

Sanjay-Salunkhe-President-and-Global-Head– Digital-and-Software-Services-AI-day-Session-4

Good morning everyone, a lot of jargon for all of you today, this morning, so I'm going to add a few more, but hopefully it's easily understood by all of you there. Thank you, my name is Sanjay Salunke, I'm based in the Boston area, and I run something called as the software engineering business for Hexaware. This is the area which has been most impacted by the early onset of AI. This was the part where coding agents came out and made all this difference that code can be now written much faster by agents. So, AI SDLC is the big part of what you today hear as the impact of AI. I sincerely hope that the impact of AI does not limit itself to only coding, it starts positively impacting businesses in the future and doing much more than just coding.

So, in that perspective, amongst the zero-based services which Keech talked about, I look after zero tech debt, zero backlog, zero defects, and zero licensing, which we talked about earlier. So those are the things which fall on my portfolio. I want to bring you back to how AI has progressed over a period of time. We started off with AI with early code complete type of tools, something which would complete a code. It was like sentence completion type of a tool, and that was a cautious, easy adoption of AI in the IT world. Slowly people were adopting it, people were finding it useful, some of them got very excited by it, and frankly our own tool, RapidX, was built in that era. We in fact originally started with thinking of building our own coding agent, and then we said no, there are going to be a plethora of coding agents in the market. We pivoted and built our own code comprehension agent. We started looking at how was code written in the past, how do I capture all that coding agent logic on how code was written, and we really started the foundations of our code comprehension engine in that era. Fast forward a year, adoption of AI became viral in this space. Everyone had some kind of a tool in the market. Copilot was sold to everyone, Kiro was sold to someone, Cursor was there, Gemini was there, there was every company had some kind of a tool, and people were producing code very fast with AI. At the pace of AI, the code was getting built. At the same pace, you are creating defects, you are creating workarounds, you are creating vulnerabilities. Fast forward went into that place, people started realizing, okay, I'm creating a new demon here. I have to manage the defects. I don't know that everything can go to production. But in that era is when we also established our harness product exactly to solve this problem, that you are going to produce code at pace, but is this code production rate that it can hit the market? Can it be the code which will manage your tokens coming out and so on? So, all of that started forming in that second acceleration phase.

We have now reached a phase where things are settling down. People have started understanding the power of AI. We are making it mainstream. It has opened up and exposed areas which are completely different for the CIO. In fact, the entire tech market has suddenly gotten exposed for the CIO, and that's where even productivity expectations have started being managed for the CIO. If you are doing a greenfield project, so just look at the next diagram there. When you look at these dimensions of what you are building, if you are doing a greenfield project, productivity can be 80 percent. You don't have any legacies. You are really just producing fresh new code. The context is new. Everything is new. You are building a new app. Startups are doing that. You are going to produce code very, very fast. But as you

start looking at the productivity expectation when there is massive legacy sitting with you, things change. When you have layers and layers of applications which you have to talk to, and they are interdependent, intra-dependent, productivity actually changes. You start getting the human element more and more in those situations. You will start seeing that business now wants to engage much faster with the IT teams. This is happening because now IT can realize products and features which the business wants much faster. Earlier, it was a situation where business would give a specification. Six months down the line, nine months down the line, the feature came out. By that time, the need has gone away, and now that gap has closed, and this business engagement has become bigger. Again, all of that is creating an impact on productivity. So, how do our offers actually tackle these problems? Where are we seeing the opportunity for our two offers which fall under this category, zero tech debt and zero backlog? Zero tech debt is our offer where we look at the offer in three archetypes. We have the functional modernization archetype where the function itself, when you started modernizing your old code, it could be a mainframe era code, it could be a client server era code which was just 10 years back, or it could be an older cloud native code. When you started looking at that application, you started thinking, do I need to rework this function? Do I need to change this function completely now that I'm modernizing it completely? So, that's when it's a deeper engagement. You really need to understand the code, what it did before, what you don't need out of that code now or that application, and what you now need to do for the new one to be built. That becomes a full engagement activity there. The non-functional parts are when you are just doing technology upgrades. You have a Java 7 version. Java 7 was the very early versions of Java. We are now running on Java 21. You just want to do a tech update because 21 addresses vulnerabilities much better. 21 is more compatible with the new way of doing things. That's just a tech update. So, that real estate is massive. That real estate runs in trillions of dollars in the CIO's budget today, and these are the problems which CIO was sitting on. They didn't have solutions at the pace at which AI is dealing with today.

