Transcript: 2020 CES Aptiv Investor Event
Las Vegas, NV | January 7, 2020

APTIV CORPORATE SPEAKERS

- Elena Rosman, VP Investor Relations
- Glen De Vos, Chief Technology Officer

PRESENTATION

Elena Rosman:

Slide 2: Good afternoon. Welcome to Aptiv's technology theater here at CES 2020, which marks our 24th appearance at the show. On behalf of the Aptiv leadership team, we're excited to showcase Aptiv's unique position as both the brain and nervous system of the vehicle, which is really making Aptiv the only architecture company capable of conceiving, specifying, and delivering the advanced hardware and software architectures that are enabling the future of mobility.

So today, our presentation is really going to focus on Smart Vehicle Architecture, which coincides with all the conversations that we're having with our customers here at the show. And SVA, as we refer to it, is really the sum of everything that we do Aptiv. What you see here on display is the embodiment of our SVA philosophy and concepts for future electrical and electronic systems that are enabling the software defined vehicle of the future. The way that today's program will run, Glen De Vos will give an overview presentation and then we'll take a break. We'll invite you all to grab a refreshment in the Aptiv cafe while we clear the technology theater of all of the chairs. And then we'll invite you back so you can participate in a number of our technology displays.

Here on my right we'll be talking about advancements in radar, enabling our active safety business. In the center, we'll do a deep dive further on our SVA hardware center display. And then on my left, we will be talking about the open server platform, which is running a number of really cool SVA enabled software applications.

So with that, I'd like to turn it over to Glen De Vos, Aptiv's Chief Technology Officer

Glen De Vos:

Slide 3: So we're here today because our industry is really at an inflection point. The megatrends that we've been talking about, safe, green, and connected, the consumers are demanding more and more feature content associated with those mega trends, all of which is driving more software, more compute, more requirements, that fundamentally the classical architecture approach simply cannot support.

We've really hit the wall in that regard. And when you think about Aptiv, Aptiv with our capability, we're uniquely positioned to solve those challenges for the OEM. And when you think about these

megatrends, with safe, green, and connected, what really gets us excited about solving those challenges is we don't have the opportunity just to meaningfully improve lives, it's really the opportunity to save lives.

Slide 4: So when you look at some of the products that we currently provide with safety as our advanced safety systems where we're enabling the vehicle manufacturers, essentially to build vehicles that will ultimately result in zero accidents, zero injuries and ideally zero fatalities. For green, our high voltage, distribution and connection systems are enabling the electrification of the powertrain. At the same time, our optimized architectures are de-weighting and lightening the car so that overall we minimize that total vehicle impact to the environment. And then with connectivity, it's that seamless integration between the passengers, the vehicle and then to the world around it. Aptiv's secure connected gateway manages the flow of information into and out of the vehicle, whether using many, many different types of protocols; cellular, satellite, Wi Fi, GPS, RF, LF, you name it. At the same time, we're increasingly enabling over-the-air updates, which means that vehicle software and firmware can be managed and updated and improved over the lifecycle of that vehicle. All of this, by the way, opens up new opportunities for Aptiv to create value and monetization and new business models.

Slide 5: So we've talked a lot in the past about content growth, but what's driving that? And it's really that whole feature growth in the architecture complexity that's driven by essentially software enabled features that are coming into the vehicle, which in turn drive more compute, more sensors, more complexity. But that complexity causes significant challenges for our OEM customers in the areas of miniaturization, quality and labor. And really, they simply cannot manage that complexity today. This is the pain point.

Slide 6: As we think about that software feature driven growth, what happens is more and more of these features are being enabled by software, which means more and more compute in the car. And while you may have linear future growth, you have exponential growth in compute. And fundamentally, where that drives the challenges for our customers is in development, manufacturing, and post production. And I'll talk a little bit more about each of these specifically. But managing that complexity, you'll hear that as a recurring theme and in what we talk about today.

Slide 7: So with Aptiv, as we've talked about, with our brain and nervous system portfolio and our systems integration capabilities, we are perfectly positioned to win. And we started talking, I think, Smart Vehicle Architecture, about two and a half years ago in Boston at one of our investor events. And since then, we've worked with virtually every OEM around the world, and have received two advanced development projects on SVA. And so what's exciting about that is it's really validated our belief not that this was going to happen, but that it is happening. And fundamentally, with our technologies, it allows us to develop those future platforms where we can sustain above market growth in all areas of our portfolio.

