

# FROM CODE TO DATA

HERE IS A SAMPLE WRITEUP I DELIVERED AT A DEV CONFERENCE IN PUNE, INDIA IN JUNE 2018. IT EXPLAINS MY VISION ON HOW THE TECH INDUSTRY IS EVOLVING WITH THE INTRODUCTION OF MACHINE LEARNING IN MOST PRODUCTS AND AS A RESULT THE CONSEQUENCES FOR SOFTWARE ENGINEERING AND PRODUCT DESIGN MOVING FORWARD.

THANKS, PHIL  
DEC. 2018



philhora.com  
415-315-9787  
phil@philhora.com

### Important note

All visuals, design, flows and text by **Phil Mora**.

**Disclaimer.** The document presented here isn't subject to any non-disclosure agreement, and is already in the public domain for more than 18 months. Confidential material has been redacted. For demo purposes only.

**Contact.** Phil Mora's portfolio is constantly updated and can be found at [philmora.com/portfolio](http://philmora.com/portfolio). For more information, networking and quotes, please contact me at [philmora.com/contact](http://philmora.com/contact) or DM me on twitter @ [philippemora](https://twitter.com/philippemora)

# FROM CODE TO DATA

(DATA IS THE NEW OIL)

PREPARED FOR YOU BY PHIL



sikkasoft.com  
1731 Technology Drive  
San Jose, CA 95110



# FOREWORD

As the tech world is changing rapidly we wanted to pause a little and take the time to think this through by understanding precisely what is happening and what is the impact on Sikka's operations.

This document is organized in 4 parts, going from a macro-trend review, to business model changes, to what all of this means for Sikka to a few immediate solutions that we may want to consider.

I hope that this will help our teams understand that to stay competitive in a tech world that is changing even faster today, we always need to stay on our toes and never, ever give in.

Thank you!  
-Phil



## INTRODUCTION: 2018 TECH TRENDS

The tech industry is currently experiencing a few major and fundamental shifts. Let's start by briefly exploring the macro-view in innovation via 3 major driver trends we see today in order to put all of this in context.

One of the first things we want to look at is personalization. With personalization, data improves engagement in experiences and drives growth and scrutiny. We see now that personal collective data provides better experiences for consumers. We're now at 2.2 billion Facebooks, 200 million Pinterests, 170 million Spotifys and 125 million Netflixes. The value comes both from people who are putting their data into these products to make their experiences better and the collective data of many other users that effect a lot of these products, in real-time, for example with Waze, SnapMap, NextDoor, or Uber Pool.

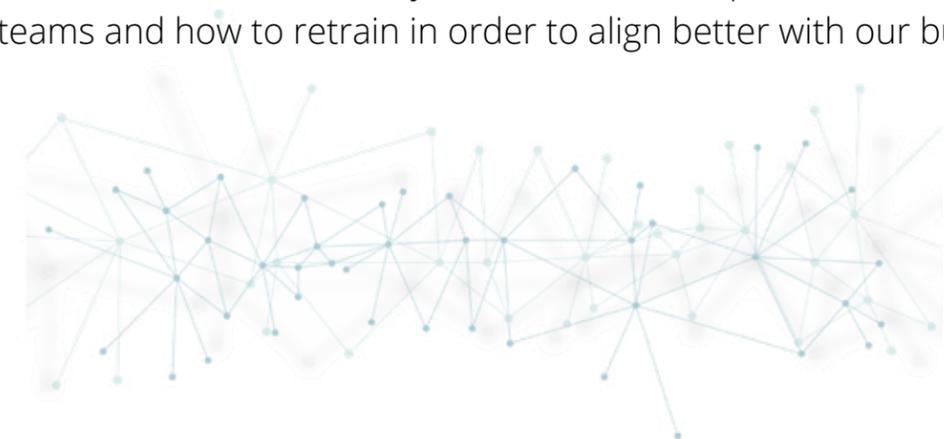
Second, there is data gathering and optimization, which has become increasingly global and competitive and is accelerating. In the past decade, we have lived through two computing Big Bangs dealing with consumer mobiles and the cloud. The cloud started in 2006 with Amazon AWS and in 2007 with the Apple iPhone. The first result has been a collapse in computing costs followed by a similar drastic decrease in storage costs. Meanwhile data gathering, sharing and optimization was enabled by consumer mobile adoption, social media adoption and sensor pervasiveness. The collective amount of user data is now growing at a mind blowing pace, from 2 ZB (a zettabyte is a trillion gigabytes) in 2010 to a projected 46 ZB in 2020.

