DevOps

Roles

Code Review?

### Making the Applications — Business Capability Roles

These teams work on the application or services being delivered. The composition of these teams changes over time as each team “gels,” learning the necessary operations skills to be “DevOps” oriented and master the domain knowledge needed to make good design choices.

These are the core roles in this layer:

* **Developer/Engineer** — those who not only “code,” but gain the operations knowledge needed to support the application in production, with the help of…
* **Operations** — those who work with developers on production needs and help support the application in production. This role may lessen over time as developers become more operations aware, or it may remain a dedicated role.
* **Product Owner/Product Manager** — the “owner” of the application in development that specifies what should be done each iteration, prioritizing the list of “requirements.”
* **Designer** — studies how users interact with the software and systematically designs ways to improve that interaction. For user-facing applications this also includes visual design.”.

These are roles that are not always needed and sometimes be fulfilled partially by shared, but designated to the team staff:

* **Tester** — staff that helps with the effort ensure the software does what was intended and functions properly.
* **Architect** — in large organization, the role or architect is often used to ensure that individual teams are aligning with the larger organization’s goals and strategy, while also acting as a consultative enabler to help teams be successful and share knowledge.
* **Data scientist** — if data analysis is core to the application being developed, getting the right analysis and domain skills will be key.

#### Developer/Engineer

These are the programmers, or “software developers.” Through the practice of [pairing](https://en.wikipedia.org/wiki/Pair_programming), knowledge is quickly spread amongst developers, ensuring that there are no “empires” built, and addresses [the risks of a low “bus factor.”](https://en.wikipedia.org/wiki/Bus_factor)Developers are also encouraged to “rotate” through various roles from front to back-end to get good exposure to all parts of the project. By using a cloud platform, like Pivotal Cloud Foundry, developers are also able to package and deploy code on their own through the continuous integration and continuous delivery tools.

Developers are not expected to be experts at operations concerns, but by relying on the self-service and automation capabilities of cloud platforms do not need to “wait” for operations staff to perform configuration management tasks to deploy applications. Over time, with this reliance on a cloud platform which cleanly specifies how to best build applications so that they’re easily supported in production, developers gain enough operations knowledge to work without dedicated operations support.

The amount of developers on each team is variable, but so far, following the two pizza team rule of thumb, we see anywhere from 1 to 3 pairs, that is [2 to 6, and sometimes more](https://en.wikipedia.org/wiki/The_Magical_Number_Seven,_Plus_or_Minus_Two).

#### Operations

In a cloud native mode, until business capabilities teams have learned the necessary skills to operate applications on their own, they will need operations support. This support will come in the form of understanding (and co-learning!) how the cloud platform works, and assistance troubleshooting applications in production. Early on you should plan to have heavy operations involvement to help collaboration with developers and share knowledge, mostly around getting the best from the cloud platform in place. You may need to “assign” operations staff to the team at the beginning, making them so called designated operations staff instead of dedicated, as explained in [Effective DevOps](http://shop.oreilly.com/product/0636920039846.do).

Many teams find that the operations role never leaves the team, which is perfectly normal. Indeed, the desired end-state is that the application teams have all the development and operations skills and knowledge needed to be successful.

As two cases to guide your understanding of this role:

* Etsy has approximately 15 operations engineers to a few hundred other engineers, according to [Effective DevOps](http://shop.oreilly.com/product/0636920039846.do).
* [As re-told by Diego Lapiduz at 18F](http://cote.io/2015/10/02/loc8/), early on when teams were learning how to use and operate cloud.gov, he and a handful of other operations staff spent much of their time with development teams, getting intimately familiar with each application. Now, because the practice of designated operations staff is less needed, he and his operations peers are less involved and have little knowledge of the applications in use…which is good, and as intended!

As a side note, it’s common for operations people to freak out at this point, thinking they’re being eliminated. While it’s true that margin-berzerked management could choose to look at operations staff as “waste,” it’s more likely that following [Jevon’s Paradox](https://en.wikipedia.org/wiki/Jevons_paradox), [operations staff will be needed even more as the amount of applications and services multiply](http://readwrite.com/2014/08/27/devops-puppet-labs-luke-kanies-q-and-a/).

#### Product Owner/Product Manager

This is the role that defines and guides the requirements of the application. It is also one of the roles that varies in responsibilities the most across products. At its core, this role is the “owner” of the software under development. In that respect, they help prioritize, plan, and deliver software that meets your requirements. Someone has to be “the final word” on what happens in high functioning teams like this. The amount of control vs. consensus driven management is the main point of variability in this role, plus the topic areas that the product owner must be knowledge of.

It’s best to approach the product owner as a “breadth first role”: they have to understand the business, the customer, and the technical capabilities. This broad knowledge helps them make sure they’re making the right prioritization decisions.

In Pivotal Labs engagements, this role is often performed by a Pivotal employee, pairing up with one of your staff to train and transfer knowledge. Whether during a project or through workshops, they help you understand the Pivotal, iterative development approach, as well as mentor and train your internal staff to help them learn agile methods and skills, which will enable them to move on with confidence when the engagement is complete.

