Strategies for competing in the automotive industry's software ecosystem: Standards and Bottlenecks

Yossi Lichtenstein *Cass Business School, City, University of London*

Stjepan Dujmovic *Robert Bosch GmbH*

Charles Baden-Fuller *Cass Business School, City, University of London*

ABSTRACT The automotive industry includes many actors engaged in software that span small and large companies within old and new sectors. This paper focuses on the controlling position of car manufacturers in the automotive software ecosystem and suggests three strategies for participating software innovators – contesting, cooperating and circumventing. The strategies are exemplified with current cases: Tesla as contestation, Bosch cooperation with a car manufacturer in a data collection project as a basis for future services, and circumventing in a project based on the mobile Internet.

As vehicles continue to evolve into 'mobile computing platforms'[1], the closed automotive software ecosystem – whose standards and rules of access has been dominated for decades by car manufacturers – is opening. If this opening up occurs, that means that independent automotive software developers have the potential to play a more important role in the software ecosystem, engaging directly with automakers and consumers. There is also the potential for open source operating systems and open innovation, especially in the car information and entertainment, or infotainment, layer. This possibility seems especially exciting because on-board software has expanded by a factor of 15 within less than a decade[1], and new car-to-cloud services, such as telematics car insurance policies, are likely to become ubiquitous among millions of drivers[2].

The opening-up of the car software platforms and related ecosystem is not going to happen without some resistance. The strategic management literature advises the established major auto firms to preserve influence, control standards and sustain profitable 'strategic bottlenecks'[3, 4, 5]. This will be challenging as the auto industry is encountering four disruptions at once – automation, connectivity, electrification and sharing. As part of the debate on access to car software and data, our research questions are as follows: What are the strategies available for digital innovators in the emerging software ecosystem? What do these strategies mean for value creation and value capture?

We present three cases that capture a current snapshot of the dynamics of the car software ecosystem – that is, the actors, how they work together and what this means for car digitalization. These cases represent examples of strategies advocated by experts: namely, the strategy of software actors cooperating with the bottleneck owners, the strategy of software actors circumventing the bottleneck, which may be possible through the wider digital ecosystems, and the strategy of software actors contesting the existing bottleneck.

Two of the cases are current projects done by Bosch – that is not a car manufacturer, but a major supplier. The company is the world's largest car-part-maker that employs 400,000 people worldwide. Bosch is very active in software and software innovation, spending nearly a tenth of its revenue on R&D with a strong emphasis on cloud data services. The first project exemplifies cooperation; it has collected about six million datasets from 2.2 million cars through a car manufacturer's licensed workshops. The second project illustrates how the mobile internet infrastructure allows Bosch to circumvent car manufacturers, although in a constrained context. The final case is about Tesla

Motors; we mention briefly the costs, risks and opportunities of contesting the industry's bottleneck by becoming a car manufacturer.

Strategic Bottlenecks

A strategic bottleneck is a place that obstructs a flow; for example, if all roads pass through a single bridge, then the bridge is a bottleneck. In the context of a software ecosystem, a strategic bottleneck is ‘a critical part of a technical system that has no – or very poor – alternatives at the present time’[3]. In the automobile industry, the major car manufacturers have historically controlled the industry’s architecture, for both hardware and software. That is, they have controlled who can do what (division of labor) and who gets what (division of revenue)[4].

Strategy scholars (such as Teece and Jacobides) suggest that software actors who are either inside, or wish to be a party in the ecosystem have a choice between three strategies:

1. Cooperate with a strategic bottleneck owner. A low risk strategy that limits the innovators ability to profit from its work.
2. Circumvent the strategic bottleneck. A risker strategy that raises the possibilities of profits.
3. Contest the strategic bottleneck. Bottlenecks are unique or expensive to copy. Contestation has the greatest opportunity to create and capture value, but it requires large investment and may be risky.

The automotive software ecosystem

Cars are possibly the most complex artefact that consumers own. The car is expensive, heavy, fast-moving, operates in public spaces and is regulated for safety and environmental effect. A typical car contains about 30,000 mechanical, hydraulic, electrical and electronic parts, between 20 and 90 Electronic Control Units (ECUs) and millions of lines code.

