the autonomous car won’t arrive overnight, but over time, more and more support will be offered to drivers to make their journeys safer and more
enjoyable. In addition, autonomous cars will normally have defined destinations, which will enable innovations in traffic management systems to balance traffic and improve its flow.

Another trend aligned with automation is energy conservation, which will increase as more and more cars move toward electrification.

Connectivity will be another key driver for change. Increased connectivity will not only allow for over-the-air software updates, as we mentioned before, but also provide more opportunities for cars that are always connected. Many navigation functions (such as routing, search, and map display) can execute on a server, and many applications on the server can be connected. As more processing power is available, UIs can further innovate and include, for instance, natural-language processing. And maybe even more important, connected cars will create an enormous amount of data that will enable totally new businesses.

An additional advantage of connectivity is that the more-innovative functions can be implemented on the server side, while the features embedded in the vehicle can be supported by a CPU that remains in the same head unit for 10 years. This hybrid setup could achieve much faster market penetration for innovative features.

Already for some years, TomTom has provided services such as routing, search, and map display, and the application of these services is growing rapidly, as Figure 6 shows. The figure shows indexed growth and costs, where January 2015 is index 100. The peak in costs in early 2016 resulted from a commercial promotion whose success exceeded the limits of a supplier’s contract. The second peak occurred when TomTom changed the supplier. Currently, an increase in the number of service requests results in a clear decrease of costs per request.

One more important trend is that car ownership will decrease and car sharing will be stimulated. Companies such as Uber have already created a revolution by substantially reducing the cost of mobility, but tons of other opportunities will arise from the connected car.

These trends will ensure that carmakers move away from the traditional vehicle and become mobility providers. The key technology will be software rather than the traditional mechanics and electronics that still dominate the automotive industry. Car manufacturers will likely become user centric rather than car centric. If they don’t do that, big software companies will provide mobility services based on standard vehicle platforms. These companies will also create business models in which people pay for the use of mobility services rather than for ownership.

The Cloud and the Value Chain

On one hand, pay per use rather than pay for ownership is attractive for companies because it creates recurrent revenues. On the other hand, it makes the business models more sensitive to volume. In the past, business cases for feature development were based on pretty reliable predictions of car sales. In the future, they’ll be based on the expected user behavior, which will be more cost-sensitive. Already, customers prefer to buy a feature such as map updating when they buy a new car rather than renew a subscription each year for a fraction of the cost.

From another viewpoint, the cost models show more variation. Whereas once the cost of the in-vehicle CPU was paid when the car was sold, in the future, each request...
to an online service will create cost. Whereas software designers in the past had to be conscious of the hardware constraints, these days they’ve been less concerned because hardware resources seem unlimited. But now, all of a sudden, they have to care about the server-side costs.

The increased flexibility, offered primarily by connectivity, will allow for more innovation. But, as we mentioned before, innovation comes with more uncertainty, and companies should organize themselves for it. And that doesn’t mean just implementing an agile process. It also requires, for instance, totally reconsidering how the automotive industry wants to deal with its suppliers. This relationship will move from contract execution to a collaboration in which the partners equally share the business risks. It’s still true that the best way to manage these risks is to leverage volume.

In 2011, we concluded that navigation had become a volume game. That will be even more the case in the connected world.

References