White Paper

A Checklist for your

Cloud Migration Strategy

Analyze your technology stack before migrating into the cloud

By Flo Adam

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

Cloud computing has driven the highly dynamic provisioning of workloads across scalable and automated infrastructures. It has allowed applications to efficiently respond to customer-driven peaks in demand, increasing both speed-to-market and flexibility.

The cloud has enabled a transition to continuous, self-service delivery. Migration to the cloud is therefore an integral process for any organization seeking to become digital. An estimated 83% of enterprise workloads will inhabit the cloud by 2020, 63% of which will do so as part of a wider digital transformation.[1] Those who fail to innovate how they consume and process information risk becoming obsolete.

While the cloud offers tremendous business value, organizations migrating their workloads to the cloud must make them cloud native. If not, they risk locking themselves into infrastructures not ideally suited to their long-term interests. This can result in spiralling costs as they attempt to scale. An application in the cloud that is not auto-provisioning, auto-scaling, and auto-redundant will be unable to take full advantage of what the cloud has to offer.

Organizations migrating to the cloud and becoming cloud native are breaking legacy applications into components that can evolve in the cloud. These organizations are integrating cutting-edge technologies and practices into their products, processes, and strategies to find creative solutions to their industry's problems. They are embracing open source solutions and changing their cultural and operational processes as part of a wider DevOps transformation.

Organizations that fully take advantage of the cloud migrate with carefully planned strategies. They are not jumping into the cloud without carefully considering their business and technical requirements nor are they doing so without re-evaluating the way their infrastructures and applications are organized. They are focusing on their long-term roadmaps and developing comprehensive migration plans.

This white paper discusses how specific technologies can be assessed and migrated to cloud native equivalents. It is ideal for team leaders who are striving to modernize the solutions supporting their architecture. Read it for a technical understanding of cloud migration.

[1] Columbus, Louis (2018, January 7). *83% Of Enterprise Workloads Will Be In The Cloud By 2020*. Retrieved from forbes.com.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

Cloud Migration Strategies

To devise a proper strategy, you will need to analyze and document all of your requirements. Bring your entire technical team together and methodically work through each component, beginning with the bottom of the stack and moving upwards. Discuss each component both on its own as well as in relationship with others. Look for patterns in your application and architecture that can easily be transformed into cloud native patterns. In doing so, you may lose some functionalities and gain cloud native equivalents. Objectively review what kinds of platforms are viable to migrate towards and what their respective advantages would be.

Analyse your Application Environment's Toolings

1. Hardware Resources

Start off by documenting the amount of resources required for each application that has been identified for migration. Take note of what is currently being used and what will probably be required post-migration. Document the lowest amounts of CPU and RAM being used when the application is idle and at peak usage. These peak values should be a part of your system requirements, but make them maximum not hard values. Make sure that your hardware resources have assigned limits and parameters for consumption. As a general rule, your hardware resources will be eaten up more quickly than you expect. They will consume as much energy as they can, and need limits to work within. If you don't assign concrete limits, your maintenance will be much more expensive than necessary. Cloud environments are generally charged by the hour or the minute, and any undocumented hardware resources will add up fairly quickly. Cloud sprawl can be difficult to manage, so it is best to plan either for the retirement or repurposing of your current hardware and fully document the required resources.

You will have to stress test the APIs of your application to document the amount of resources needed. Use real payload or real messages to call your API and then monitor how many messages can be passed per second. Once that has been done for multiple calls on the APIs, it becomes possible to experiment with different requirements for the API and choose the one that takes the most space. Performance testing should always be a requirement before putting something into production. However, you may not need to do this if you have a performance team as they may already have data on the times it takes for API calls to respond. Likewise, the purchase orders for hardware may indicate how much RAM would be needed for different actions. Nonetheless, verify your resources.

After analysing your hardware resources, you can define the requirements for your future VM and container images. This will depend on how you have decided to distribute your workloads, either as VMs, containers, or on a serverless environment. Choose the operating system and the amount of CPU and RAM that will be needed. Go into as much detail as possible without deciding upon a specific cloud implementation.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

2. Operating System (OS)

Cloud infrastructures tend to use VMs, containers, or a combination thereof, on either Windows or Linux OS. Cloud native technologies do not usually support other OSes, so you may want to adapt if you are using something else. Keep in mind that different cloud providers contract support for different OSes. While it is possible to upload or configure any OS that you choose, a degree of built-in support is sometimes necessary. Likewise, your OS will affect your available choices for middleware, software, and licencing. Assessing your application holistically will therefore help you analyze the requirements for your OS. View the decision tree in figure 1.