Third area, which is a very big category, are product companies. For product companies, features are survival. If they don't update their tech date, they are dead. These companies will not exist. These product companies are software product companies, big ISVs. They have gone through their journeys from being license-based software to a SaaS product, and now from a SaaS product, now they have to become an agentic product. So, massive, massive opportunity. For us, it is a full green opportunity because this is new business for us, coming at a different pace altogether. Zero backlog actually can touch upon what you were doing before. Backlog is a feature. In fact, this zero-tech debt, many a time, feeds into what we want to create as backlog of features for tomorrow. Many a times, you are building newer products from that angle. So, zero backlog is an area where you are creating new product line setups. AI has also done one more new thing. AI has, like what happened from the waterfall era to the agile era, there is now going to be a change in process. And that change in process also is bringing how to do things differently. So, a lot of those areas, both zero-tech defect and zero backlog are feeding into a lot new, different type of pipeline which is getting created for us. What are the problems these two areas are solving for us? So, these two areas are actually interdependent with each other. As I talked about zero tech debt, once you start resolving it, you want to modernize it, you want to do forward engineering with a different story there. You get into a level where you start writing these new requirements. Writing new requirements has become a fantastic task nowadays. Writing new requirements, you engage with the business and say that this is what we had before. Do you want it different now? We

have a tool which we use business vibing for, where we work with the business, creating this new way where business engages now with IT saying, hey, I don't want this. Can you change this to something different now? Business vibing has gotten inserted in that whole process. And tomorrow we take that same story to a zero backlog story, fast forward, realizing the product at a much higher pace, faster pace.

Because coding is no more a problem. The real problems both these offers are now solving. Problem number one it solves is that it is actually solving the trapped maintenance related problems in the CIO's budget. This is a big, big challenge. The CIO is feeling trapped that I have this massive tech debt and this has suddenly gotten exposed. Keech talked about ethos. This entire legacy tech debt which was sitting in your house has suddenly gotten exposed to vulnerabilities. It was not known to people, but ethos has started exploring things which were never possible before. So that has become a nightmare for the CIO to deal with.

Second big problem is that you are sitting with this tech debt. This is now coming directly in your way to do new things with AI. You cannot do new things with AI at a pace with the tech debt dragging you back. So it has become a real problem to solve. We are talking about a half a trillion based services business which is now delivered, which IT services companies are doing maintaining tech debt today. This morning's economic times talks about a 80 billion dollar tech debt legacy modernization time. We've talked about it. There's 220 billion lines of cobalt code still in production. Some in your areas, financial services and banking carries a big load. Retail systems are still running on old mainframe systems. So massive, massive opportunity for updation, massive opportunity for doing transformation for that set of debt. There's almost 12,000 plus TIPCO installation companies, 12,000 plus companies still deploying TIPCO in their site, which is a middleware solution which is such a dated solution, middleware which actually manages as a hub for activities in the future is still on older solutions. We are already in the API world, we are already into even streaming world and we are still doing things which are much more dated. Tremendous opportunity for us to grow. In fact, where do we differentiate with tech debt, zero tech debt? We have three solutions which we talked about, which was referred earlier in the talk today with Saeed and Keech. We have a fantastic deterministic context engine which actually code comprehension engine which can derive context from your legacy code. It is such a deterministic system. Why it is different? Because it just doesn't look at code as data. It looks at code as a behaviour. It's the behaviour of what you are trying to do with that code itself. It looks at a portfolio of applications where the code has been built with. It brings in all the relationships between the applications there. It looks at code or how, what is the blast radius if I make a change in code here, what all applications it's going to impact. And that's very, very rich information when you're doing modernization.

