

5 Key Areas for Tech Leaders to Watch in 2020

Roger Magoulas & Steve Swoyer



5 Key Areas for Tech Leaders to Watch in 2020

Roger Magoulas and Steve Swoyer



Beijing • Boston • Farnham • Sebastopol • Tokyo

Table of Contents

5 Key Areas for Tech Leaders to Watch in 2020	1
In Programming, Python Is Preeminent	3
Trends in Software Architecture, Infrastructure, and	
Operations	6
A Drill-Down into Data, AI, and ML Topics	8
Cloud Continuing to Climb	10
Surging Interest in Security	12
Concluding Thoughts	13

5 Key Areas for Tech Leaders to Watch in 2020

O'Reilly online learning contains information about the trends, topics, and issues tech leaders need to watch and explore. It's also the data source for our annual usage study, which examines the mostused topics and the top search terms.¹

This combination of usage and search affords a contextual view that encompasses not only the tools, techniques, and technologies that members are actively using, but also the areas they're gathering information about.

Current signals from usage on the O'Reilly online learning platform reveal:

- **Python is preeminent.** It's the single most popular programming language on O'Reilly, and it accounts for 10% of *all* usage. This year's growth in Python usage was buoyed by its increasing popularity among data scientists and machine learning (ML) and artificial intelligence (AI) engineers.
- Software architecture, infrastructure, and operations are each changing rapidly. The shift to cloud native design is transforming both software architecture and infrastructure and operations. Also: infrastructure and operations is trending up,

¹ This article is based on non-personally-identifiable information about the top search terms and most-used topics on O'Reilly online learning. We compared aggregated data for the last three years; a full year of data for 2017 and 2018, and through the end of October for 2019.

while DevOps is trending down. Coincidence? Probably not, but only time will tell.

- ML + AI are up, but passions have cooled. Up until 2017, the ML+AI topic had been amongst the fastest growing topics on the platform. Growth is still strong for such a large topic, but usage slowed in 2018 (+13%) and cooled significantly in 2019, growing by just 7%. Within the data topic, however, ML+AI has gone from 22% of all usage to 26%.
- Still cloud-y, but with a possibility of migration. Strong usage in cloud platforms (+16%) accounted for most cloud-specific growth. But sustained interest in cloud migrations—usage was up almost 10% in 2019, on top of 30% in 2018—gets at another important emerging trend.
- Security is surging. Aggregate security usage spiked 26% last year, driven by increased usage for two security certifications: CompTIA Security (+50%) and CompTIA CySA+ (+59%). There's plenty of security risks for business executives, sysadmins, DBAs, developers, etc., to be wary of.



Figure 1. Normalized search frequency of top terms on the O'Reilly online learning platform in 2019 (left) and the rate of change for each term (right)



Figure 2. High-level topics on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each topic (right)

In Programming, Python Is Preeminent

In 2019, as in 2018, Python was the most popular language on O'Reilly online learning. Python-related usage grew at a solid 6% pace in 2019, a slight drop from 2018 (+10%). After several years of steady climbing—and after outstripping Java in 2017—Python-related interactions now comprise almost 10% of all usage.

But Python is a special case: this year, more than in year's past, its growth was buoyed by interest in ML. Usage specific to Python as a programming language grew by just 4% in 2019; by contrast, usage that had to do with Python and ML—be it in the context of AI, deep learning, and natural language processing, or in combination with any of several popular ML/AI frameworks—grew by 9%. The lag-gard use case was Python-based web development frameworks, which grew by just 3% in usage, year over year.



Figure 3. Normalized search frequency of top programming languages on the O'Reilly online learning platform in 2019 (left) and the rate of change for each language (right)



Figure 4. Programming languages on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each language (right)

This is consistent with what we've observed elsewhere: Python has acquired new relevance amid strong interest in AI and ML. Along with **R**, Python is one of the most-used languages for data analysis. From pre-built libraries for linear or logistic regressions, decision trees, naïve Bayes, k-means, gradient-boosting, etc., there's a Python library for virtually anything a developer or data scientist might need to do. (Python libraries are no less useful for manipulating or engineering data, too.)

Interestingly, R itself continues to decline. R-related usage on O'Reilly online learning fell by 8% between 2017-18 and by 6%, year-over-year, in 2019. It's likely that R—much like Scala (-33% in usage in 2018-19; -19% in usage 2017-18)—is a casualty of Python. True, it might seem difficult to reconcile R's decline with strong interest in AI and ML, but consider two factors: first, ML and statistics are not the same thing, and, second, R is not, primarily, a developer-oriented language. R was designed for use in academic, scientific, and, more recently, commercial use cases. As statistics and related techniques become more important in software development, more programmers are encountering stats in programming classes. In this context, they're more likely to use Python than R.