There are a dozen or so large car manufacturers (OEMs for Original Equipment Manufacturers). Over the last few decades, they have captured around three quarters of the industry’s value added, measured by total market capitalization[5]. This dominance is based on the manufacturers’ role as system integrators, their control of the customer experience, and their regulatory accountability. The industry’s slow clock speed also helps this domination[5].

Even though the car appears to be a very physical object, software controls car components such as fuel injection, steering systems and brakes since the 1980s. In 2003, Autosar (Automotive Open System Architecture) and its related software ecosystem was established by a partnership manufacturers and suppliers. Autosar includes a standardized set of hardware-dependent ‘software components’ and a large set of hardware-independent application interfaces. This enables the development, integration and validation of applications independently and cater for multiple suppliers. There are many related standards and tools, for example, standards for diagnostics and calibration (ASAM), guidance for static analysis of C programming (C MISRA), and the Japanese car software standard (JasPar). It is notable that Autosar requires that the final deployment of software into the car will be done by the car manufacturer, preventing deployment by third parties or consumers[7], thus making the related software ecosystem closed.

The current industry software ecosystem includes hundreds of companies that develop software for the vehicles’ Electronic Control Units (ECUs). Many are traditional hardware suppliers, like Bosch, that are also engaged in software and the related physical-subsystems. There are also many that sell only software that is not embedded in hardware. In addition, there are several open source automotive software projects and related communities such as COMASSO and BUSMASTER; these are active and popular projects, but not yet major players in the industry.

Currently, the automotive industry is encountering four disruptions – automation, connectivity, electrification and sharing – that are all related to software. In response, OEMs try to shape the future ecosystem[8]. Standard-setting consortia are currently enlarging their scope to include communication, telematics, user-interface modelling, etc. Several operating systems of the vehicle's communication control have emerged, including Automotive Grade Linux and Android Car. Commercial initiatives include HERE, a mapping company which is majority-owned by BMW, Audi and Daimler. OEMs offer connected car applications, such as Volvo's Sensus, Volkswagen's Car-Net; they provide maps and hand-free connection to the digital world. Ford has already integrated Amazon's Alexa into one of its models, Android Auto and Apple CarPlay reports on hundreds of compatible car models.

One result of these developments is new car digital services that are already offered. For example, insurance products that monitor driving behaviour and reduce premiums for careful drivers have been taken-up by almost one million customers in the UK[2], and Tesla famously extended remotely the range of some of its electric cars during Hurricane Irma. Detailing these services and initiatives is beyond our goals. However, a possible future industry architecture may be seen through a recent position paper published by the European Automobile Manufacturers Association. Given considerations of safety and privacy, the association's policy is that OEMs have the responsibility for transferring data from vehicles and that third parties will be access data only via the OEMs.

Our research questions are what are the strategies available for digital innovators in this emerging software ecosystem? What do these strategies mean for value creation and value capture?

Three strategies and cases

To illustrate the three strategies we present three mini-cases.

The first shows how Bosch is cooperating with a car manufacturer in order to collect data from millions of cars, as they are used by customers. Currently, the cooperation is achieved at a technical level and its current value lies in Bosch obtaining new shared knowledge that can be used to improve current components and systems. Bosch believes that these benefits have been considerable. In the near future, we expect similar technology to materialize as predictive maintenance services. For example, a driver will be able to buy a service that will monitor the car subsystems while driving and suggest crucial maintenance prior to a long trip. Another example may be temperature controlled trucks, their operators would buy a predictive maintenance subscription to prevent mishaps while carrying expensive cargo.

The project concerns an Electronic Stability Control (ESC) system that controls the braking system; a system that includes sensors, actuators, pumps, ECU and related software[9]. Bosch obtained the right to collect and analyse the data, and in return, Bosch is conducting annual workshops to inform the manufacturer's engineers. The data collection module was developed as part of the ESC software. The sensor and actuator drivers were written in C, while data and control flow were modelled with the ASCET code-generator, which is popular for embedded software. The full ESC software project employed four engineers, and followed a documented waterfall methodology (V-Model) and Automotive SPICE process assessment.

