Cloud Transitions done right.
We are Xpirit.

Xpirit

Cloud Transitions done right

It is no longer a question whether your organization will move applications to the cloud; it's only a matter of when and how it must be done. In this article, we will share our insights on what is required to make this transition successful. We will highlight various perspectives that should be taken into account when you consider a cloud migration and explain how you can determine the right strategy to follow.

Do you have a business case for migrating to the cloud?

When we talk with organizations about cloud transitions, we see that a lot of different approaches are taken. The reason for this is that the chosen approach and steps to take are highly dependent on the perspective of who in the organization is asked to lead the transition.

If you ask the developers how they can move their application to the cloud, they will come up with great plans on how to change their application's architecture to micro services and how they can use the latest .NET Core framework, since it has been optimized for cloud workloads. This will lead to a high investment before things can be moved, because the fundamental differences of this type of architecture require the application to be rewritten.

If you ask the IT operations department how to make the transition, they will come up with a new way to provision infrastructure and set up a service catalog from which customers can request new virtual machines that will now be provisioned in the cloud. Furthermore, you will see extensive network architectures and a lot of complexity, because they try to implement their current systems using cloud infrastructure, which is quite different when you want to get the maximum benefits from the cloud.

These are two examples of many other perspectives. Are they wrong? We don't think so. But we do think these approaches are suboptimal and will incur high costs and low return on investment. To prevent this, an answer should be given to the question: which migration strategy will contribute to your organization's business and IT goals? In other words: what is the business case for migrating an application to the cloud?

From CAPEX to OPEX

The cloud is a real game changer. Not only from a technical perspective but even more so from an economical perspective. In the past, an organization had to spend significant amounts of money to start a competitive online service. However, these capital expenses (CAPEX) are mostly gone, and all costs are moving to operational expenses (OPEX). This is because you don't have to invest in hardware, but instead you pay the cloud provider for the resources you use. This is the on-demand,

Pay-As-You-Go nature of the cloud. From this shift, we can see two forces that require our customers to change the way software is delivered. The first force concerns independent Software Vendors (ISV's) that are now asked to provide their Software-as-a-Service, because their customers want the same model for the software they buy as they now do with hardware in the cloud. The second force concerns enterprises that are driven to reduce their operational costs and one way to make this happen is by adopting the cloud. You see many enterprises state in their plans to totally move to the cloud and get rid of their own datacenters. This sounds very lucrative at first, but sometimes one tends to forget that just moving your existing machines to machines in the cloud is not at all economically beneficial. Your overall costs will probably become much higher.

...

» **CAPEX** = Capital Expenditures, investment costs for developing a system.

» **OPEX** = Operational Expenditures, the returning costs when using a system

How do you move to the cloud the right way?

The first thing to understand is that moving to the cloud is not a matter of one size fits all. For example, if you are the ISV as mentioned in the previous paragraph, you need to look at your current software and determine the cost involved if you are now hosting this software yourself. You are now confronted with the incurred costs your customers had. These costs should be replicated for every customer you have. You need to look at what is the state of the software and in what part of the lifecycle it is. Has it just been built, has it been out there for a long time and does it already need significant rework, or is it a product that is at the end of its lifecycle and you need a way to provide SaaS but you don't want to invest?

Depending on the lifecycle phase of your application, you can project it on a cloud migration strategy model such as the Gartner 5-R model (Rehost, Refactor, Revise, Rebuild, Replace), the Azure 5-R model or the AWS 6-R model. These "R" models state that you need to pick one of these strategies based on your company's cloud migration goals as well as the requirements and constraints of the specific application.

To demystify the options you can choose from, the strategies from the various "R" models are combined and explained in the following table.

Strategy	Description	Pros & Cons
Remove/Retire	Turn off applications from the portfolio that are no longer useful/not contributing to business goals.	**+** ○ Positive impact on business ca ○ Frees up time to spend on other applications
Retain/Revisit	Applications that have no priority for moving now. Leave as-is and evaluate again when most of the application portfolio has been shifted to the cloud.	**+** ○ Focus on applications that deliver most business value
Rehost	Lift-and-shift your solution from on-premise to IaaS. Practically this means moving your current (virtual) machines to virtual machines in the cloud. No changes are made to your code. Another way of rehosting is containerization of your application.	**+** ○ No changes to the code (minimal changes in case of containerization) **−** ○ No utilization of cloud benefits like scalability
Refactor/Replatform/Revise	Move to PaaS with minor adjustments, making relatively small changes to the existing application so it becomes more efficient in resource use, especially when you are serving multiple customers with the same software. This is better known as making an application multi-tenant.	**+** ○ Better OPEX cost model compared to Rehost ○ Reuse of code considered as strategic or differentiating. **−** ○ Cloud lock-in on PaaS
Rebuild/Refactor/Re-architect	Cloud-native move to PaaS, completely replacing the application and maximum use of cloud benefits. This is the developer's dream, where you go in the whole way, and use as much as possible Platform--as- A-Service from the cloud, so you obtain the best operational expense model.	**+** ○ Full cloud-native benefits such as scalability ○ Most optimal OPEX cost model for custom software **−** ○ Lock-in when using PaaS services
Replace/Repurchase	Discard the application and move to a subscription-based Software-as-a-Service product.	**+** ○ Fully outsourced application **−** ○ Possible data lock-in ○ Potentially difficult to customize and integrate