Keep in mind that some applications were written specifically for the mainframe and consequently may not be easily moved away. Reinventing the wheel is not usually a good idea, and it may be wise to keep applications that don't have a cloud equivalent on the mainframe. For example, ATM machines should not be moved away from the mainframe. Mainframe applications that have been written with modern languages in mind may have an equivalent, cloud-friendly platform that your vendor may be able to help you with. Identify your mainframe applications and contact your mainframe vendor to see how much time and money would be involved in converting them.

3. Middleware

Detail an inventory of all the middleware that you are currently using, and discuss how each component could fit into a cloud native architecture. If you have decided on an IaaS, they may provide certain middleware that you are already using. In that case, you may decide to retire or repurpose those tools in favour of their cloud hosted equivalent. Any middleware to be retired will need to be accounted for in your current hardware retirement plan. Note that middleware is taking a less prominent role in cloud native architectures following the rise of the service mesh. Many are migrating their middleware to the cloud by declaring APIs and then having services call each other instead of using middleware as a third party. If you decide to use Envoy, Istio, or any service mesh solution, understand that you will need to significantly reevaluate the usage of your middleware as part of your cloud native strategy.

4. Software

Detail an inventory of the software being used to host your application as well as the tools used for monitoring, debugging, logging, and all other functions. Take note of your Quality Assurance suite of tools for building engines and automating functions. Be especially comprehensive in analysing and documenting your current application runtime as a baseline for your future requirements.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

5. Software Support and Licencing

An assessment of the amount of support that your application will require is essential in deciding whether to choose an open source solution or not. Unless provided by a third party at cost, such as Red Hat's OpenShift, open source tools and applications come with no support. If choosing an open source solution, be sure to review the level of support that you will need and define the relevant support requirements. If you are currently using an open source solution, there may be a supported equivalent that you could easily migrate towards. The closed source RedHat JBOSS provides support for the open source WildFly just as IBM Liberty provides support for OpenLiberty. Your cloud provider may have knowledge on how licences are billed and how to count the number of licences needed and used.

6. Licensing

Every tool that that will be a part of your migration strategy needs its licensing accounted for. Every instance needs a licence covering the product(s) that are being run. Although most cloud providers will bundle software licences together as part of the contract, you are responsible for ensuring that all applications being migrated to the cloud have their licences accounted for. Some of your current contracts may not be relevant because their licensing model predates the cloud. Others may offer cloud-specific software licenses that may come with their own complications but could help transfer the licensing into the cloud. It's important to examine the licensing models for all software before beginning your migration strategy. Keep in mind that licences tend to come with support, and the amount of support you need will influence the licences you choose to have.

7. Analyse the Applications

Analyse your applications and define a common strategy to apply across most or all of your applications. These decision trees demonstrate what you can do with the servers that your native and web applications are running.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

8. Native Applications

Any applications written in C, C++, C#, Perl, Python, or any similar language will need to be ported on Linux VMs, Windows VMs, or containers before being moved to the cloud. Likewise, any native application on a mainframe or on Unix will need to be adapted for the cloud. Keep in mind that Windows is not available as a container image, so you will have to use Linux OS for container-based solutions.