Slide 8: So let's talk about where we are today and what's driving this complexity and what is creating the opportunity. So if you look at the chart here, 100 controllers, 200 million lines of code, and 1000

interdependent functions that use all of those. So if you think about putting all of that together and making it work, over every use case, over every scenario, it's a massive task. And really, one of the greatest quotes from our customers was “the complexity is killing us”, “it's paralyzing us.” It’s limiting the ability of our OEM customers to innovate, to drive reduce cycle times to be able to deliver incremental features in a cost effective manner. That's today's architecture, though. And what it says is, you really have to, you cannot continue down the current path.

Slide 9: So as we think about the challenges that it creates, it really affects us in three areas. There's development, there's manufacturing, and there's post production. When we talk about development, the issue there is when you think about software today, it's a monolithic approach. It's a monolithic architecture. And what I mean by that is the software and the hardware for each of those ECUs that we showed, is tightly coupled. They're not decoupled, they're completely integrated and tightly coupled. What that means is it's very difficult to reuse, it's very difficult to change, it's very difficult to qualify and to validate. And that's the headache that our customers are going through. That's what we do as a systems integrator, actually. That's one of the values we provide to our customers, the ability to manage all that.

So we see it firsthand as you bring all of that together. That complexity slows down development times, it puts a massive risk to vehicle launch. In fact, software integration and software debug becomes the biggest risk area for vehicle launch now and ultimately from an OEM, they start over again. Once they get it done, you get through a launch, you breathe, you take a breath, relax for a bit, and then you get to start the entire cycle over again. That's where we are today. And as you add all those more ECUs us into the box into the vehicle, it just makes the matter even worse.

When you think about manufacturing, the harness is, physically the harness is to connect all of those boxes, are just not manageable anymore. It takes 10 people on a premium vehicle to manage the installation of those wiring harnesses into those vehicles. And it’s an incredibly difficult task, you get cut leads, damaged wiring harnesses. So from a manufacturing standpoint, it's absolutely incredible how much of a pain point that is for them. In addition, where do you put the next box? The vehicle is packed. There's just not space for routing of bigger harnesses, more boxes, more devices into that vehicle. So from a manufacturing standpoint, the complexity of the packaging and the assembly has just gotten beyond their control.

And then finally post production. And if you think about vehicles today with that approach, there's really no way to upgrade or improve that vehicle once it's left the customer or once it's in the customers hands. With the exception of Tesla really, OTA is not deployed and that means any issue that's in that vehicle is with that vehicle until it comes back into the dealership. That's current state. As you add more complexity, the likelihood of more problems goes up which means post production management becomes even more of a problem. So those are just some of the challenges. But those are the main challenges that our OEM customers are facing.

Slide 10: So as we think about that, and as we think about that from our unique position of having the brain and the nervous system, understanding architecture holistically in the vehicle, we then believe we understand how to solve that problem for our customers. And really, it consists of three basic elements.

The first is to simplify. And what we mean by that is simplify the vehicle complexity and that interdependence. So you want to simplify how that software is architected on that vehicle. You want to unite. And what I mean by that is, take all of those diverse applications I mentioned earlier. One function has to maybe travel through five or six different boxes, sensors, controllers, displays, you name it. You want to simplify, unite that, so you close that loop so it's much more manageable and it's much more maintainable across the vehicle development cycle and the vehicle lifecycle. And then finally, and this is really critical. Empower, empower the OEM to fully control the software that's in the vehicle, the features that are in the vehicle, and ultimately, the user experience of their vehicles for their customers.

And, you know, we've heard a lot, I think, over the last six months to a year about OEMs wanting to become software companies or own the content in the vehicle. What's really behind that and the issue that's really under that, is they want to control the software in the vehicle. Today, they don't have control. If you think back to that chart with all of those boxes, guess what? A whole bunch of different suppliers provide that software, like Aptiv, integrate all that software and help them do that, but they don't control that. And so what they're really after is how do they get control of that so they can add value where they're best able to.