This was not possible, not available in the past. AI and our Zerovity tool has made it possible. It has gone into the depths of unearthing software and explaining it to the developer on what was built before, what was the behaviour intended from this system before, what were the challenges faced by the developer that time to build this system.

All of that comes out through the Zerovity tool and this makes it a very powerful and accelerated modernization project for you. It becomes much more viable for the customer to do that in a different way. The first two points which I talked about again get added by the massive legacy states which clients carry today. We have a fast-growing set of parsers. These parsers are things which makes this job of doing modernization easy. They can parse your language environments, your tech environments, and quickly start making them readable for

you. This list is growing very fast. Last time when Keech spoke a month back, I think it was a list of 65 parsers. We have added 10 new and this is going to be crossing the 100 mark very, very fast. The last part is 600 plus purpose-built agents. These agents were built to manage this legacy. There are agents like domain mapper agent which is you can map the domain saying that, okay, this is in the banking domain or financial services domain. Can you explain the term? Can you have multiple interpretations of this term? Which application uses it for what use and so on? So, multiple agents which actually cover this entire domain differently. I'm going to take a few examples on the zero-tech debt side on how we are using this in real life with some of our clients.

So, this is an example of a Midwestern mid-sized bank. This bank is a very acquisitive bank. In fact, it has grown by acquisitions. The bank had its own tech debt and then they bought four other banks with their own tech debts. Eventually, when you make acquisitions, there is an efficiency factor which the finance team puts in place that when two things come together, I'm going to eliminate some things and make a more optimal option. Not working, not possible. You added the tech debt more what you bought from the bank and this tech debt keeps on growing and reached a level where they started feeling that we cannot move fast or I cannot acquire more unless I start cleaning up this debt. So, they came to us with a request saying that, can you set up a factory for us? Can you build a factory which will start addressing cleaning up this tech debt in a very agentic way? AI is making it possible. Can you set up this way in a way that I can start feeding it my older tech debt, start merging things, throwing out overlaps, taking out things which I don't want or modernizing them which I want, taking the value of an acquired bank. That function might be very good in this bank.

I want that to be put into my system. How do I do that determination? So, our system actually has helped them do this setup. We have set up a factory for them. It's a massive factory which is running at a different speed altogether. So, we have these planner agents. We have the planner agents which look at the plan saying that, okay, these are the 10 applications to look at. They put a plan on for the IT team on how should we deal with this. What is easily doable? Is there a parser available for that? If it's there, use that. If not, build one very quickly and start dealing with the planning process. The plan gives you an exact execution strategy on how soon can you get this thing done. And that pace then runs into the execution story and there is a reviewer agent who says, yes, there is parity with what we were building before and here are the features you eliminated. Here are the new features which are coming in.

Now, this factory is an ongoing factory with them. We have done tons of tons of applications. The speed at which we are running, the migration time has reduced by 88 percent. And this is now making it viable for them to go after newer acquisitions, viable for them to take quick calls whether the acquisition is going to be fruitful for them or not. Second example is a topic I think I've talked to some of you have already been aware of our big airline client which I've been working with in some of their core systems. This was a project which we launched earlier, more than a year back. It was for the crew management system of an airline. The airline realized after a massive security strike that they need to modernize their systems. They set aside 150 million dollars for their modernization budget and then they said that, okay, now I need to go after this modernization in a structured manner. They took up the crew management system and I'm sure you have dealt with an airline's crew management system, especially in the U.S. or global setup. There are rules and laws by which the crew operates and by which if the shift is over, if there's an eight-hour shift, the shift is over, they cannot,