Data today improves the predictive ability of many services (we'll get to that later) and data volume is foundational to algorithmic refinement in Artificial Intelligence modeling, performance, predictability and capability. It's rapidly becoming foundational to tool and product improvement.

And lastly – lifelong learning is becoming absolutely crucial in the evolving work environment, and luckily the tools are getting better and more accessible. Coursera has 33 million learners, up 30 percent year on year, with top courses such as machine learning, neural networks and deep learning, introduction to mathematical thinking, algorithms, and neural networks. Lifelong learning educational content usage is also ramping very fast. YouTube has more than a billion views of daily learning videos, 70 percent of users use the platform to help solve work, school or hobby problems.

As a result of those macro-trends we see in tech today, two things come to mind: the way tech companies are conducting business as well as the skills necessary to achieve success are changing rapidly - which does have a direct impact on Sikka's organization and employee pool. (But the good news is that the ways to acquire knowledge today are becoming virtually free).

We're going to first have a look at the micro-economics of the equation, in other words how the business models are changing in the tech industry and the impact, or how Sikka is responding to the shift. Second, we will look at the changes in how tech products are built and the skills required today and in the future to build them. And finally we'll look at the implications for Sikka's dev ops teams and how to retrain in order to align better with our business goals.



## A FUNDAMENTAL PLATFORM SHIFT: DATA NETWORK EFFECTS

As discussed previously, today we are undergoing one of the largest platform shifts in a generation as applications move to the cloud and are consumed on iPhones, Echoes and Teslas, while being built on new stacks and fueled by AI and data. And with it, the way tech companies conduct business is also changing .... and with it the employee skillset necessary for tech companies to stay ahead is also being rapidly disrupted.

But before we look at these implications, let's have a quick look at how the tech business models are evolving from network effects to data network effects by inserting AI-driven systems of intelligence in the current business models.

### Traditional tech business models

To establish themselves as powerhouses companies like google, facebook, amazon, or microsoft all were built on economies of scale and network effects.

- Economies of scale: the bigger a company is the more operating leverage thus lower costs. SaaS and Cloud Services have strong economies of scale, they can scale revenue and customer base while keeping the product's core engineering flat
- Network effects: a product or service has "network effects" if each additional user of the product accrues more value to every other user. Slack, WhatsApp, Facebook are good examples of strong network effects. Same for windows, iOS and Android because the more users the more apps are built on top of it

- IP/Trade Secrets: most tech companies always start with proprietary software and/or methods - for example new solutions to hard technical problems, new inventions, techniques, and then patents to protect the new intellectual property
- High switching costs: if a customer is using your product, it may be difficult to switch to a competitor (example the "walled gardens" of windows and office)
- Brand and customer loyalty: a strong brand can be a good defense: with each positive interaction between the product and the customer, the brand advantage gets stronger over time ( ... only to quickly evaporate if customers lose trust in the product, example Chipotle)

### Going full stack

Today the market is favoring "full stack" companies, i.e. SaaS offerings that offer application logic, middleware, and databases combined.

Technology is becoming an invisible component of a complete solution (Example no one cares what database backs your favorite mobile app as long as your food is delivered on time). In the consumer world, Apple made the integrated or full stack experience popular with the iPhone which seamlessly integrated hardware with software.

This integrated experience is coming to dominate enterprise software as well. Cloud and SaaS has made it possible to reach customers directly and in a cost-effective manner. As a result, customers are increasingly buying full stack technology in the form of SaaS applications instead of buying individual pieces of the tech stack and building their own apps.