#### Designer

One of the major lessons of contemporary software is that design matters, a tremendous amount more than previously believed. The [“small batch” mentality of learning and improving software](https://blog.pivotal.io/pivotal-cloud-foundry/features/dear-developers-small-batch-releases-are-your-friend) afforded by cloud platforms like Pivotal Cloud Foundry gives you the ability to design more rapidly and with more data-driven precision than ever before. Hence, the role of a designer is core to cloud native teams.

The designer focuses on identifying the feature set for the application and translating that to a user experience for the development team. Activities may include completing the information architecture, user flows, wireframes, visual design, and high-fidelity mock-ups and style guides. Most importantly, designers have to “get out of the building” and not only see what actual users are doing with the software, but get to know those users and their needs intimately.

#### Testers (partial/optional)

While the product manager, and overall team are charged with testing their software, some organizations either want or need additional testing. Often this is “exploratory testing” where a third party (the tester[s]) are trying to systematically find the edge cases and other “bugs” the development team didn’t think of.

It’s worth questioning the need for separate testers if you find yourself in that situation to make sure you need them. Much routine “QA” is now automated (and can, thus, be done by the team and automated [CI/CD pipelines](https://blog.pivotal.io/pivotal-cloud-foundry/p-o-v/continuous-delivery-among-the-donkeys)), but you may want exploratory, manual testing in addition to what the team is already doing and verification that the software does as promised and functions under acceptable duress. But even that can be automated in some situations as the [Chaos Monkey](http://techblog.netflix.com/2012/07/chaos-monkey-released-into-wild.html) and [Chaos Lemur](https://blog.pivotal.io/pivotal-cloud-foundry/products/chaos-lemur-testing-high-availability-on-pivotal-cloud-foundry) show.

#### Architect (partial/optional)

Traditionally, this role is responsible for conducting enterprise analysis, design, planning, and implementation, using a “big picture” approach to ensure a successful development and execution of strategy. Those goals can still exist in many large organizations, but the role of an architect is evolving to be an enabler for more self-sufficient, and decoupled teams. Too often this role has become a “Dr. No” in most large organizations, so care must be taken to ensure that [the architect supports the team](https://en.wikipedia.org/wiki/Servant_leadership), not the other way around.

Architects are typically more senior technical staff who are “domain experts.” They may also be more technically astute and in a consultative way help ensure the long-term quality and flexibility of the software that the team creates, share best practices with teams, and otherwise enables the teams to be successful. As such, this role may be a fully dedicated one who, hopefully, still spends much of their time coding so as not to “go soft” and lose not only the trust of developers but an intimate enough knowledge of technology to know what’s possible and not possible in contemporary software.

#### Data Science (partial/optional)

If your application includes a large amount of data analysis, you should consider including a data scientist role on the team. This role can follow the dedicated/designated pattern as discussed with the operations role above.

Data Science today is where design might have been a few years ago. It is not considered to be a primary role within a product team, but more and more products today are introducing a level of insight not seen before. Google Now surfaces contextual information; SwiftKey offers word predictions based on swipe patterns; Netflix offers recommendations based on what other people are watching; and Uber offers predictive arrival times of their drivers. These features help turn transactional products into smart product.

#### Other Roles

There are many other roles that can and do exist in IT organizations. These are roles like database administrators (DBAs), security operations, network operations, or storage operations. In general, as with any “tool,” you should use what you need when you need it. However, as with the architect role above, any role must reorient itself to enabling the core teams rather than “governing” them. As the DevOps community has discussed at length for nearly ten years, the more you divide up your staffing by function, the further you move from a small, integrated team, and your goal of consistently and regularly building quality software will become harder.

### Agile Operations Roles

Roles here focus on operating, supporting, and extending the cloud platform in use. These roles typically sit “under” the Agile and DevOps teams, so the discussion here is briefer. Each role is described in term of roles and responsibilities typically encountered in Pivotal Cloud Foundry installs. These can vary by organization and deployment (public vs. private cloud, the need for multi-cloud support, types of IaaS used, etc.)

For a brief definition of a cloud platform, see [“the use of a cloud platform”](https://docs.google.com/document/d/1IqGTdBKCtPgcYdVuDSjT5dbHJ49LMI7Wiq1l5A7y1b8/edit#heading=h.74mcs4ji75vw)section below.

#### Application Operator

These are typically the “operations” people described above and serve as a supporting and oversight function to the business capabilities teams, whether designated or dedicated. Typical responsibilities are:

* Manages lifecycle and release management processes for apps running in Pivotal Cloud Foundry.
* Responsible for the continuous delivery process to build, deploy, and promote Pivotal Cloud Foundry applications.
* Ensures apps have automated functional tests that are used by the continuous delivery process to determine successful deployment and operation of applications.
* Ensures monitoring of applications is configured and have rules / alerts for routine and exceptional application conditions.
* Acts as second level support for applications, triaging issues, and disseminating them to the platform operator, platform developer or application developer as required.