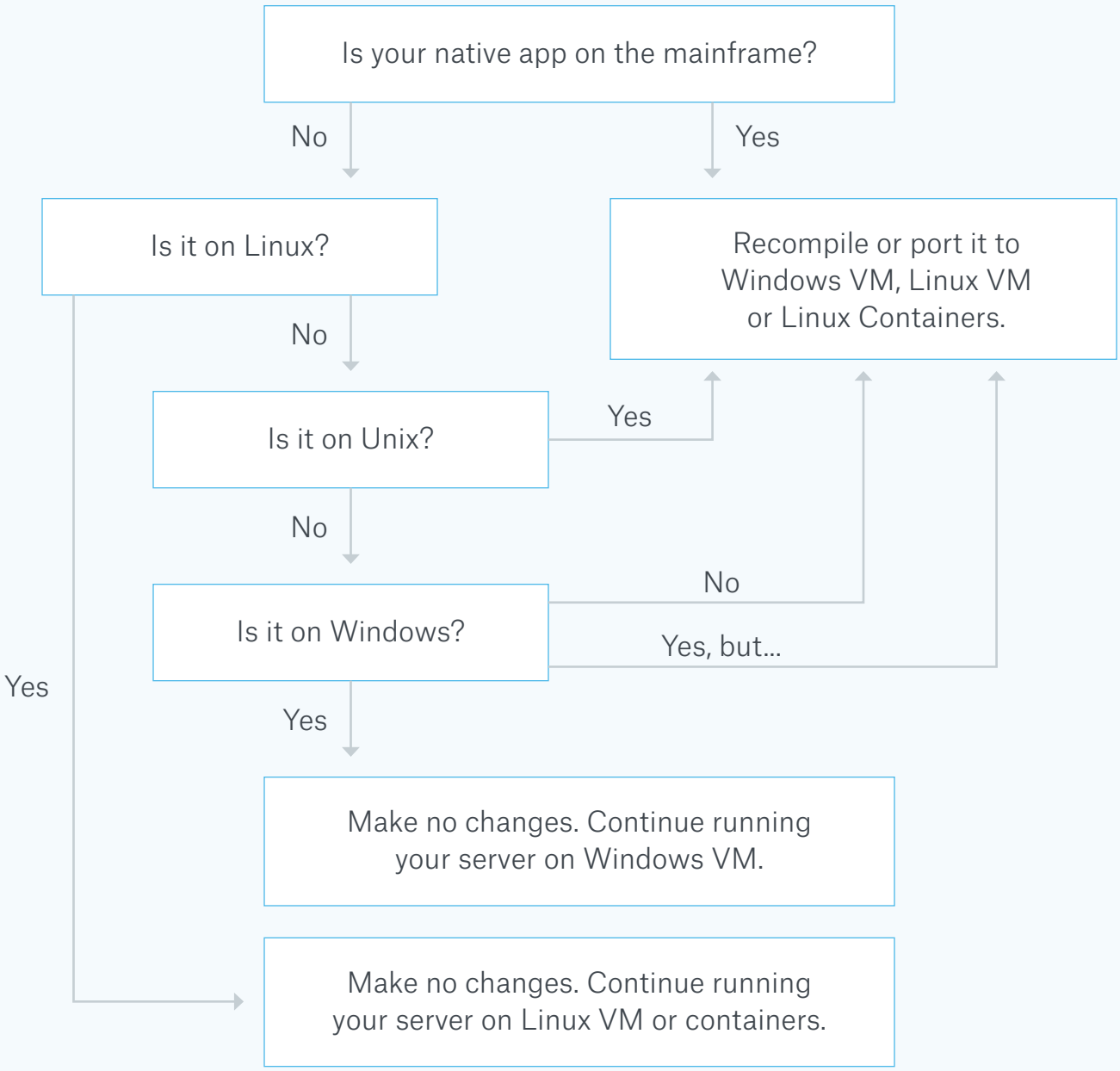

Figure 1

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

9. Web Applications

While preparing for a migration strategy, you will need to choose whether you want to use open source to host your web application or website as well as select a hosting tool. Your choices will depend on the technical requirements of your current environment. Document the server that your web applications are running on and verify that it is available in the target environment. Take note of the framework currently being used, whether it is JEE, Spring, or any other. Which dependencies are being used, and are they server specific? For Java-based applications, analyse the memory usage of each application to extract minimum and maximum limits for heap/stack. Regardless, you should choose to move to Windows or Linux.