Slide 11: So how do we actually do that? How do we empower? How do we unite? How do we control that? So there's three basic principles behind SVA and Lee Bauer will walk you through that with a demo here today, because it's beautifully illustrated on the platform.

But there's three steps and the first and the first principle is to extract software from hardware. Today, like I mentioned, software is tightly coupled with hardware, which means you can't reuse it, you can't port it over easily to other applications. It's a hardware software development activity. So the first very important principle is abstraction of software from hardware.

The second is abstracting I/O from compute, input/output from compute. And what we mean by that is if you think about all the data from the sensors, if you think about the data coming from actuators, sensors on the vehicle, other controllers, all of that data flow tends to be tied directly back to the controller. An example would be as if your mouse was hardwired to your laptop. You have it completely tightly coupled. And what we want to do is separate that out so that compute can exist independently of all that surrounding data and power.

And what that does, when you've done the first two, what that enables you to do is what we call serverization of compute. And that means you take those hundred different ECUs and you now bring that compute together. And so you basically consolidate the number of boxes, you simplify and eliminate all of that redundancy, that duplication, all of that inefficiency associated with that distributed architecture, that fragmented architecture. You now centralize or serverize that compute. And if you

think about that, that's very analogous to what's done in other verticals. When we were here a year ago, I think you heard Lee talk about, you know, the mobile computing market, the server, the server in the cloud industry. Other industries have done this. So we understand, hey, serverization of compute is that third critical step.

*Slide 12: (*Redacted)

Slide 13: So, now let's talk about some of the benefits associated with going in this direction. Why do we want to do this? Because, ultimately bringing all these features into the car and the technology in the car is great, but as I mentioned we have to be able to solve the total cost or the total cost of ownership for our OEM customers. So as we think about this approach, it's not just about tech and enabling features, it's about lowering that cost. And I'll walk through the various areas of design and development, manufacturing and post production, and how this saves our OEM customers total costs.

Slide 14: So let's first talk about design and development. So when we think about how we design today, speed to market obviously is very, very important for us. But the fact is, as we've added more feature content, as hardware is tightly coupled with software, that complexity is actually lengthening design and development cycles, not shortening them. So for our industry, we're going the wrong direction. And a big part of that is because with that monolithic software approach, software design ends up predominately following hardware design. Until you have your target hardware, you really can't get into the debug of that heart software on those vehicles. So there's a sequential nature to today's current development processes that extends it out quite a bit.

When you've decoupled or abstracted software from hardware, what you now are able to do is develop hardware and software in parallel. Because basically, you'd have standard interfaces between the two. So, you know as a software developer what your hardware is going to look like, you know what it needs to perform. So you've now decoupled those. That development can be done in parallel. And what that allows you to do is significantly simplify the software development cycle, as well as de-risk the validation and it's a massive improvement.

The other point is, with that decoupling or that abstraction of software from hardware, you no longer have software development tied directly to a model year launch or a vehicle platform launch. You're maintaining that software, basically, on an ongoing basis every day. A launch point just becomes a release point for that software. It isn't the end state, what we used to call feature complete. You now continue to evolve, improve and develop that software over time.

And what that ends up doing is for a development program, you get about a 75% lower system integration testing cost. The standard software or standard software interfaces, centered hardware, you have a much lower system integration and testing cost. And then if you think about this approach, with the advent of OTA, issues that appear in the field when they appear are much more manageable than today. If you think about today, what do you have to do you have to bring that vehicle back into the dealership so that you can re-flash that vehicle. Without OTA, that's not possible for the vast majority of

software related issues. So significant savings in simplification during the development phase of the vehicle.

Slide 15: So if you look at the diagram, you can see this is an existing architecture for a premium vehicle. And you can just see that what really looks like a rat's nest of electrical wiring connectors, everything. That gets replaced by Smart Vehicle Architecture. So that's the transformation that we're really talking about here. A dramatic change in how vehicle architecture is implemented, and installed in the vehicle, that planar multi-layer assembly we talked about. By doing this, you really are addressing many of those manufacturing concerns. First of all, the simplification and the architecture, realizes just a reduction in weight and packaging space required for compute, like up integrating all those boxes by eliminating all those independent connectors by eliminating all the mounting for those, keeping in mind that every box has to have clearance around that box for installation, removal, etc. So all of that consumes a massive amount of volume. By up integrating that compute into these servers and power data centers, you get at least a 25% reduction in weight and packaging space required.