Interest in some languages seems to be trending up, and interest in others, down. Exhibit A: Java-related usage dropped by a noteworthy 13% between 2018 and 2019. Is this the harbinger of a trend? Not necessarily: Java-related searches increased by 5% between 2017 and 2018. On the other hand, Java's cousin, JavaScript, also appears to be in decline. True, theirs is only a conceptual relation, but interest in JavaScript, too, really does seem to be waning: JS-related usage dropped on O'Reilly online learning by 4% between 2017-2018 and by 7% between 2018-19. It's possible that microservices architecture is hastening the move to other languages (such as Go, Rust, and Python) for web properties.

Among the JavaScript-based web application frameworks, React gained in popularity (+4% in usage) as Angular (-12% in usage) slipped between 2018 and 2019. Vue.js—a competitor to both React and Angular—settled down to steady growth (+8% in usage) in 2018-19, after almost doubling in usage (+97%) between 2017-18.

One possible trend-in-the-making is that of a slowing Go, which—following several years of rapid growth in usage (including +14% from 2017 to 2018)—cooled down last year, with usage growing by a

mere 2%. But Go is now the sixth most-used programming language, trailing only Python, Java, .NET, and C++. Drop .NET from the tally on methodological grounds,² and Go cracks the top five.

Trends in Software Architecture, Infrastructure, and Operations

Cloud native design is a new way of thinking about software and architecture. But the shift to cloud native has implications not only for software architecture, but for infrastructure and operations, too. It exploits new design patterns (microservices) and adapts existing techniques (service orchestration) with the goal of achieving cloudlike elasticity and resilience in *all* environments, cloud or onpremises. O'Reilly Radar uses the term "Next Architecture" to describe this shift.

It's against this backdrop that what's happening in both software architecture and infrastructure and ops must be understood. In the generic software architecture topic, usage in the containers topic increased in our 2019 analysis, growing by 17%. This was just a fraction of its 2018 growth rate (+56% in usage), but impressive none-theless. Kubernetes has emerged as the *de facto* solution for orchestrating services and microservices in cloud native design patterns. Usage in Kubernetes surged by 211% in 2018—and grew at a 40% clip in 2019. Kubernetes' parent topic, container orchestrators, also posted strong usage growth: 151% in 2018, 36% this year—almost all due to interest in Kubernetes itself.

² A reasonable enough decision. .NET isn't so much a language as a software framework: i.e., a superset of C# and a few related languages, including Visual Basic .NET, J#, and C ++/CLI, the latter of which is a .NET-specific implementation of C++.



Figure 5. Software architecture topics on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each topic (right)

This also helps explain increased usage in the microservices topic, which grew at a 22% clip in 2019. True, you don't necessarily *need* microservices to "do" cloud native design; at this point, however, it's difficult to disentangle the two. Most cloud native design patterns involve microservices.

These trends are also implicated in the rise of infrastructure and ops, which reflects both the limitations of DevOps and the challenges posed by the shift to cloud native design. Infrastructure and ops usage was the fastest growing sub-topic under the generic systems administration topic. Surging interest in infrastructure and ops also explains declining usage in the configuration management (CM) and DevOps topic areas. The most popular CM tools are DevOps focused, and, like DevOps itself, they're declining: usage in the CM topic dropped significantly (-18%) in 2019, as did virtually all CM tools. Ansible was least affected (-4% in usage), but Jenkins, Puppet, Chef, and Salt each dropped off by 25% or more in usage. It can't be a coincidence that DevOps usage declined again (-5%) in 2019, following a 20% decline in 2018.



Figure 6. Infrastructure and operations topics on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each topic (right)

The emergence of infrastructure and ops suggests that organizations might be having trouble *scaling* DevOps. DevOps aims to produce programmers who can work competently in each of the layers in a system "stack." In practice, however, developers tend to be less committed to DevOps' operations component, a fact that gave birth to the idea of site reliability engineering (SRE). Even if the "full stack" developer isn't a unicorn, she certainly isn't commonplace. Organizations see infrastructure and ops as a pragmatic, ops-focused complement that picks up precisely where DevOps tends to fail.

A Drill-Down into Data, AI, and ML Topics

The results for data-related topics are both predictable and—there's no other way to put it—confusing. Starting with data engineering, the backbone of all data work (the category includes titles covering data management: relational databases, Spark, Hadoop, SQL, NoSQL, etc.). In aggregate, data engineering usage declined 8% in 2019. This follows a 3% drop in 2018. Both years were driven by declining usage of data management titles.

