# Sourcing for DevOps
## Practical considerations for setting up DevOps with service providers

The need to bring new product features faster to their customers is forcing organizations to adopt new operating models. With information technology being a key enabler for most product innovation, organizations want to adopt DevOps as their way of working. But in doing so, they face challenges with regard to the readiness, the set-up and the team composition needed to achieve this. In this article, we provide our thoughts on how service providers can play a major role in the realization or improvement of a DevOps enabled organization. We discuss what an organization must know about the generic DevOps enabling capabilities of service providers, how to select the right service providers, what to bear in mind when contracting and how to compose and run the DevOps team.

**R. Keur**
is a senior manager, IT Sourcing department in Finance & Business Services at KPMG.
keur.rutger@kpmg.nl

**O. Rodrigues BE**
is a manager, IT Sourcing department in Finance & Business Services at KPMG.
rodrigues.oscar@kpmg.nl

**Y. Heymans MSc**
is a consultant, IT Sourcing department in Finance & Business Services at KPMG.
heymans.yuma@kpmg.nl

## INTRODUCTION

It is inevitable that organizations have to accelerate the time to market of their products and significantly reduce the time needed to incorporate user feedback. Faced with this reality, they are considering adopting DevOps as a means of bringing products faster to the market. DevOps builds on a collaborative organizational environment between business, IT development and IT operations, in order to deliver IT services in a short timeframe through the adoption of agile and lean practices. To achieve this, the organization needs to consider how DevOps will be implemented, bearing in mind the current internal and external IT landscape. External IT service providers, with their capability and experience in similar engagements, can play a defining role in enabling organizations to realize their DevOps ambitions. The challenge is to understand how service providers enable DevOps, and how this will result in a successful end to end implementation. This article will elaborate on why and how to source for DevOps.
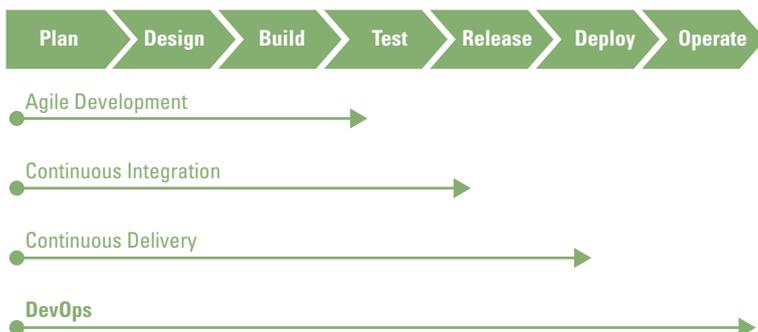
## ADOPTION OF DEVOPS

The traditional software development method was too rigid in its ability to accommodate changing business requirements and to accommodate the quick release of software. This resulted in issues such as misaligned requirements, long lead times and software that delivered features that were not fit for purpose. Agile software development partly provided answers by enabling faster and better delivery of software product features. But this was only applied to the development process, and it created a backlog for operations as product builds could not be tested and released fast enough. Parallel to these developments, there was of course a rapid evolution of tooling to facilitate continuous delivery – thus facilitating continuous integration, automated testing and rapid deployments. All these developments contribute to a great extent to the adoption of DevOps in an organization.

> The challenge is to understand how service providers enable DevOps and how this will result in a successful end to end implementation

**Why do organizations want to use DevOps methods?**

DevOps typically benefits organizations because it:
- promotes faster building, testing and deployment of product features;
- brings together business, development and operations, thus breaking down silos;
- makes faster defect detection possible, which leads to a more reliable product;
- results in better innovation and faster recovery times due to collaboration between the various teams.

KPMG recognizes the following success factors for DevOps:
- involvement of a product owner to set the product direction and lead the team;
- multifunctional teams with the mindset to collaborate;
- infrastructure as code – where operations infrastructure is managed using the same guidelines as used for software development;
- availability of enablement services (this includes tooling for the provisioning of Cloud environments, continuous testing, integration and deployments, monitoring of environments, etc.);
- DevOps is applied to applications in the IT landscape that provide an organization with the capability to differentiate.

**Figure 1.** Definition of DevOps: from plan to operate.



| Plan | Design | Build | Test | Release | Deploy | Operate |

Agile Development

Continuous Integration

Continuous Delivery

**DevOps**

## TRENDS IN SOURCING THAT ENHANCE THE ADOPTION OF DEVOPS

When looking at various sourcing drivers of traditional IT, we can identify an overall shift from a cost focus to a value focus. Cost is still relevant, but this is not the key goal anymore. Businesses are looking for an IT environment that is able to quickly adapt to business changes. Businesses want access to innovation that will help them overcome existential threats posed by changing technology. Trends in IT landscapes, service models and ways of working have made an impact on how easily organizations can work with DevOps.