## Today's SaaS stack: systems of engagement and systems of record

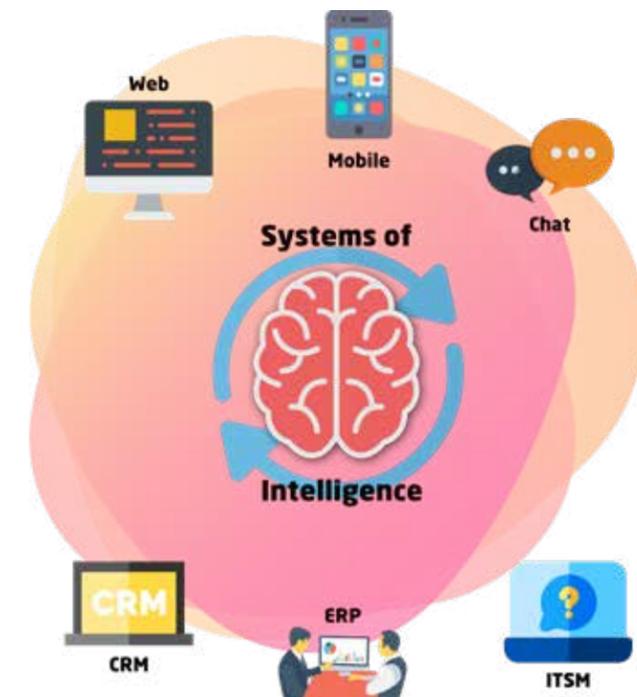


- Systems of record: At the bottom of the SaaS stack is usually a database on top of which an application is built. If the data and app power a critical business function, it becomes a “system of record.”
- Systems of engagement: Systems of engagement are the interfaces between users and the systems of record and can be powerful businesses because they control the end user interactions. In a multi-channel world, owning the system of engagement is most valuable if the business controls most of the end user engagement or is a cross-channel system that can reach the end-users wherever they are

## The new tech business model: data network effects

At the core of data network effects are AI-driven systems of intelligence, which typically cross multiple data sets by inserting themselves in between systems of record and systems of engagement.

Success is then achieved by using customer and market data to train and improve models that make the product better for all customers, which spins the flywheel of intelligence faster. Ultimately the product becomes tailored for each customer (that’s the “personalization” aspect we talked about in the intro).



## THE END OF CODE

So, we have talked about the trends we see in the tech marketplace, going from data aggregation to making sense of data in order to enable full personalization of a product that, in a way, self evolves in the hands of the user as they continuously interact with it. We also discussed that these tech product trends are making users evolve rapidly in their relationship and expectations, and with it, the tech business models are rapidly changing from pure user network effects and economies of scale to data network effects and AI-driven systems of intelligence.

Now let's talk about the implications this may have for the skillsets required to implement and successfully deploy those business models.

**“Soon we won't be programming computers.  
We will be training them like dogs.”**

Software has already eaten the world and today we're all surrounded with machines that convert actions, thoughts, and emotions into data - the new raw material (“data is the new oil”) - for armies of coders to manipulate.

Life itself seems to be ruled by a series of instructions that can be discovered, exploited, optimized and perhaps re-written as companies use code to understand us as intimately as possible from sleep patterns to spending habits and everything in between.

As a result, today, the ability to code has become not just a desirable skill but grants insider status to those who speak it fluently: “if coders don't run the world, they run the things that run the world” (Bloomberg).

However, coders should not get used to it: our machines are today starting to speak a different language, one that even the best coders don't fully understand.

Indeed, over the past several years, machine learning in Silicon Valley has aggressively become front and center, and for very good reasons. See, in traditional programming, an engineer writes explicit, step-by-step instructions for the computer to follow. But with machine learning, engineers don't code computers, they train them. Example if you want to teach a neural network to recognize a cat, you don't tell it to look for whiskers, ears, fur, and eyes. You simply show it thousands and thousands of photos of cats, and eventually it works things out. If it keeps misclassifying foxes as cats, you don't rewrite the code. You just keep coaching it.

This approach is not new, the math behind it has been around for decades—but it has recently become extremely cheap due to a massive reduction in storage and processing power costs, as discussed previously.

As a result, today, machine learning already powers a lot of apps, example Facebook uses it to determine which stories show up in the News Feed, Google Photos uses it to identify faces, Microsoft's Skype Translator using machine learning to convert speech to different languages in real time. Self-driving cars use machine learning to avoid accidents.

As machine learning changes drastically user experiences, the world of coders is also definitely changing rapidly: the days of writing linear programs are basically over. Example, after a neural network is trained on speech recognition, as a programmer you can't go back inside and look and see how the learning happened - because when you look inside a neural network you see a multilayered set of billions of data points that constantly evolve in their relationships and simply generate guesses about the world.

## Indeterminacy and un-parsable machine language has direct implications for coders

The current shift is giving users a more rewarding relationship with technology, an experience that is more personal. But this also means that experiences can no longer be reduced to a series of comprehensible instructions, indeed, machine learning basically goes in the exact opposite direction: indeterminacy.

As for the past two decades, learning to code has been one of the surest routes to reliable employment - today the world is increasingly run by neural-networked deep-learning machines, this requires a very different workforce ....