Cloud migration strategies

After you have selected one of the strategies, you need to look at two factors. The first factor is the Capital Expense if you choose to Rehost, Refactor, Rebuild or Replace. With all these strategies, you need to invest in your solution before you can run in the cloud. Next you need to look at the operational expense of running this application in the cloud for the next 5 years. This will result in a graph that can show you the total amount you will spend in the next 5 years, given a selected strategy.

To give you an idea of the result of this approach we will give an example of one of the cases we have seen in the field. The following graph depicts a 5-year plot of the capital and operational expenses of a product that needed to be moved to the cloud.

The problem statement that needed to be answered in this case was: how can we move our product to the cloud as quickly as possible with a capital expense as low as possible and still have a cost-effective operational expense for running the product for our customers in the long term.

For this specific case an estimate was made of the number of users of the product in each year and of the required changes to the product for each strategy.

— Rehost (Lift and Shift) — Rebuild (Cloud Native)
— Refactor (Multi tenant) — Replace (Saas)

When you look at this graph you see that a Rehost strategy will have a low upfront investment (capex) in the case of this product, but operational expenses make the total costs increase linearly. This is due to the costs of required virtual machines being used each year.

The Refactor strategy focuses on changing the product to make it multi-tenant, which is more resource-effective than a Rehost strategy. This is because the product will no longer require dedicated resources for each customer.

The Rebuild strategy will have the highest initial cost, because of the impact of required changes to the product. On the other hand, annual operational costs are relatively low because of the optimal usage of native cloud services, resulting in zero server maintenance.

The Replace strategy will substitute the product for a Software-as-a-Service solution. This is an interesting scenario from a cost perspective. The subscription of the product will be paid each period. The important question here is whether the product is a strategic differentiator. In other words: does the product contribute to innovation and differentiation of the product compared to the competition.

Looking at the problem statement, the question is: when does it become economically viable to make the investment to change the product, allowing the investment to be earned back in the long term.

You can see that the Rebuild strategy has the best operational cost model. After the initial investment, the cost line hardly increases compared to the other cost lines. Eventually it will be the cheapest, but the return on investment will take a couple of years because of the high initial costs. Another drawback is that time-to-market of the product will be longer compared to

strategies requiring less significant changes. If you only do the lift-and-shift, you will see that within one year it will be more expensive then refactoring the application to support multi-tenancy. The latter scenario is cheaper because one instance of the application will provide service to multiple customers whereas a dedicated application provides service to a single customer.

Based on this graph you could deduce a strategy of starting with a lift-and-shift, then start the investment first for multi-tenancy (Refactor) and then Rebuild it to cloud-native. Since it probably won't make a lot of sense to refactor and rebuild at the same time, you may want to choose a scenario where you look for options to gradually refactor towards a full cloud-native product. The main benefit of this example is that this way of assessing applications will yield a set of possible scenarios and it will give you insight into a hard business case to make the investment and determine the best strategy.

An Enterprise perspective

When you are in an enterprise organization you can also make this same kind of assessment, but the difference will be that you need to deal with a portfolio of applications. In this case, you are best off to first divide the applications roughly into three categories.

» Custom-built
» Custom-built by a partner
» Off-the-shelf and hosted on-premise

The category Custom-built involves software that is built to make a difference for the company. Gartner calls these systems "Systems of Innovation". These systems can be assessed in exactly the way we described for the ISV and from that you can pick the best scenarios.