When we look more specifically at data engineering topics, *excluding* data management, we see a small share, but solid growth in usage, up 7% in 2018 and 15% in 2019 (see Figure 7).

Within the broad "data" topic, data engineering (including data management) continues as the topic with the most share, garnering about one-twelfth of all usage on the platform. This is almost double the usage share of the data science topic, which recorded an uptick in usage (+5%) in 2019, following a decline (-2%) in 2018.

Elsewhere, interest in ML and AI keeps growing, albeit at a diminished rate. To wit: the combined ML/AI topic was up 7% in usage in 2019, about half its growth (+13%) in 2018.



Figure 7. Data topics on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each topic (right)

Ironically, the strength of ML/AI might be less evident in dataspecific topics than in other topic areas, such as programming languages, where growing Python usage is—to a large degree—being driven by that language's usefulness for and applicability to ML. But ML/AI-related topics such as natural language processing (NLP, +22% in 2019) and neural networks (+17%) recorded strong growth in usage, too.

Data engineering *as a task* certainly isn't in decline. Interest in data engineering probably isn't declining, either. If anything, data engineering *as a practice area* is being subsumed by both data science and ML/AI.³ We know from other research that data scientists, ML and AI engineers, etc., spend an outsized proportion of their time discovering, preparing, and engineering data for their work. We've seen that popular tools and frameworks usually incorporate data engineering capabilities, either in the form of automated/guided self-service features or (in the case of Jupyter and other notebooks) an ability to build and orchestrate data engineering pipelines that invoke Python, R (via Python), etc., libraries to run data engineering jobs concurrently or, if possible, in parallel.

³ ML and AI aren't in any sense the same thing, either. We combine them here for simplicity's sake.

Terms that correspond with old-school data engineering—e.g., "relational database," "Oracle database solutions," "Hive," "database administration," "data models," "Spark"—declined in usage, year-over-year, in 2019. Some of this decline was a function of larger, market-driven factors. We know from our research that Hadoop and its ecosystem of related projects (such as Hive) are in the midst of a protracted, years-long decline. This decline is borne out in our usage numbers: Hadoop (-34%), Hive (also -34%), and even Spark (-21%) were all down, significantly, year-over-year.

We discuss likely reasons for this decline in more detail in our analysis of O'Reilly Strata Conference speaker proposals.

Cloud Continuing to Climb

Interest in cloud-related concepts and terms continues to increase on O'Reilly online learning, albeit at a slower rate. Cloud-related usage surged by 35% between 2017 and 2018; it grew at less than half that rate (17%) between 2018 and 2019. This slowdown suggests that cloud as a category has achieved such a large share that (mathematically) any additional growth *must* occur at a slower rate. In cloud's case, while growth is slower, it's still strong.



Figure 8. Cloud topics on the O'Reilly online learning platform with the most usage in 2019 (left) and the rate of change for each topic (right)

Interest in cloud service provider platforms mirrors that of the industry as a whole: Amazon and AWS-related usage increased by 14%, year-over-year; Azure usage, on the other hand, grew at a speedier 29% clip, while Google Compute Platform (GCP) surged by 39%. Amazon controls a little less than half (per Gartner's 2018 numbers) of the overall market for cloud infrastructure-as-a-service (IaaS) offerings. It, too, has reached the point at which rapid growth becomes mathematically prohibitive. Both Azure and GCP are growing much faster than AWS, but they're also much smaller: Azure notched nearly 61% growth in 2018 (per Gartner), good for more than 15% of the IaaS market; GCP, at around 60% growth, accounts for 4% of IaaS share.

Also intriguing: cloud-specific interest in microservices and Kubernetes grew significantly last year on O'Reilly. Microservices-related usage was up 22%, year over year, following a decline in 2018. Kubernetes usage was up by 38%, year over year, following a period of explosive growth (+190%) from 2017 to 2018. Both trends mirror what we're seeing via user surveys and other research efforts: namely, that microservices has emerged as an important component of cloud native design and development.

The bigger takeaway is that the essential tendency of modern software architecture—namely, the priority it gives to loose coupling in emphasizing abstraction, isolation, and atomicity—is eliding the boundaries between what we think of as "cloud" versus "onpremises" contexts. We see this via sustained interest in microservices and Kubernetes in both on-premises *and* cloud deployments.

This is the logic of cloud native design: specific deployment contexts will still matter, of course—which features or constraints do developers need to take into account when they're developing for AWS? For Azure? For GCP? But the clear boundaries that used to demarcate the public cloud from the private cloud are starting to disappear, just as those that distinguish on-premises private clouds from conventional on-premises systems are falling away, too.

