






The 5 Rs of rationalization

12/10/2018 • 3 minutes to read • Contributors     

In this article

[Rehost](#)

[Refactor](#)

[Rearchitect](#)

[Rebuild](#)

[Replace](#)

[Next steps](#)

Cloud rationalization is the process of evaluating assets to determine the best way to migrate or modernize each asset in the cloud. For more information about the process of rationalization, see [What is a digital estate?](#)

The "5 Rs of rationalization" listed here describe the most common options for rationalization.

Rehost

Also known as a "lift and shift" migration, a rehost effort moves a current state asset to the chosen cloud provider, with minimal change to overall architecture.

Common drivers could include:

- Reducing capital expense
- Freeing up datacenter space
- Achieving rapid return on investment in the cloud

Quantitative analysis factors:

- VM size (CPU, memory, storage)
- Dependencies (network traffic)
- Asset compatibility

Qualitative analysis factors:

- Tolerance for change
- Business priorities
- Critical business events
- Process dependencies

Refactor

Platform as a service (PaaS) options can reduce operational costs associated with many applications. It can be prudent to slightly refactor an application to fit a PaaS-based model.

Refactor also refers to the application development process of refactoring code to allow an application to deliver on new business opportunities.

Common drivers could include:

- Faster and shorter updates.
- Code portability

- Greater cloud efficiency (resources, speed, cost)

Quantitative analysis factors:

- Application asset size (CPU, memory, storage)
- Dependencies (network traffic)
- User traffic (page views, time on page, load time)
- Development platform (languages, data platform, middle tier services)

Qualitative analysis factors:

- Continued business investments
- Bursting options/timelines
- Business process dependencies

Rearchitect

Some aging applications aren't compatible with cloud providers because of the architectural decisions made when the application was built. In these cases, the application may need to be rearchitected prior to transformation.

In other cases, applications that are cloud compatible, but not cloud-native benefits, may produce cost efficiencies and operational efficiencies by rearchitecting the solution to be a cloud-native application.

Common drivers could include:

- Application scale and agility.
- Easier adoption of new cloud capabilities.
- Mix of technology stacks.

Quantitative analysis factors:

- Application asset size (CPU, memory, storage)
- Dependencies (network traffic)
- User traffic (page views, time on page, load time)
- Development platform (languages, data platform, middle tier services)

Qualitative analysis factors:

- Growing business investments.
- Operational costs.
- Potential feedback loops and DevOps investments.

Rebuild

In some scenarios, the delta that must be overcome to carry forward an application can be too large to justify further investment. This is especially true for applications that used to meet the needs of the business, but are now unsupported or misaligned with how the business processes are executed today. In this case, a new code base is created to align with a [cloud-native](#) approach.

Common drivers could include:

- Accelerate innovation
- Build apps faster
- Reduce operational cost

Quantitative analysis factors:

- Application asset size (CPU, memory, storage)
- Dependencies (network traffic)
- User traffic (page views, time on page, load time)
- Development platform (languages, data platform, middle tier services)

Qualitative analysis factors:

- Declining end-user satisfaction.
- Business processes limited by functionality.
- Potential cost, experience, or revenue gains.

Replace

Solutions are typically implemented using the best technology and approach available at the time. In some cases, software as a service (SaaS) applications can meet all of the functionality required of the hosted application. In these scenarios, a workload could be scheduled for future replacement, effectively removing it from the transformation effort.

Common drivers could include:

- Standardize around industry-best practices.
- Accelerate adoption of business process driven approaches.
- Reallocate development investments into applications that create competitive differentiation or advantages.

Quantitative analysis factors:

- General operating cost reductions
- VM size (CPU, memory, storage)
- Dependencies (network traffic)
- Assets to be retired

Qualitative analysis factors:

- Cost benefit analysis of the current architecture versus a SaaS solution.
- Business process maps.
- Data schemas.
- Custom or automated processes.

Next steps

Collectively, these 5 Rs of rationalization can be applied to a digital estate to make rationalization decisions regarding the future state of each application.

What is a digital estate?