there are times when the crew actually exits the plane and says there's a new crew coming because five minutes back my shift got over and there's a new crew which is going to replace it. Those are the type of problems which this system deals with. This system has been built over two or three decades and every decade had three or four contract negotiations happening with this crew. I'm sure you have heard the nightmares of airlines dealing with pilot payrolls and crew management and unionized staff which is actually running these new rules and negotiations which are happening with them. Each negotiation run will add new sets of rules and modifications to the system and this was built, the system was built in mainframe era. So modifications were done in the mainframe era and they have carried them, some documented, some not documented. Every new airline which was brought in, their rules came in. Very, very complex system amounting to today's more than six million lines of code. That entire system had to be modernized. 150 million, one part of the budget has now come to us for this. Where it has started with the payroll system, we have now moved down to the scheduling system. The entire project has been now given. We started with five APEX, 25 APEX have been now awarded to us and this is a structured move by this airline to deal with their tech debt. So great example of how and the size which people are dealing with and this is one such system in the airline.

I want to come to the second problem which we talked about today, the backlog. Backlog is a problem and an opportunity. It is, AI is the next big wave. You remember the digital wave which came in? Digital wave turned physical stores into online stores. Digital wave turned your physical banking into online banking. AI is going to do something different in businesses and if it doesn't do, it's a shame. That is the real power of AI and we are starting to see customers make those move today. We are starting to see customers now drift towards that direction where AI is not just a code efficiency tool but it's a tool to build my new, reimagine my business into something different. And the best, if I correctly, we see how that's how we now engage with the customers. Our customers have already invested in AI tools. Some of them have brought co-pilots. Some of them have already invested in co-pilots and are starting to see that, okay, I'm hitting a road there right now on my co-pilot.

How can I make this better usage of this co-pilot? It is not just coding tool setups. We engage with them on a tool-led adoption. We go for them from that agile to agentic move.

You know, if you are familiar with agile as a process, it talks about scrum teams. It talks about regular cadence meetings. It was a tool built for doing coding efficiently and today's world, coding is done by agents.

So, the tool and the process is not needed. It has moved to the problem has shifted from coding as a problem to trusting the code which has been produced. What are the artifacts I can trust and how well can I trust? So, this is a big shift which many customers are seeing in their organization, creating a second set of opportunity for us in the zero backlog. Starting with tool-led adoption, how do I make it usable? Going down, maturing to changing the process in the organization to then moving to big programs, managing the tech debt tomorrow and then launching greenfield programs or new projects come into play. Each of these streams are creating new opportunities for us to create zero backlog for them. Zero backlog for us is clearance of feature as it comes. You have a new idea. You have a new business plan. Business comes up saying, I want this. I have a window of opportunity to open. How fast can IT build this and make it viable for me? You can move this fast. And this is the inflationary opportunity which we see in a big way in the market.

Here our differentiation comes not only with our context engine or the comprehension which was built, but it starts with the harness. The harness is the tool which connects your context with the raw LLMs. Every LLM today comes with some level of basic harness into play. It is not enough and it is not built with the purpose which the enterprise wants. Enterprise wants to put in its own context into the story. It wants to do prompt caching.

It wants to compress the rag in such a way that your token usage is not a runaway. Token economics has become a big story today just because the cost of tokens today has gone from anywhere, any company spends \$500 per user per month to \$2,000 per user per month. User spends around \$2,000 per developer per month, which is almost becoming equivalent to a developer itself as a cost structure. And it's going to go more if you don't harness it well. We are in trouble. We are actually not saving the cost factor which is going to come. We are going to get speed, but we are going to get cost at the same speed. And our harness design, in fact, it's such a well-crafted setup that it manages those token economics in a different way.

The third topic is what I talked to you about. The process needs to change. It cannot be agile. We have, in fact, built a prescriptive process today to really move from that direction to how do I move from agile to a now AI-driven SDLC? How do I deal with this new loops which are coming in where business and IT are very quickly concluding what they want and now I can produce and I can now release it in the market fast and I have to get ready for the next step after that. So that stage is something which we actually help our customers with how process transformation has happened. And again, our 600 agents come into play there to really run this like a factory.