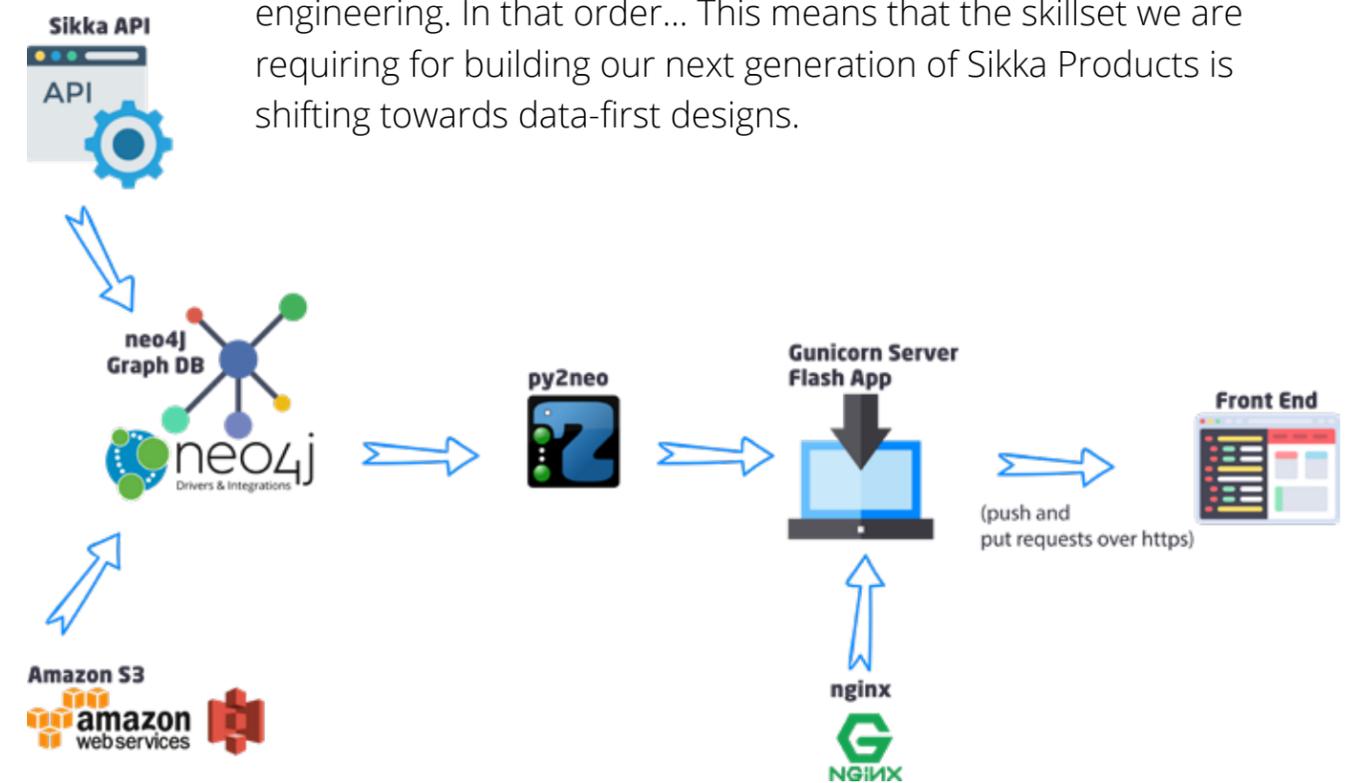
Engineers have to be able to create a combination of handwritten linear code that uses the power of machine learning to adjust it - and to train these systems. This is still a rare combination of skills, because the job requires both a very solid high-level grasp on math and a very good intuition ... In addition to all the "old" programming skills.

At Sikka our dev teams will have to adapt to those changes very soon.

## DATA-CENTRIC AND INFERRING EVOLUTIONAL CULTURE

In the previous sections we have discussed that the days of programming systems linearly with predictable outcomes are numbered and that this cultural shift in software engineering is accelerating: machines today are increasingly not "programmed", they are trained, and data inferencing, not coding, is now the new center of inertia of the DevOps teams.

The Self Driving Practice and aRt are being built as systems of intelligence by data scientists who know classic software engineering. In that order... This means that the skillset we are requiring for building our next generation of Sikka Products is shifting towards data-first designs.