> ### DevOps helps build a 'multi-functional' team – myth or reality?
>
> Business and IT have in the past operated from their own corners. IT has mostly been viewed as a support function. Issues with application functionality, performance and stability have always been 'IT' problems, no matter what the origin of the issue has been. The proliferation of Software as a Service (SaaS) solutions has not helped solving this problem either. Business teams can easily bypass the IT organization and contract SaaS solutions. IT is then burdened with the task of integrating the SaaS solution with other organizational systems and has to ensure interoperability. On the other hand, initiatives like DevOps are wrongly considered within organizations as an IT initiative, and are set up without a product owner from the business to guide product decisions.
>
> However, the relationship between business and IT is changing. Every organization depends on software for business innovation, and IT is no longer a support function but a business enabler. With this precedent, DevOps plays a key role in bringing business and IT together. Business should always be a part of the DevOps teams and should play its part in initiating DevOps. This provides the IT team a direct relationship to the product owner and thus to faster decision making around the product. Business can bring new product features to their customers and bring back customer feedback to the DevOps teams quicker. The IT part of the team can focus on incorporating this feedback and enhancing the product to the satisfaction of the business. A successful DevOps team is multi-functional and united by a single objective.

The changing sourcing drivers and emerging technology trends have made it necessary for most organizations to nurture and grow their DevOps ambitions. The next few paragraphs describe the implications of these trends.
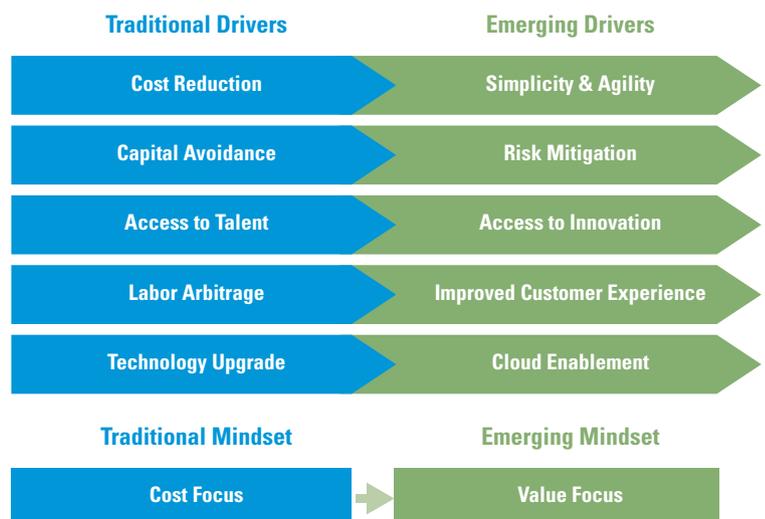
### Software as a Service versus DevOps

A number of SaaS solutions are available for business support areas like HR, CRM, Collaboration, Finance, ERP and IT/Facility services and organizations are subscribing to them. But this is not the case for an organization's core applications (systems of differentiation). This is where KPMG sees a shift from the traditional development methods (V model etc.) to a DevOps methodology. What also becomes apparent is the need for integrating SaaS solutions into the core applications.

### (Cloud) Enablement

There is a still growing availability of IaaS/PaaS solutions and public Cloud in multiple regions/countries. In addition, the more traditional outsourcing service providers are shifting towards a hybrid Cloud environment, providing orchestration and integration between the on-premise private Cloud and the various public Clouds used by customers. The Anything as a Service (XaaS) solutions are maturing from the initial hype to actual offerings that play an essential part in the IT portfolio of a wide range of companies. The ability to manage such a hybrid environment and to move workloads between the various solutions is a key success factor for efficient and effective IT services. The availability of highly automated environments facilitates DevOps teams to move workloads and achieve rapid deployment of changes.