Data collection was planned carefully: the project defined a-priori what data to measure in order to answer specific engineering issues. Such data included minimal or maximal values, averages, histograms or other multi-dimensional structures. The resulting dataset was only 1.7k bytes, and the software updated it frequently during the vehicle operation. The data was collected at the manufacturer's licensed workshops/garages using On Board Diagnostics protocol, transferred to the manufacturer and then to Bosch. To-date, 6 million data sets have been collected from 2.2 million cars, worldwide. These datasets created new understanding of the actual operation of the braking system along its life.

Although Bosch believes that this project has had value, Bosch is limited in the kind of data it can access and the ways it can exploit it. Other companies, considering this strategy, may be similarly constrained, but they may also feel that additional value will be created when offering new services based on data, for example predictive maintenance. In other words, the current project is anchored in a manufacturer-supplier dyad, but future projects should create a more open ecosystem where other players will provide new services based on similar data collection technology.

The second example shows how Bosch has been able to circumvent the car manufacturers control by its exploitation of open standards and moving towards an open vehicular software ecosystem. The case is a Wrong way Driver Warning (WDW) service which is possible because of the openness that mobile phones introduced into the ecosystem. Interestingly, the core components of the technology – including maps and algorithm – are also aken from the public domain.

WDW warns drivers when they attempt to enter a highway in the opposing direction of traffic. An average of 265 fatal wrong-way crashes occurred annually in the United States, resulting in 355 fatalities[8]. Recently, Nissan, Toyota and Mercedes installed warning systems in luxury models.

Bosch started to develop a cloud-based WDW in 2015. The team includes 15 programmers, data scientists, quality, operations and project management experts. They use Scrum as the agile development methodology, Java for the backend, NodeJS for the frontend and Python for the algorithm. GPS is used to identify location, and Open Street Maps are the reference. The team has found that the maps are accurate and are updated often. Identification of wrong-way-driving is done with a Monte-Carlo algorithm called ParticleFilter executed 'on the cloud'. The system was initiated as a pilot in Germany in April 2017, and was downloaded by 40,000 drivers; it 'travelled' 90% of the 31,000 driveways in Germany and identified one true wrong-way drive, and misjudged only 0.004% drives as wrong-way.

At present, value and value capture for Bosch are uncertain and many possibilities are yet to be explored. For example, Bosch could charge drivers, but has also considered charging insurance companies. Despite these uncertainties, Bosch is pleased with the project as it fits its established automotive expertise and its new open data skills.

Finally, the direct way to contest the bottleneck is to become a car manufacturer - a strategy adopted by Tesla Motors. Investments in Tesla from 2004 to 2017 are approaching 12.5 billion dollars. Currently, Tesla's customer experience is good, but mass manufacturing has not been realized yet. Specifically, car owners report high satisfaction rates and there are over 500,000 customers who ordered Model 3. However, Tesla has manufactured up to now only 250,000 cars and production delays continue and its negative free cash has exceeded one billion dollars a quarter[11]. So, at present, Tesla's contestation is promising but still not successful.

Tesla's contribution to the automotive industry's software ecosystem is already being felt. Tesla's approach is simultaneously open and closed. Tesla's over the air update of car software and other software-related innovations promote demand for the connected car and pushes the industry towards software and connectivity. Similarly, Tesla's announcement that 'it will not initiate patent lawsuits against anyone who, in good faith, wants to use our technology' (Elon Musk on 12 June 2014) promotes openness. On the other hand, Tesla prefers vertical integration in manufacturing[12] and its software operation is also relatively integrated. It announced that it will not provide a Software Development Kit to create third-party apps for its user interface (Elon Musk on 25 January 2015) and its software and data are available only to its suppliers. Indeed, Tesla promotes software innovation, but in a closed ecosystem.