In addition, when you think about that rat's nest that we showed earlier, we love that content, but from a vehicle assembly standpoint, that's a nightmare. That's the 10 people we talked about earlier. You get a significant labor savings now when you think about installing this type of assembly with advanced cable management, as opposed to a lot of loose wires. And that also directly applies to the reduction in factory floor space to staging of the wiring harness becomes much more efficient, the room that's required for the installation of that harness, the storing of the harness and everything else. The other issue is, we now actually have optimized the harness, so we get a benefit just from weight reduction on the hardest itself.

And then finally, SKU reduction. With this type of approach, more and more of the end model proliferation or the part number proliferation or the vehicle proliferation is done in software, not hardware, so you're dealing with many fewer part numbers. And for me, that's a fairly conservative estimate. We can really think about narrowing and reducing the number of SKUs, which simplifies just that whole logistics side of vehicle manufacturing.

So significant benefits from the vehicle manufacturing side and what I would highlight is, a lot of times you hear about vehicle architecture from the standpoint of the software complexity in the silicon and all of this, that tends to be the starting point for that. The reality is we're getting as much pull from the OEMs on the just the wiring harness and the electrical distribution side and the connection system side because of these issues. That's as strong of a driver for these advanced architectures as the content. It really is. It's really been an interesting experience for us and a learning for us over the last two and a half years that I talked about, as we've talked to OEMs how important this is for them.

Slide 16: And then finally, if you think about post production with the life of that vehicle, you know, today the vast majority of vehicles once they're sold, the dealerships really hope not to see them again or the OEMs don't want them coming back for repair. And in 2014, the majority of warranty costs actually flipped from hardware to software and continue to grow in that trend and as more software

comes in, those costs continue to grow. So we think about how is software organized for Smart Vehicle Architecture. While you've abstracted it and what the chart shows each of those little blocks, you can think of that as a software module. And so with that abstraction of software from hardware, we're now able to modularize and basically containerize all those software modules. And for the OEM, create a library, a certified library, of proven software features, applications and functions. They can then extract from that those modules what they need for a particular vehicle build. Much, much simpler and kind of as in what I talked about earlier, a much simpler development and deployment model than today. Now over time, we'll add new modules to that software library, we'll make existing modules better, we'll fix bugs will do all of that that will happen on a continuous basis. So the OEM's software library will continue to grow and become better and better on a continuous basis.

Now, what can happen is, with OTA and with this architecture supporting it, the OEMs are able to push out those updates to add those features to that vehicle at any point in time in the future. So what that does is that means that car will continue to get better, can have feature enhancement over time, and really, you know as we've seen, and this is an area where Tesla has been a great example, the OEMs can enjoy a completely different customer relationship than they have today.

Now, what that means for us is if you think about what Aptiv does, we develop those, we develop quite a few of these modules, we do the integration, we do the deployment, but now we have the opportunity for new business models associated with the maintenance, the development and the ongoing support of those modules, independent of what that vehicle platform is doing. So this is what really enables that potentially recurring revenue stream or other new business models beyond the traditional on-vehicle content. And it really enables the OEM to achieve essentially elimination of a hundred percent of the costs associated with model year updates. That's that huge investment in launch costs that happen as OEM is getting ready to launch a new platform with all new software. That goes away. That's an incredible benefit for our OEM customers.

Slide 17: So I mentioned earlier, you know, one of the things that we're excited to talk about is the fact that this is happening. But the question keeps coming up, well, how does this actually roll out? What's the timing? How do the OEMs adopt these architectures? And as we've talked earlier, there's really if you think about it, we talked about the complexity that's driving it. One of the other factors that is driving and as a move towards electrification. When an OEM goes with a battery electric vehicle architecture, it's the ideal point for them to basically start with virtually a clean sheet on that vehicle. And as you've seen the acceleration of BEV architectures and electrification in the car, what that translates to is even stronger push for these new vehicle architectures and SVA.