**Figure 2.** The shifting drivers in sourcing require a different mindset.



| Traditional Drivers | Emerging Drivers |
|---|---|
| Cost Reduction | Simplicity & Agility |
| Capital Avoidance | Risk Mitigation |
| Access to Talent | Access to Innovation |
| Labor Arbitrage | Improved Customer Experience |
| Technology Upgrade | Cloud Enablement |

| Traditional Mindset | Emerging Mindset |
|---|---|
| Cost Focus | Value Focus |

### Automation

Initially, the cost savings that were associated with outsourcing were realized by standardized processes, economy of scale and labor arbitrage. While these are still relevant, the next level of efficiency is gained by increased automation leveraging Artificial Intelligence (AI) and Robotic Process Automation (RPA). Major investments in this field are delivering commercially sustainable solutions, potentially also resulting in insourcing activities that cannot be robotized. Automation is an important part of a successful DevOps setup; it enables the rapid change required by the organization.

### Multi-service provider

As part of the continuous drive to increase efficiency, service providers are focusing on the processes where they can differentiate, while at the same time leveraging best of breed solutions and outsourcing non-core activities themselves. Multi-service provider solutions are therefore the standard. The governance of any organization implementing DevOps should be able to integrate the various components of a service, with each component possibly delivered by different DevOps teams from different service providers. When sourcing DevOps capabilities, teams can be staffed by multiple service providers. If this setup is used, it is important to understand the obligations of the service providers in the team. Multi-service provider governance also requires a conscious decision about which services to contract directly, and where to leverage the service integration capabilities of a prime contractor.

### Infrastructure as Code & Continuous Testing

With true Infrastructure as Code (IaC) becoming widely implemented, operations infrastructure is managed using the same rules as used for software development – probably the single most important factor that enables Cloudification. Service providers have integrated their automation platforms with continuous delivery tool sets, and have developed tooling for the provisioning of Cloud infrastructure, and have experience on implementing continuous change propagation and orchestration. The provisioning layer gets easier to work with from a user perspective, orchestration tools become more mature, and automated testing is part of the chain. All testing (operations and development) is automated. As DevOps initiatives grow, continuous testing can become a constraint in the process; DevOps frameworks and toolsets evolving without a plan and covering the DevOps lifecycle can easily result in a low level of integration and automation. Planning a good framework is a must to avoid clutter.

## CHALLENGES ORGANIZATIONS FACE IN SOURCING FOR DEVOPS

In order to increase the speed, efficiency and quality of software delivery, organizations are looking for new ways of working that match their ambitions and taking advantage of evolving trends and new capabilities in the market. DevOps can be the model that enables the organization to do this successfully, by removing the traditional silos between the development and operation teams. The majority of organizations do not have the in-house skills to setup and maintain such a DevOps model, and look into alternative ways of sourcing this. Organizations in all industries face common challenges when they are realizing a DevOps way of working in their organization by using service providers:
- What is the organization's readiness to implement DevOps?
- How should a DevOps team be composed?
- What to look for in a prospective service provider that can help enable DevOps?
- How to contract DevOps enabling services?
- How is DevOps incorporated into a multi-service operating model?

The following sections will provide approaches to these challenges and will help envision the organization's future state of DevOps.

## SOURCING FOR DEVOPS

We see the future DevOps organization as one with different delivery models matching the organizations time to market requirement for various IT products or services. Organizations will be using DevOps and traditional waterfall models, which will be coexisting and supporting different functions and speeds, next to the utilization

> The majority of organizations do not have the in-house skills to setup and maintain such a DevOps model

of various SaaS and Business Process as a Service (BPaaS) solutions. While an online retail platform demands an agile delivery model to bring product assortment and promotions faster to the market, a product management system does not need to be enhanced at the same rate.

Within the (hybrid) delivery model, responsibilities have to be divided between the service provider and the own organization. Based on the maturity of the organization in DevOps, a growth model can be sourced where the level of maturity of a primarily internal skillset can be increased by utilizing the capabilities of the (more mature) service provider (see Figure 3).

With Figure 3 as a starting point, the organization can start assessing the service provider landscape and engage the market. The phases in Figure 4 must be considered as part of the DevOps sourcing approach.

### 1. Define Target Operating Model (TOM) and assess internal organization

In the TOM, the organization must decide how people will interact with each other and with processes and technology aspects. An important consideration in the way DevOps teams will operate in the future is if the team will be directed by the customer or by the service provider. In a customer led team, the majority of the team is from the customer organization and the customer decides on capabilities, processes and technology. In a service provider led team, the service provider decides on these aspects.