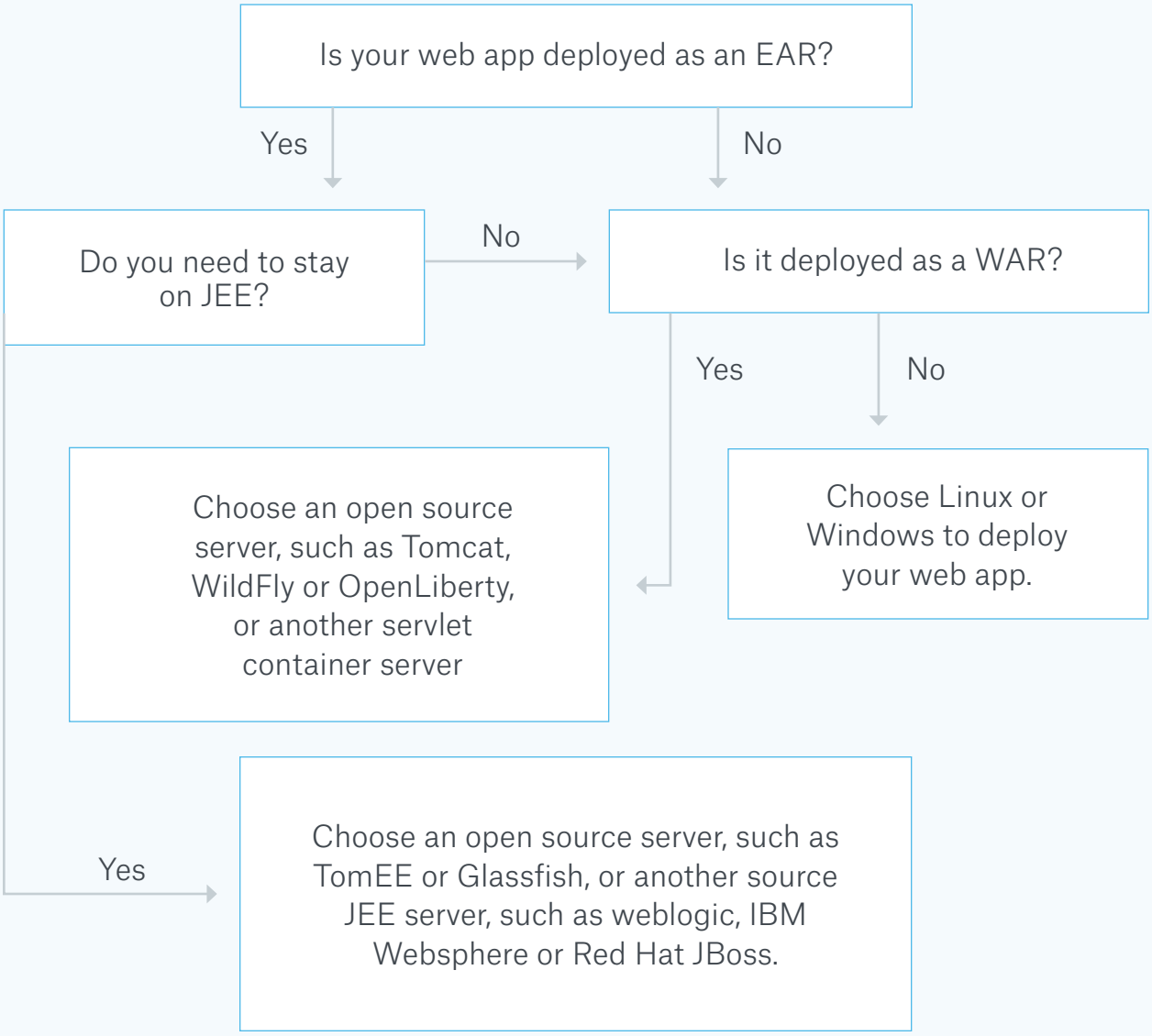

Figure 2

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

Analyse the Properties

Identify your application properties that point to external resources, such as databases, queue managers, ESBs, etc. Any external services that change depending on the application's environment must have their properties externalized. Ideally, VMs and containers should be suitable for many different environments, development, testing, and production without needing their images to be rebuilt. The application will need to retrieve its properties via its runtime environment. Look out for any red flags in your application properties that could cause problems down the road.