So where we are today is really domain centralization. We have 11 programs today with OEMs around the world that are doing significant domain centralization. When I talk about those programs by the way, this isn't like doing a controller or an infotainment system for a vehicle. This is doing the ADAS system that goes across that OEMs complete range of vehicles. So these are very large scale programs. And that domain centralization, which is our satellite architecture that we talked about, that's happening today, and especially as you hit level two systems from an ADAS perspective, we are seeing a

very strong moving in that direction. That's really the first step on this journey towards Smart Vehicle Architecture.

The next, which is really exciting is the zone controller or breaking the vehicle into zones. And what I mean by that is, if you think about the wiring harnesses that we showed earlier, they go from the back of the vehicle, to the front of the vehicle, from side to side. That's why you get that massively complex wiring harness. And while we love complexity, you can't simply add more copper and more length to that. So you break the vehicle into zones typically into quadrants or into whether six zones or two zones, but fundamentally, you break the vehicle into zones. And by doing that, that's how you want to extract I/O for compute. But also means that you can get the wiring harnesses and all of the electrical functions in that zone centralized into a zone controller. And we call that our power data center.

So that's starting now. Our first zone controller will launch in 2022 and we're in active discussions with a number of other customers about zone control programs. But that's that next step on the journey to SVA. And again, we're seeing tremendous traction of OEMs adopting zone control or zonal architectures. As we said earlier, breaking I/O from compute, abstraction of software from hardware, that really sets the stage for the next step, which is that software defined vehicle where you have centralized compute now. And depending on the OEM, it may be four compute clusters as you see here, or it could be more or it could be less. Every OEM may have a slightly different philosophy on exactly how they want to allocate compute.

But the point is in 2025, that's when we really expect to see that centralization take hold. So over the next years is when that really gets started and over this year and next really get started with launch points in 2025. And at that point, that vehicle now has full SVA and can really be thought of as a software defined vehicle.

Slide 18: So SVA as we think about it, we really think about it as a platform. And the chart here is our attempt to explain what we mean by that and where we play, where others play, and how this can work. Essentially, if you look at the bottom of the chart, that's SVA and it consists of all the underlying hardware as well as the software architecture that supports all of the applications that are in the upper portion of the chart. And what we're trying to show here are some examples of what some of those applications might be. And the key point here is applications can come from anybody. They can come from OEMs, they can come from Aptiv, like point gesture to search, audio enhancements or ADAS algorithms and features, or they can come from third parties. The point is SVA really is a platform onto which you can apply those applications. And you can manage that software development and deployment.

And so as we think about it, one of the questions we always get is well, who does the software and how does that all work? What we're saying here is what we provide with SVA is that platform. That's essentially all of the underlying infrastructure that enables that software development and those feature developments to occur on top. And so what's exciting about this is really, it's an it's an extension of our existing portfolio. So if you think about all the products we make today as part of our brain and

nervous system portfolio, it's all of that plus all of these other opportunities for us to create value and growth. And so this is what gives the OEM the control that they want over the features that are going into their vehicles. And so, a great example for third party, we're launching with Volvo infotainment system that has native Android and has all of the Google Automotive Services. It's a great example of how this type of an approach allows third parties to come in, but come in in a way that the OEM dictates and that the OEM controls. For our content, there's a couple examples there. But that's where, for example, with some customers, we do the full ADAS feature set, for others we integrate their features. So again, it really gives you that flexibility to meet the OEM's needs for where they want to add content, where they want to bring it in manage content, but all of that sits on what we call SVA as a platform.

Slide 19: So to summarize, if you think about what we've just covered, SVA is a sustainable architecture. What it does is really two things. It allows us to really deliver that feature rich, high content vehicle, software defined vehicle of tomorrow. But it also allows us to do that while reducing total operating costs for the OEM. Why is this so important to Aptiv? Because when we think about our portfolio, our capabilities, we believe this is what allows us to continue to grow our business at above market rates in both of our segments, both AS&UX as well as S&PS, and to do so really well into the next decade. So for us, you'll hear us talk a lot more about it as part of the theater, but this is just as exciting as it can get.