In addition, since the DevOps skillset is different from the traditional one, the organization needs to have a good insight into which skills are available in-house and how to get the roles filled. Where the business representative needs to be mandatory in the team as an in-house resource, the other roles like release manager and software developer can be sourced differently. In doing so, the following roles are of value:

- *Business owner*: although not different when compared to non-DevOps development methods, this role needs to be filled within the DevOps team, because this person is the sponsor and is actively responsible for stakeholder management, being the face of the business and accountable for the outcome. The business owner guides the product owner by communicating the business requirements.
- *Product owner*: sets the product direction and decides on product differentiation. This role must remain in-house, but the product differentiation should be challenged internally to avoid the 'not invented here' syndrome. For technical products, the product owner could be someone from the technical team.
- *Enterprise architect*: the ability to set the technological direction (making technological choices that aid business decisions) must remain in-house. This typically means that the enterprise architects remain in-house. The architects also contribute to keeping tabs on the tooling used to enable continuous delivery in order to avoid vendor lock-in.

**Figure 3.** The assignment of responsibilities depends on the DevOps maturity of the organization.

| | | Low | | High | |
| --- | --- | --- | --- | --- | --- |
| | | Client | Service Provider | Client | Service Provider |
| **Technical Application Management** | Application Configuration | ✔ | | ✔ | ✔ |
| | Deploy Code | ✔ | | ✔ | ✔ |
| | Appserver/Webserver Management | ✔ | | | ✔ |
| **Platform as a Service** | Provision Database | ✔ | ✔ | | ✔ |
| | Provision Middleware | ✔ | ✔ | | ✔ |
| | Provision Operating System | ✔ | ✔ | | ✔ |
| **Infrastructure as a Service** | Configure Virtual Machine/Network | ✔ | ✔ | | ✔ |
| | Provision Network | | ✔ | | ✔ |
| | Provision Virtual Machines | | ✔ | | ✔ |

(DevOps Maturity)

**Figure 4.** Phases in sourcing for DevOps.

| Assess and define internal organization | Assess DevOps enabling capabilities of service providers | Select DevOps enabling service providers | Contract DevOps enabling services | Compose a successful DevOps team |
| --- | --- | --- | --- | --- |
| **Execute the following:** | **Assess which service providers have the right:** | **The following should be taken into account:** | **Agree upon the following and make DevOps specific:** | **Execute the following:** |
| • Assess in-house skills<br>• Define the Target Operating Model (TOM) | • Mindset<br>• Tooling<br>• Experience | • Company need and required scope<br>• Integration<br>• Core technology<br>• Commercial proposal | • Legal framework<br>• Way of defining work packages<br>• Pricing<br>• KPIs and monitoring | • Fulfill missing roles<br>• Integrate other teams and enablement functions<br>• Choose location |

## 2. Assess generic DevOps enabling capabilities of service providers

DevOps service providers must be fluent with continuous delivery tooling, collaboration and incorporating quick user feedback into their development mechanisms. In addition, enablement must be sufficiently taken into account (Hybrid Cloud, automated testing, integration, security). A service provider can provide organizations with a jump start in implementing DevOps. But to achieve this, a thorough service provider assessment is required to understand to what extent the service provider brings the right mindset, tooling and experience.

### Mindset
Contrary to popular belief, DevOps requires a structured way of working. This is needed in order to balance the business' requirement for speed with the development engineer's requirement to test and roll out releases. The service provider must bring the steadying influence by providing the required structured way of working and communicating the DevOps principles.

### Tooling
We recommend that the service providers demonstrate capabilities on enablement services for continuous delivery, monitoring and operations. Service providers can provide access to the in-house environment to comply with the organization's security requirements where needed, but also to a secure environment in the Cloud or a hybrid set-up for the organization's diverse IT landscape. Service providers should have developed tooling frameworks for the provisioning of Cloud infrastructure, using best of breed solutions and easily changeable components in the framework. In addition, they should have experience of implementing continuous change propagation and orchestration.

## A service provider can provide organizations with a jump start in implementing DevOps

### Experience
The right mindset and mastery of the tooling can be no substitute for actually having implemented the DevOps way of working. The service provider must have experience with collaborating within DevOps teams while taking responsibilities for project outcomes. The DevOps organization should assess the service provider's experience by examining the skills and knowledge of the people (employee profiles) and relevant DevOps references from previous clients.