With that, I'm again going to take a few examples in the backlog side of things. These are very exciting examples because this is where the customer is now changing their business model itself. The first one is for a pharmacy benefits manager, a PBM company. These are typically entities which exist in the U.S. competing with the likes of United Healthcare, Optum, or CVS, Caremark, and so on. This is a smaller size PBM. Seeing this as an opportunity that AI is going to allow them to now attack these goliaths which are there in the market. This is a small-size PBM which is saying U.S. introduced a new rule that pharmacies can now sell this product directly to the consumer or the patient or a pill to the patient directly. They are now building these portals for these pharmaceutical companies. This portal allows them to directly reach the customer in a regulated way, the way the government has structured it, eliminating the middleman completely. Now this portal, this window of opportunity is small because everyone wants to do that. And by the time the big guys wake up, they would have lost the market. So they have their original plan where the portal would take them around 12 to 18 months to build such portals with AI, with an AI-driven SDLC. In fact, we have moved that cycle down to one quarter. Within three months, we are releasing, we released five portals for them in the last quarter. Each one of them is running live. Each one of them is actually supporting a new, big, large pharma company in the setup where consumers are now buying from them. It has created a new revenue stream completely for them. And frankly, they have slowed down their older way of working. Their entire process is now creating this pipeline of portals with different pharma companies and trying to disrupt this big goliaths coming into play. The speed at which we build things were there in the metrics. In fact, this is a perfect example where Agile got thrown out over a period of time. Agile was giving them 60 to 70 user story points per sprint. We moved into AI-SDLC. The first run, we came to 150 story points per sprint, 200 story points per sprint. The moment we started using agents into play, the

agent connection became stronger. The number became so fast, they said that Agile is no more needed because the KPI does not matter. KPI of story point has gone way beyond the human production numbers and what am I matching here? So, I'm now producing portals. KPIs have changed where we are releasing portals at a snap of a fingertip. So, this actually is transformative for the business.

The next example, the last example which I want to take to you is about a legal tech consulting services company. This is a 5,000 people legal tech consulting services company which helps a litigation to produce evidence from past documentation. So, if you're going through an M&A, if you're going through a legal contract negotiation, if you're going through a litigation, they do case preparation for them. 5,000 people going through piles and stacks of papers, doing redaction, doing manual redaction in there, doing markups of those documents and then coming to a closure for those setups. The entire process was turned upside down with AI now. So, they have gone from becoming a legal tech human-based consulting services company now to a legal platform company and this is the transformation which AI will drive tomorrow. They have started building their own product platform which does that and this is a perfect example where we call it as AI for AI.

So, we are using AI-based development techniques for building an AI-based product. So, this is a completely on-prem AI product which is releasing, doing these services very fast and in the past the limitation was the number of people who could look at documentation. That limitation is gone because productivity for them has skyrocketed after AI has come into play. So today the cost and efficiency, the review cost has come down by 40 to 50 percent. The document review, now only 5 percent of the humans have to review, 5 percent of the documents have to be reviewed by humans. Earlier it used to be whooping 40-50 percent. Everyone had to go through a proper review till it went out to the market. Accuracy and quality has just skyrocketed up 36 percent plus accuracy on the documentation which is going out. Cases are much more easily handled now. Volumes in which they can now handle this job has just skyrocketed.

So, in summary frankly I feel that this is a tremendous opportunity. Last two quarters we have ourselves seen a change in the pipeline of the type of jobs which are coming up.

CIOs proactively bringing in tech topics, CIOs addressing zero licensing topics which we talked about. So, CIOs coming out now saying that can you launch three programs where I can do complete net new development completely AI natively. We have actually have 45 plus clients who have onboarded our versions of Zerovity platforms in their setup.

They've asked for this platform to be launched so that they can use it and deliver that productivity. And this list is growing every week. We are getting newer and newer clients asking for that. 900 plus agents, 600 of our own purpose-built agents plus a few hundred more agents are in production with the clients. 70 percent of the delivery is now AI assisted for us. So massive massive movements happening with both these offers. We feel that these are going to be our flagship offers building the new business for us for tomorrow. So that's it from my side. Thank you very much.

I'm going to call in Girish Ravindran to take up the next zero. So, thank you again.