Surging Interest in Security

Security usage (+26%) grew significantly in 2019 (see Figure 2). Some of this was driven by increased usage in the CompTIA Security+ (50%) and CompTIA Cyber Security Analyst (CySA+, 59%) topics.

Security+ is an entry-level security certification, so its growth could be attributed to increased usage by sysadmins, DBAs, software developers, and other non-specialists. Whether it's to flesh out their full-stack bona fides, address new job (or regulatory) requirements, or simply to make themselves more marketable, Security+ is a pretty straightforward certification process: pass the exam and you're certified. CySA+, on the other hand, is relatively new. This *could* explain the explosion of CySA+-related usage in 2018 (+128%), as well as last year's strong growth. Unlike the CISSP and other popular certifications, CySA+ recommends, but doesn't require, real-world experience. Like Security+, it's another certification sys admins, DBAs, developers, and others can pick up to burnish their bonafides.

Certifications weren't the only thing driving security-related usage on O'Reilly in 2019. A rash of vulnerabilities and potential exploits, too, had some impact. If 2018 (+5% growth in security usage; +22% growth in search) gave us Meltdown and Spectre, 2019 gave us sobering information about the far-reaching implications of Meltdown and, especially, Spectre. For 2019, security-specific usage (+26%) and search (+25%) increased accordingly. System and database administrators, CSAs, CISSPs, and others were keen to acquire expert, detailed information specific to patching and hardening their vulnerable systems to protect against no less than 13 different Spectre and 14 different Meltdown variants, as well as to mitigate the potentially huge performance impacts associated with these patches. Developers and software architects had questions about rewriting, refactoring, or optimizing their code to address these same concerns. Against this backdrop, the spike in security-related usage makes sense.

There was a great deal going on with respect to information security and data privacy, too. After all, not only was 2019 the first full year for which the EU's omnibus GDPR regime was binding, but—as of January 1, 2019—updates to Canada's GDPR-like PIPEDA regime officially kicked in, too. The sweeping California Consumer Privacy Act (CCPA), which has been called California's GDPR, went into effect on January 1, 2020.

Taken together, an analysis of these trends seems to support a glasshalf-full assessment of the state of security today. If the sustained growth in security usage on O'Reilly is a reliable indicator, it's *possible* that security may, finally, be getting the attention it deserves in an increasingly digital world. It's *possible* that organizations have accepted that the financial and reputational penalties entailed by a data breach or high-profile hack are just too costly to risk, and that money spent on information security is, on balance, money well spent.

The same analysis also lends itself to a glass-half-empty assessment, however: namely, that security spending is cyclical; that a confluence of circumstances has helped to boost security spending; and that—let's be honest—organizations tend to bounce back from high-profile security incidents. Only time (or future installments of this survey) will tell.

Concluding Thoughts

It's hard to imagine that the hottest trends of 2019 won't be reprising their roles, in more or less the same pecking order, in next year's analysis. Programming languages come into and go out of vogue, but Python appears poised to keep growing at a steady rate because it's at once protean, adaptable, and easy to use. We see this in the widespread use of Python in ML and AI, where it has supplanted R as the *lingua franca* of data engineering and analysis.

The same is true of ML and AI. Even if (as some naysayers warn) the next AI winter is nigh upon us, it's hard to imagine interest in ML and AI petering out anytime soon. The same could be said about trends in software architecture and, especially, infrastructure and operations. They're each sites of ceaseless innovation. Their practitioners will be hard-pressed to keep up with what's happening.

It's helpful to think of what's hot and what's not in terms of a modified "Overton Window." The Overton Window circumscribes the *human cognitive bandwidth* that's available in a certain place at a certain time. No combination of policies—or issues, or trends—can exceed more than 100% of available bandwidth. This is true, to a degree, of the activity on O'Reilly online learning, too. A decline in usage doesn't have to correlate with a decline in use (or usefulness) in practice. It's just being crowded out by other, emergent trends.

This also underscores why the decline in a bellwether topic such as JavaScript might be hugely significant. Even if these topics are no longer sites of rapid and sustained innovation, they are likewise important for day-to-day use cases, especially with respect to general-purpose information gathering or more specialized problem solving. It isn't that JavaScript is less important than it used to be; after all, React, Angular, and Vue.js are sites of development and innovation, and all three are based on JavaScript. It's that we're coming to understand and relate to JavaScript—or data engineering, or Docker, or DevOps and change management—in a different way. We're appropriating them differently.

It's this difference that Radar aims to capture. Not the change that's obvious for everyone to see, but the coalescence of change itself as it's happening.

Visit O'Reilly Radar for research and analysis on need-to-know trends shaping business and technology.