## 3. Select DevOps enabling service providers

When selecting the right DevOps enabling service provider, the following must be taken into account:
- *Required skillset*: as part of the service provider assessment described in the previous section, the service provider needs to be able to provide the right skill sets and have these readily available. References of successful implementations and getting to know the team are important selection criteria, along with the clear requirements around the required skills.
- *Responsibilities and performance measurement of the service provider*: explore the manner in which responsibility is set in the DevOps team and assure that the performance of the service provider is measured in a meaningful way.
- *Company need and required scope*: depending on the scope, service providers can help in implementing the operating model and/or facilitating organizational change management to achieve a (more successful) DevOps way of working. An example is a team member from the service provider coaching the organization's employees in adopting the DevOps way of working.
- *Integration*: the enablement capability can be sourced in different ways: small cells (not linking up), a subset of applications or a hybrid environment enabling both the traditional IT and DevOps part. In addition, the integrated services should adhere to security standards, while maintaining flexible models in both enablement and in development teams.
- *Core technology*: fast changing technological knowledge is something that the organization should not concentrate on, but source it. For instance, the knowledge and skills regarding to infrastructure components that enables business applications can be sourced from a large set of service providers.
- *Commercial proposal*: in the end the commercial offering needs to fit the company budget. In the selection phase the commercial aspects need attention to see if the business case can be made.

These aspects need to be translated into requirements and should be communicated with the potential service providers during the selection phase. The service provider which can best meet these requirements will be selected.

## 4. Contract DevOps

As part of the vendor selection to select the right service provider, the contracting phase is also initiated.

Firstly, the organization should create a legal framework like a Master Service Agreement (MSA).

Secondly, a clear way of defining work packages must be agreed upon. For instance, it should be clear whether work packages are translated into a Minimal Viable Products (MVP's) or Definition of Done (DoD). While doing this, it is important to maintain flexibility in the possibilities within contractual agreements.

Thirdly, pricing must be agreed upon. The various elements are provided in a broad range of pricing models, from time- and material-based consulting to an End-to-End DevOps service, that is typically priced per application or business unit, including change management skills and managed operations per application or environment.

Finally, to monitor the service provider's performance, DevOps specific Key Performance Indicators (KPIs) must be implemented. The KPIs in Table 1, with their corresponding measurement methods, can be incorporated into DevOps contracts to assure quality in a controlled manner.

## 5. Compose a successful DevOps team

A DevOps team, while composed of people taken from the silos of development and operations teams, must be united around one single objective – that could be a business product, a technical integration platform or a common services platform. The team is cross functional and self-sufficient. The organization must consider how the aspects mentioned below are achieved.

### Roles

The team typically consists of different roles, such as the product owner, business analysts and consultants, a solution architect, developers, security analysts, testers and operations experts. This makes collaboration within the team very important. Collaboration helps different team members understand each other's point of view, so that they can make informed product related decisions affecting all aspects of the product lifecycle. Collaboration enables knowledge sharing within the team and leads to a sharing of responsibilities.

### Integration

While collaboration within the team is important, collaboration outside the team is also important. The IT landscape seldom consists of simple, stand-alone products. More often than not, it is a complex integration of functional applications and technical components that deliver the final functionality to a larger product. When focusing on a single objective within a DevOps team, it

**Table 1.** Setting KPI's for managing performance in a DevOps way of working.

| KPI | Description | Measured by |
|-----|-------------|-------------|
| **Business Performance** | the extent to which services add value to achieving business goals | earnings, market share, share value, new customers, new markets, strategic objectives or brand reputation |
| **Service Quality** | a relative measure of the quality of work delivered during a sprint | Story Points, deployment success rate, uptime, Customer Perceived Value (CPV) and/or Net Promoter Score (NPS) |
| **Velocity** | a relative measure of the amount of work delivered during a sprint | Story Points, cycle time and/or frequency of deployment |
| **Process Efficiency** | shows how efficiently the team members spend their time working on committed stories | Story Points, cost per release, cost per transaction, mean-time to resolution and/or response time |
| **Sprint Burndown** | shows how much work the team has left versus the ideal amount of work remaining | Story Points and/or hours |
| **Release Burnup** | shows how much a team has left to complete a release versus the ideal amount of work remaining | Story Points and/or hours (the remaining amount of Story Points is updated every day) |
| **Defect Count** | the number of defects in the software | number of identified bugs in the Backlog |
| **Severity of Faults** | the severity of the fixed defects | three different categories: minor faults, major faults and critical faults (the Scrum team itself can categorize these faults). |
| **Recovery Rate** | the time needed to recover an instance of systems outage | earnings, market share, share value, new customers, new markets, strategic objectives or brand reputation |

is fairly easy to lose sight of the larger picture and start operating in silos again. Thus, collaborating with other DevOps teams which have common interfaces is critical to success. The overall integration of services, process and technology is a key success factor for a successful DevOps implementation. All three need to be setup and maintained properly. Only focusing on the small team can lead to duplication of work, tools and processes, with the associated costs and lead times accompanying this duplication. The DevOps team needs to link up to the other teams and enablement functions, to assure their choice of technology, tools and processes are right for the overall organization.