Discussion

Authoritative observers predict a transition of the car industry 'from hardware- to software-defined vehicles'[1], which may also open up the software ecosystem in this industry. This short note considers the strategies software innovators should take to participate in this transition. Applying ideas about industry architecture and strategic bottlenecks[3, 4], we expect that car manufacturers will remain central[5], although new level of openness will emerge as a result of connecting cars to digital platforms[8]. The three simple strategies we suggest reflect these conclusions, namely innovators can cooperate with car manufacturers, circumvent them with the new digital platforms or try to become a car manufacturer.

The current-day illustrations of the three strategies show their viability, but also raise many questions. The strategy to cooperate with a bottleneck owner exemplifies the need for specific deep expertise as well as access to millions of cars for several years, in order to create new value. We suggest that it is still unclear how services such as predictive maintenance may emerge. Will the car manufacturers be ready to share enough of the value with the experts? Will there be others in the ecosystem that can create such services? The strategy to circumvent shows that value can be created quickly and relatively inexpensively using public domain resources and standards. However, it is still unclear how to commercialize such services. Will drivers be ready to pay? If not, who will be ready to subsidize the service? And the strategy of contestation shows with Tesla the investment and risk, but also that the disruptor keeps its own software ecosystem closed.

At a larger scale, the analysis of the automotive industry as it is entering the new era of the connected, electric and shared car (not yet autonomous), shows the commercial realities of technology. We have seen in the last few years how web and mobile standards can enable monopolies. Similarly, the manufacturing scale and customer experience ownership of the car manufacturers will continue to be central to cars as mobile computing platforms. Innovators should straddle strategically openness and control as they are offered by these companies.

References

1. **O. Burkacky, J. Deichmann, G. Doll, and C. Knochenhauer. Rethinking car software and electronics architecture. McKinsey & Company, February 2018**
2. **Ralph, O. Drivers put the brakes on car insurance with a black box, Financial Times, 11 August 2017.**
3. **Baldwin, C. Bottlenecks, modules, and dynamic architectural capabilities. Harvard Business School Working Paper, 2015.**
4. **M. G. Jacobides, , T. Knudsen, M. Augier, M. Benefiting from innovation: Value creation, value appropriation and the role of industry architectures. Research policy, 35(8), October 2006, 1200-1221.**
5. **M. G. Jacobides, J. P. MacDuffie, C. J. Tae. Agency, structure, and the dominance of OEMs: Change and stability in the automotive sector. Strategic Management Journal, 37(9), September 2016, 1942-1967.**
6. **S. Jansen, M. A. Cusumano. Defining software ecosystems: a survey of software platforms and business network governance. Software ecosystems: analyzing and managing business networks in the software industry, 13, 2013.**
7. **U. Eklund, J. Bosch. Architecture for embedded open software ecosystems. Journal of Systems and Software, 92, 2014, 128-142.**
8. **M. Bosler, C. Jud, G. Herzwurm. Platforms and Ecosystems for Connected Car Services, Proceedings of 9th International Workshop on Software Ecosystem (IWSECO 2017), November 2017.**
9. **Reif K. (editor). Automotive Mechatronics. Bosch Professional Automotive Information, Springer, 2015.**
10. **M. Jalayer, M, Pour-Rouholamin, H. Zhou. Wrong-way driving crashes: a multiple correspondence approach to identify contributing factors. Traffic injury prevention, 19(1), August 2017, 35-41.**
11. **Lex, Tesla: flat battery. Financial Times, 2 November 2017.**
12. **Reichow G., Tesla's Secret Second Floor, Wired, 18 October 2018.**

Dr. Yossi Lichtenstein is a Research Fellow at Cass Business School, City, University of London. He also teaches Economics of Digital Technology at University College Dublin. Contact him at yossi.lichtenstein@city.ac.uk

Dr. Stjepan Dujmovic is leading a Bosch-wide project for using vehicle data for product improvement. His research interests include the technical, organizational and legal aspects of vehicle data collection and analysis. Contact him at stjepan.dujmovic@de.bosch.com

Prof. Charles Baden-Fuller is the Centenary Professor of Strategy at Cass Business School, City University London and is a Senior Fellow at the Wharton School, University of Pennsylvania. His current research examines Business Models and their deployment in an increasingly digitalized world. Contact him at c.baden-fuller@city.ac.uk