The category Custom-built by a partner includes the systems of differentiation, where you often buy a partial off-the-shelf solution, but fully tailor this to the needs of the organization. These systems can also be moved to the cloud. In this case, you often ask the partner that built the product to take care of this.
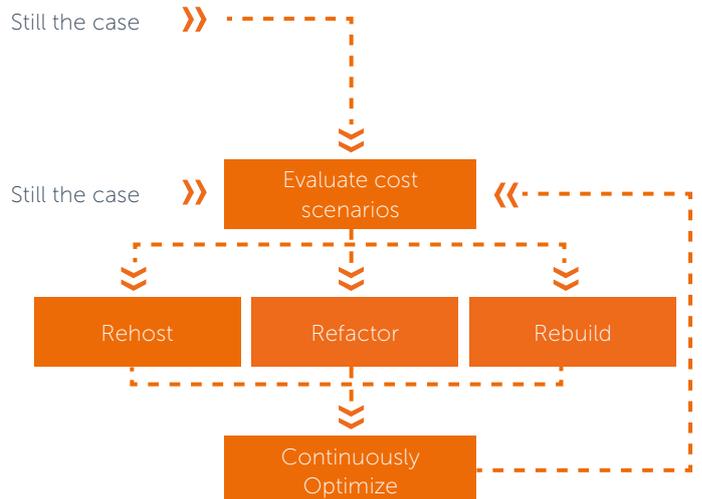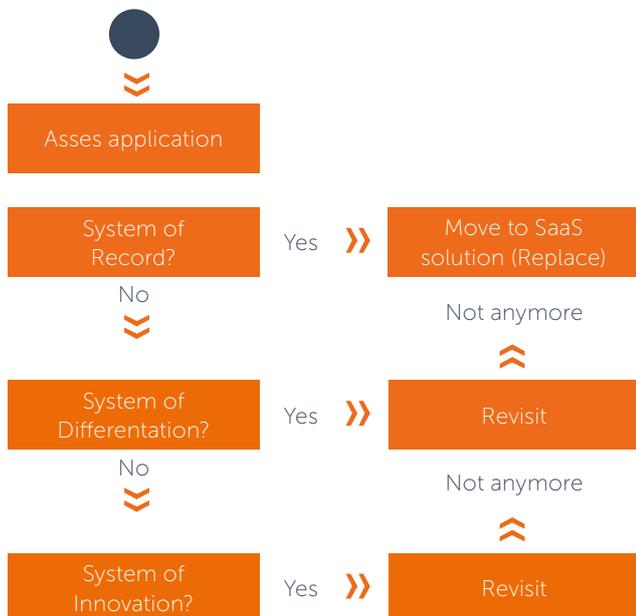
Decision tree for application portfolio assessment

The category off-the-shelf, usually means "Systems of Record". These are the systems you need, but everyone has the same and it is just there to ensure that you can run your business. There is no way you can gain an advantage by doing this differently from your competitor. For this category, you can often just move to the Software-as-a-Service solution of the vendor.

How to assess the applications you will move?

When assessing the applications that are going to move to the cloud, they need to be assessed on multiple levels. As stated before, it is important to determine the current phase in the application lifecycle. Next: where do we want to go with this product? At one time it may have been a "system of innovation", but by now each competitor also has it. This is the moment you choose to replace it with a SaaS solution. If the system will still be differentiating or innovating, it makes sense to look at the scenarios Rehost, Refactor and Rebuild. Practically this means that the initiative for cloud transformation allows you to Revisit the strategic importance of applications in the portfolio, and gradually migrate towards the cloud. The following figure shows the flow of assessment.

Cloud transitions done right

Looking at cloud migration strategies there is no single strategy that will be appropriate for all applications in the portfolio of an ISV or enterprise. A mix of different approaches is required, based on the value that an application delivers versus the costs (investment and operational) of any selected strategy. Because these strategies depend highly on the situation, application, and types of cost involved, there is no one-size-fits-all solution.

In this article, we have described an approach for creating a business case for your applications when you move them to the cloud. If you ask your development team, you will get a different solution than when you ask your operations team. The major difference is that we have added the economical perspective, and this will allow you to create a balanced view on how to make the transition and predict the costs and benefits. **...**

» xpirit.com

A group of honest, energized and passionate technology professionals, willing and able to do things differently. We are innovators, transformation enablers, and early adopters who embrace change. We advise, create, build, train, and guide you. We are Xpirit. Experts in new Microsoft technology.

At Xpirit we are all deeply passionate about our own particular expertise. Whether it's ALM, DevOps, the Azure Cloud, or Artificial Intelligence, everyone brings their full potential to achieve the customer's end goal: solve IT problems. We do this with a hands-on, down to earth mentality, which we call Pure Consultancy. Honest, close to ourselves and driven by technology. Always seeking the most suitable solution(s).

In our strong relationship with Microsoft we stay ahead in the game. Xpirit CTO Marcel de Vries, a worldwide respected speaker and trainer, has for years been recognized as Microsoft Regional Director. Our consultants also participate actively in the Microsoft community by sharing knowledge, organizing meetups, blogging, and contributing to open source projects.

Get ready for the next wave.
We are on it. You'll like it!