

The Future Of Coding Is Here, And It Threatens To Wipe Out Everything In Its Path

Author : Chet Kapoor

Date : October 28, 2015



APIs — the rules governing how software programs interact with each other — not user interfaces, will upend software for years to come.

When **Intel CEO Brian Krzanich** [doubled down on the Internet of Things](#) at the company's annual Developer Forum in August, he emphasized what many of us have already known — the dawn of a new era in software engineering. It's called API-first **design**, and it presents a tremendous opportunity for developers who adapt — not to mention a major risk for developers (and companies) who don't.

Intel isn't the only heavyweight recognizing the value of APIs. **IBM** recently got into the world of API management with [IBM Bluemix](#), which allows companies to discover how other developers are using their APIs and **design** around that feedback. Then **Oracle** [extended its API management suite in June](#) to capitalize on growing revenue opportunities. Other players have been stacking the deck in preparation for API-centric software development for years.

Typically, when people **design** new products and capabilities, they're asked to **design** the UI screens and show how the user experience will look. There are plenty of reasons this approach took off with developers. Touchscreens unleashed a new generation of computing, and fundamentally changed the ways we interact with hardware.

*Connected devices, driverless cars and advanced health tech are just a handful of the new technologies API-first **design** will enable.*

Apple and **Google** have proven that ease of use is both a consumer and enterprise priority. What's more, the emergence of Augmented and Virtual Reality platforms prove people are constantly exploring new ways to experience content. But as devices proliferate, system-to-system interactions will dominate people-to-system interactions. Systems don't need pretty interfaces, they need well-defined contracts. They need APIs.

Just in mobile, one can think of 10 different interfaces. Then there are web, client-server and thin client — it can be overwhelming. The only way to gain control is to focus on the API layer; it's not even worth thinking through the fragmentation of the interface layer, especially if one is

providing a service. Take **Netflix**, for example. How can a video streaming service with such a simplistic user interface scale to more than 63 million users accessing their video library from hundreds of device types from all across the globe? Excellent APIs.

The Internet of Things (IoT) — which will soon be at the center of the tech universe, [according to Business Insider Intelligence](#) — is really driving this paradigm shift. This proliferation of devices is sealing the deal on a trend already gaining steam.

As devices have outnumbered people, the systems we use to connect them have become incredibly complex. APIs are the foundation of these connections: the mortar between our **digital** hardware. This complexity has set the stage for a tectonic shift away from full-stack engineering toward building pieces on top of existing layers within larger ecosystems.

*API-first **design** presents a tremendous opportunity for developers who adapt — not to mention a major risk for developers who don't.*

Other tech giants like **Apple** and **Google** are driving the same API-centric future. The emergence of new interconnected product categories — wearables like the **Apple** Watch and **Google's** driverless cars, to name two prominent examples — signal the growing importance of APIs in our daily lives. When the smartest people in the room get behind something new, it's best to pay attention. This doesn't mean computers and screens are going away, but it does signal a whole new world of opportunity for developers.

The consequences of failing to move to API-centric development are as real for individuals as they are for the companies that employ them. Developers who fail to adapt their talent around APIs run the risk of rapidly devaluing their skills and decreasing their job security.

For companies, the consequences may be magnified. Startups that fail to embrace this technological revolution could become less competitive. They could make inferior products. Some startups could wither away, altogether. Companies that don't live on the edge of **innovation** will become pieces of a shrinking pie.

As we move into a more interconnected world, amazing new possibilities emerge. Developers like to consume “bite-sized” stuff. **Amazon** popularized this approach — they told developers what the system does and got out of the way. For tech companies, the “telling” will be handing over APIs. It's no wonder we've moved toward microservices that enable best-of-breed platforms to thrive.

Connected devices, driverless cars and advanced health tech are just a handful of the new technologies API-first **design** will enable. For these **innovations** to happen, they must be built on a solid foundation. That means starting system **design** at the foundational layer — APIs.

FEATURED IMAGE: [LIGHTSPRING/SHUTTERSTOCK](#)

//