A highly related, sometimes overlapping, role is that of centralized development tool providers. This role creates, sources, and manages the tools used by developers all the way from commonly used libraries to, version control and project management tools, to maintaining custom written frameworks. Companies like Netflix maintain “tools teams” like this, often open sourcing projects and practices they develop.

#### Platform Operator

The is the typical “sys admin” for the cloud platform itself:

* Manages IaaS infrastructure that Pivotal Cloud Foundry is deployed to, or co-ordinates with the team that does.
* Installs and configures Pivotal Cloud Foundry.
* Performs capacity, availability, issue, and change management processes for Pivotal Cloud Foundry.
* Scales Pivotal Cloud Foundry, forecasting, adding, and removing IaaS and physical capacity as required.
* Upgrades Pivotal Cloud Foundry.
* Ensures management and monitoring tools are integrated with Pivotal Cloud Foundry and have rules / alerts for routine and exceptional operations conditions.

#### Platform Engineering

This team and its roles are responsible for extending the capabilities of the cloud platform in use. What this role does per organization can vary, but common tasks of this role for organizations using Pivotal Cloud Foundry are to:

* Makes enhancements to existing [buildpack(s)](https://docs.cloudfoundry.org/buildpacks/custom.html" \t "_blank) and builds new buildpack(s) for the platform. Builds service broker(s) to manage lifecycle of external resources and make them available to Pivotal Cloud Foundry apps.
* Build Pivotal Cloud Foundry tiles with associated [BOSH](https://bosh.io/) releases and service brokers to enable managed services in Pivotal Cloud Foundry.
* Manages release and promotion process for buildpacks, service brokers, and tiles across Pivotal Cloud Foundry deployment topology.
* Integrates Pivotal Cloud Foundry APIs with external tool(s) when required.

#### Physical Infrastructure Operations

While not commonly covered in this type of discussion, someone has to maintain the hardware and data centers. In a cloud native organization this function is typically so highly abstracted and automated — if not outsourced to a service provider or public cloud altogether — that it it does not often play a major role on cloud native operations. However, especially at first as your organization is transforming this new way of operating, you will need to work with physical infrastructure operations staff, whether in-house or with your outsourcer.

**The DevOps evangelist**

This person must promote the benefits of DevOps by identifying and quantifying the business [benefits that come from the greater agility DevOps delivers](http://techbeacon.com/devops-4-benefits-using-agile-methods). As a change agent, the DevOps evangelist ensures buy-in from development and operational teams, identifies the key roles to support DevOps delivery methods, and makes sure IT professionals are trained and empowered to make those changes.

### ****The release manager****

Release managers work to address the management and coordination of the product from development through production. Release managers oversee the coordination, integration, and flow of development, testing, and deployment to support continuous delivery. They're focused not just on creating, but also maintaining the end-to-end application delivery tool chain.

### ****The automation architect****

automation architects analyze, design, and implement strategies for continuous deployments while ensuring high availability on production and pre-production systems.

The automation architect role becomes critical in the DevOps world. "DevOps organizations must provide an extremely reliable environment that is fully automated and free from obstacles," says Mizrachi. "With waterfall, everybody had to have 4x4 cars to drive off-road on tough terrain. The DevOps automation role is tasked with building the highway so the rest of us can use fast cars."

### ****The software developer/tester****

The software developer is at the heart of the DevOps organization. Under DevOps the title of software developer may remain the same, but the new role of software developer/tester dramatically increases the scope of responsibilities. The developers are responsible not only for turning new requirements into code, but unit testing, deployment, and ongoing monitoring as well. "They don't just build code to a spec and throw [it] over the wall to the QA team," .

This shift often requires a move to more automated testing so that quality doesn't suffer. "The problem is that teams still think they can do manual testing and still be agile. You can't," says Mizrachi. "If you need to test new builds every day and you're doing manual tests that last two weeks, you're in an impossible situation. Product quality usually degrades until the organization and processes change accordingly."

### ****The experience assurance (XA) professional****

While the [quality assurance (QA)](http://techbeacon.com/10-best-practices-qa-teams-deliver-quality-software-fast) function is often part of software development, a new type of control becomes necessary when organizations embrace DevOps. The need for QA testers is replaced by a need for XA experts charged with ensuring that all new features and functions are released with the end user experience in mind. "The current expectation for QA roles is to test functionality, but the role needs to evolve to include user experience testing,"

### ****The security engineer****

DevOps-minded shops have security engineers working side by side with developers, embedding their recommendations much earlier on in the process. "They build security into the product, not at the end,"

### ****The utility technology player****

The fast-paced DevOps environment requires a new breed. "Those operations or admin experts are now getting involved throughout the development process," Warren says. "Today, it wouldn't be uncommon for these operations experts—sometimes called DevOps engineers—to be involved in sprint planning to ensure that improved quality of service, resource management or security are prioritized alongside those delivered from the [business]."

Whether their traditional background is IT development or operations, "DevOps requires utility team members who can operate effectively across development platforms, tools, networks, servers and databases, and even across development and support," says Leitao.

References:

<https://techbeacon.com/7-devops-roles-you-need-succeed>

.