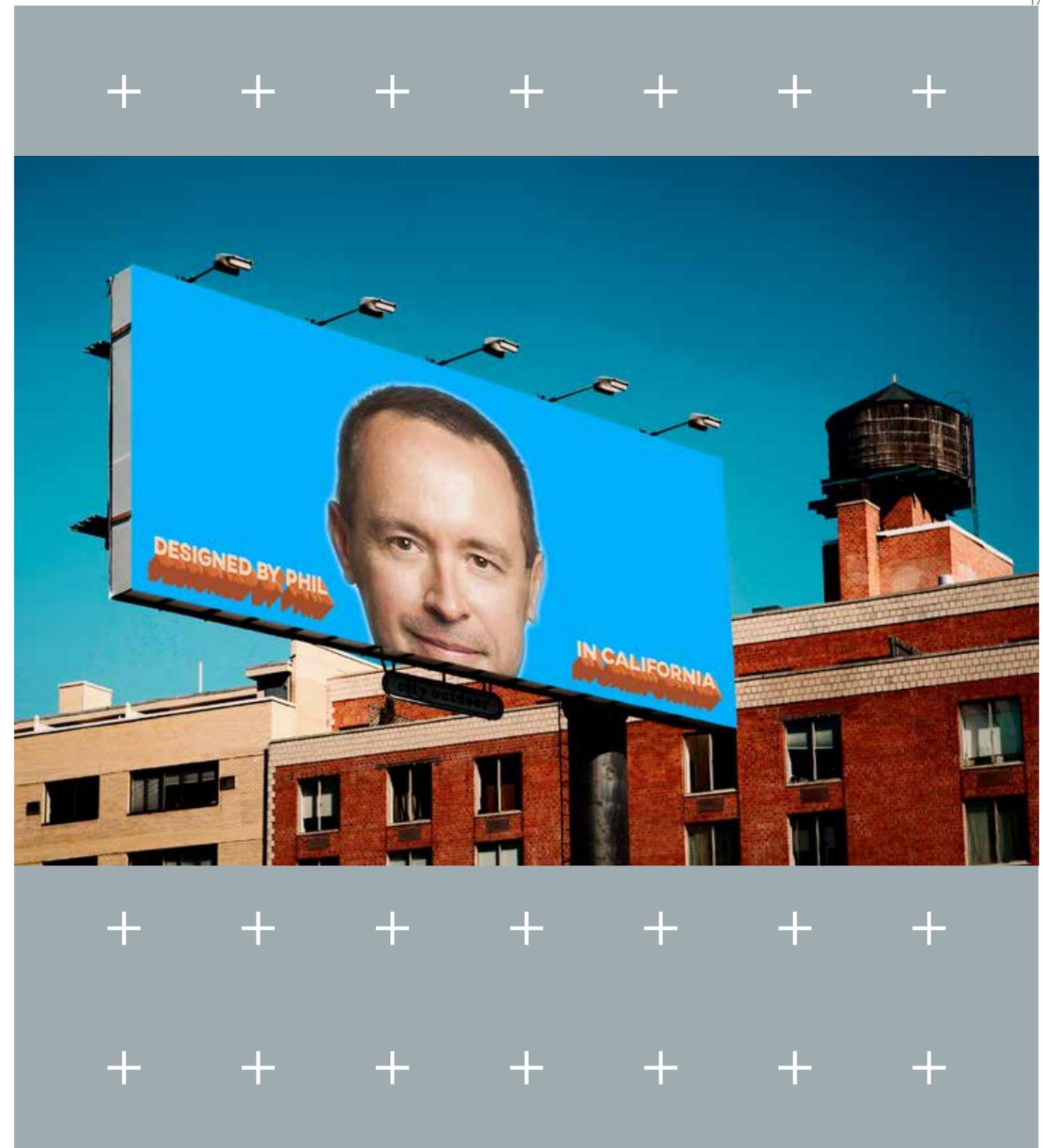


What are the implications for our current Sikka team structure? First of all, we're looking at blurring lines between data science and devOps with the addition of machine learning for personalization (as we discussed earlier) and data at the everything center of inertia. It's almost certain that product managers and traditional marketing are going to disappear, to be replaced by full stack designers who go all the way from design ideation to code and the use of data science techniques towards data marketing science. With the addition of the feedback loop created by usage data, users are directly training their software to perform the way they want, generating inferences at the aggregate of a population of users, which means that no single users will experience the same product identically. This will in turn change the way with qualify and support software. In other words, with Indeterminacy and un-parsable machine language combined with data being the customer (and vice versa) comes a new way of looking at how products are built, maintained and supported.

In order to cope with such a dramatic culture shift, we're looking at enhancing our team skillsets. We discussed this in the first section: lifelong learning today is a given and we believe that at Sikka we have reached this cross-roads where we must learn new stuff while maintaining our existing technology advantage. Or fall behind very quickly.

## ACKNOWLEDGEMENTS

- **Masters of scale**, Reid Hoffman, Greylock Partners
- **The end of programming**, Peter Levine, Andreessen Horowitz
- **The new moats**, Jerry Chen, Greylock Partners
- **Internet trends 2018**, Mary Meeker, Kleiner Perkins
- **The autonomy ecosystem**, Frank Chen, Andreessen Horowitz



I WOULD LIKE TO HEAR FROM YOU

831-252-2610  
@philippemora