### Location

With a focus on agility and collaboration, DevOps teams have a need to work closely together. This often translates to the team being co-located. But this is not always the case. DevOps teams can comprise of in-house or service provider resources operating from different locations. Teams involved in delivering business functionality and requiring a higher degree of collaboration are best

co-located. This also ensures quick decision making. Teams that handle data with security restrictions (like source protection) should be located within the same country as where the data is housed and governed by local laws. Teams that are not co-located must pay additional attention to aspects like building trust, deciding on the timing of work that ensures that time is available for interaction between team members and a means of shortening the turnaround times.

## ASSURING DEVOPS READINESS

When approaching the market to realize or improve a DevOps enabled organization, the following prerequisites must be in place:

### 1. Create a selected set of applications
Analyze the IT landscape to identify which applications qualify for a DevOps approach. The approach of looking at the application landscape from a Systems of Record, Systems of Differentiation and Systems of Innovation perspective provides a good starting point for qualifying applications for DevOps. Verify that the selected applications are aligned with the business strategy. Begin small by implementing this for a selected set of applications and then scale the approach to other applications. This will provide the organization with the opportunity to gather learning, finetune ways of working, calibrate tooling and demonstrate the successes to the rest of the organization to encourage wider adoption. Based on this, regularly update the service model for each application and domain, should it be BPaaS, SaaS, Commercially of the Shelf (COTS) or Build Your Own (BYO). Where BPaaS and SaaS have application development as part of the service, both COTS and BYO have an Application Development and Maintenance (ADM) approach, which will differ for each application.

### 2. Create an enablement layer to facilitate DevOps
To be able to facilitate the full DevOps cycle, an automated Cloud environment and testing needs to be implemented to achieve the rapid deployment of changes to the application and realize the associated testing. This can either be a small public Cloud setup, or leveraging on a bigger Hybrid Cloud environment for multiple applications, depending on the need from the business. Service integration and this typical setup is a service all larger IT service providers provide as part of their portfolio.

### 3. Design an integration and orchestration layer
In order to maintain a secure and integrated environment, the IT Enterprise architectural design needs to be created to assure security and compliancy requirements are satisfied, and to keep these flexible. In this way, the environment can be monitored on cost and availability.

Elements like identity and access management, interconnect ability, orchestration and integration processes should be taken into account.

With the mentioned approach and prerequisites, the organization can be confident to realize their sourcing for DevOps ambitions.

## RECAP AND ADVICE

It is only a question of time before organizations will make DevOps a part of their working culture. The degree to which it is implemented will vary. It is important to realize that DevOps may not be applicable to the entire IT landscape. Organizations should start small by identifying a limited set of applications, that will be developed and maintained with DevOps. The actual implementation can greatly benefit through the involvement of service providers, who can bring in their knowledge and experience to support organizations in this journey. When sourcing for DevOps, the organization must first assess the generic DevOps enabling capabilities of the service provider. The service providers must then be assessed on their ability to satisfy requirements related to integration, core technology and the commercial proposal. When the service provider is selected, the right KPIs for managing the collaboration have to be agreed upon. Finally, the DevOps team is composed and run while taking the roles, team integration and location into consideration. Enablement services and tooling can and should assist the organization in this realization of a successful DevOps way of working, but the essence of the successful DevOps team is the involvement of all the right people, doing the right job, enabled by the right technology and incentives.

**About the authors**

**Rutger Keur** is a senior manager at KPMG Finance and Business Services, with extensive knowledge on IT sourcing, advising customers on their current strategy, designing the future IT sourcing and monitoring the specific elements of the customers IT sourcing. With over 15 years of experience in this area, Rutger has worked on various roles at customers as well as service providers, as part of customers vendor management and service provider sales teams.

**Oscar Rodrigues BE** is a manager at KPMG Finance and Business Services. Oscar has experience in various roles within IT demand and supply organizations. His experience extends to a solid foundation in the areas of application development, application management and project management.

**Yuma Heymans MSc** is a consultant at KPMG Finance and Business Services. He has experience in IT and business from a service provider and client perspective. He focuses on IT Sourcing Strategy development and organizational models.