Development Environment

Categorize properties as a part of your DevOps framework. It's important to know the difference between your dev environment and every other environment between you and your production. This will allow you to understand the properties for the application being developed and those needed to be externalized. You can therefore reuse your container and allow the environment properties to work. To quote the 12-factor app, "keep development, staging, and production as similar as possible." Implement DevOps practices throughout your migration strategy. On both ends use similar tools, such as VM hypervisors, Docker registries, Docker runtimes, Kubernetes clusters, and OpenShift clusters. This consistency will facilitate engineering. Streamline your migration pipeline, and aim for consistency throughout. Be careful with the clean up to help avoid cloud sprawl.

Automation

A cloud native application auto-provisions, auto-scales, and is auto-resilient. There are many steps that should be automated in your migrated application. Define the build strategy for your Docker images; Jenkins can work well as a solution. Save dockerfile with your code or use it as a template in your build pipeline. Define the deployment strategy for your image repository. Define the build strategy for your config map - This will tell your application how your config maps should be specified in the source repo and imported after the image is deployed. In pure VM or docker-based environments, these would be the application's environment variables. Define the build strategy for secrets, including the naming convention for files and parameters for what is stored.

There are currently open source pipelines deployed for Jenkins that you could use to create docker images and deploy your application in the cloud. You could also use another pipeline where you use a VM image deployed as a template in the cloud. Automation should be kept top of mind throughout any cloud migration.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

How CloudOps can help

With a cloud native application and infrastructure, your organization will be able to easily respond to the peaks of business activity while increasing its flexibility and speed-to-market. Your technical teams will be able to seamlessly self-service and provision environments for development, staging, and production. The flexible infrastructure and interoperable, right-sized components will encourage experimentation and progression. Compiling an in-depth strategy will facilitate your move to the cloud and pave the way to future success.

Migration strategies can be obscured and held back by the number and complexity of decisions involved. CloudOps can help you understand your application landscape, technology stack, and appropriate cloud options to build a roadmap for your long-term success. Our seven-part transformation process will walk you through what's required to seamlessly migrate your application into the cloud, resulting in an efficient, automated, and scalable application designed to help you meet your business goals.

1. Discover

CloudOps will survey your existing application portfolio within its related business context and provide you with a detailed report on its current state.

2. Assess Toolings

We will identify the applications that are 'cloud-ready', that would be good candidates for migration. The technical requirements of each application will be carefully analyzed to determine its best path. This could mean refactoring, replatforming, repurchasing, retaining, retiring, or rehosting. Your business projection will affect the size and price of your migrated application, and we will relate your technical requirements to your business model throughout the assessment.

3. Assess Processes

We will analyze the operational practices and processes of your organization. As your migrated application will use different solutions, it will require adapting the methods used to operate toolings. We will assess the processes of your organization as part of a migration strategy.

4. Plan

We will create a migration plan that evolves your current infrastructure to a private, public, hybrid, or multi-cloud infrastructure, taking into consideration the impact and dependencies within your application environment, existing hardware, and business goals.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364

5. Execute

Our team of consultants will work with your internal IT organization to migrate existing applications from legacy environments to the cloud while minimizing disruption to your business.

6. Training

We will equip your technical teams with the necessary skills to build, migrate, or operate your application. Through a series of workshops, including Infrastructure as Code, containers, service meshes, secrets management, DevOps monitoring, and CI/CD, your team will learn about all the toolings that comprise your application as well as the organizational processes and practices required to successfully leverage their potential.

7. Support

CloudOps provides SOC 2 compliant managed services and augmented support. You can access our team of DevOps experts anytime to complement our workshops or to fully operate your application platform.

CloudOps' team can assist you in creating and executing a migration strategy that meets your current workload and security requirements and that can scale according to your future needs. This will likely involve a new way of working with your development and operations team and can begin with getting them up to speed on DevOps tools and processes with our hands-on **workshops**. Cloud agnostic but opinionated, CloudOps can migrate your infrastructure into any cloud: AWS, GCP, Azure, or cloud.ca for free*. **Click here** to get more information on how we can provide you with the right path and momentum to migrate your application into the cloud.

**Certain conditions apply.*

This white paper was written by Flo Adam, a Senior Cloud Developer at CloudOps with significant experience engineering enterprise grade middleware on Linux platforms.

info@cloudops.com
@cloudops_

200 - 423 rue Saint-Nicolas,
Montreal, QC H2Y 2P4
1-888-796-8364