Cloud Migration Methods 2017 State of the Market Report

GLASSHOUSE TECHNOLOGIES

Authored by: John Merryman & David Baird

Table of Contents

1 Executive Summary 3

2 Migration Approaches 4

2.1 Parallel Design & Deploy (Rearchitecting / Refactoring) 4

2.2 Parallel Build & Uplift (Replatforming) 5

2.3 Lift & Shift (Rehosting) 5

2.4 Disaster Recovery 6

2.5 Summary 7

3 Cloud Provider Migration Solutions 9

3.1 Amazon Web Services 9

3.2 Microsoft Azure 9

3.3 Google 9

4 3rd Party Migration Technologies 10

5 Getting Started 11

List of Figures

Figure 1 Parallel Design & Deploy (Rearchitecting) 4

Figure 2 Parallel Build & Uplift (Replatforming) 5

Figure 3 Lift & Shift (Rehosting) 6

Figure 4 Disaster Recovery 7

Figure 5 - Migration Technologies 10

1 Executive Summary

This report outlines commonly used approaches and related technologies for moving existing applications into public cloud environments. In the current public cloud technology landscape, Amazon Web Services, Microsoft Azure, and Google Cloud Platform dominate the collective market, yet provide varied methods, approaches, and technologies to enable customer application transitions.

The intended audience includes technical directors, managers, architects, and systems administrators, who are tasked with transitioning applications to public cloud. Note, as a vendor independent consultancy, GlassHouse provides a neutral view of market options and technologies, based on our field and industry experience.

The benefit of consuming this report is to obtain a point in time expert view of market options and approaches to moving on premises applications and workloads to public cloud providers. While important and related to this topic, we do not cover DevOps, PaaS, or application redesign/development in the scope of this report. We provide a view of typical application migration approaches as seen in the field, public cloud provider migration enablement programs, and a consolidated view of relevant 3rd party migration tools in the market.

2 Migration Approaches

To set appropriate context, we start with a summary overview of four common approaches for application migration to the public cloud. While there are other variants or methods, these methods are the most commonly applied by customers and systems integrators performing migrations.

2.1 Parallel Design & Deploy (Rearchitecting / Refactoring)

Entailing maximum effort, and maximum benefit of public cloud characteristics, the Parallel Design and Deploy approach begins with detailed design of target systems, typically focused on right-sizing compute and storage components, and maximising the leverage of public cloud service characteristics. This approach requires more effort, planning, and detailed design, as the platform is re-architected for use in the public cloud, although the costs of which are offset by the 'long-game benefits' of building a new environment for an application environment.

1. Detailed application and service discovery
2. Mapping of existing system components to Cloud components
3. Determination of viability of containerisation or microservices architecture
4. High-Level Design / Low-Level Design
5. Deployment of Target Systems Infrastructure
6. Application Stack Build
7. Integration / Replacement of System Components
8. Testing / Validation
9. Production Cut-Over
10. Transition to Target Operations/DevOps

Where used, this approach frequently addresses the migration of one application at a time, with varying degrees of integration to core public cloud service layers.

Whilst re-architecting of components to best fit the cloud provider's service offering requires significant effort, the approach is well suited where the business is looking to undertake both transformation and transitioning activities as part of the migration process.

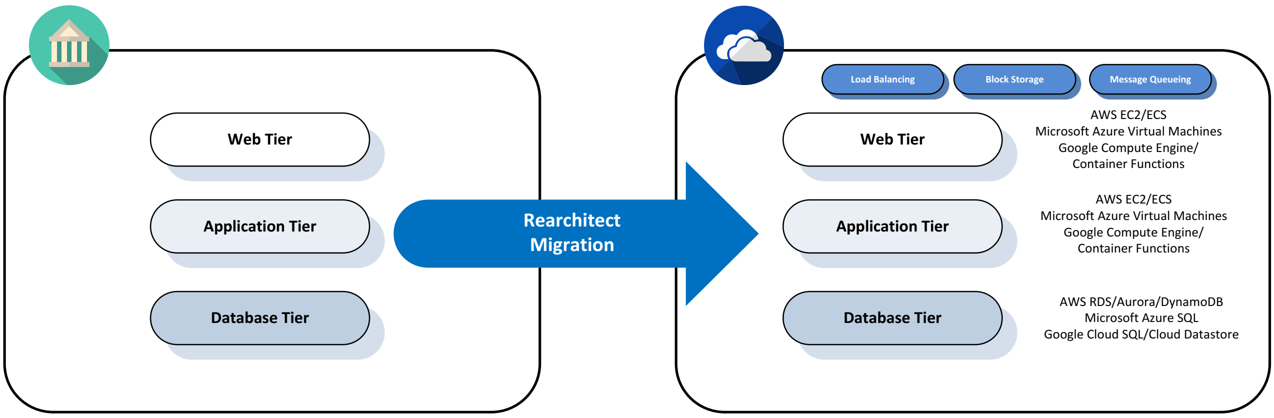


Figure 1 Parallel Design & Deploy (Rearchitecting)

The use of containerisation or a microservices architecture may be considered as part of the transformation activities, further increasing the complexity of the migration, but allowing a fundamental change in the way the application is developed and deployed.

It should be noted that transformation of the application, such that it relies heavily upon the cloud provider's distinct service offerings and ecosystem can lead to long term lock-in with that cloud provider. The possible impact of this should be addressed as part of the planning and design activities.

2.2 Parallel Build & Uplift (Replatforming)

An alternate method for parallel design and deploy is to Replatform the existing environment. This has a lot of similarity to the Rearchitecting / Refactoring method, although the degree of transformation of application components and subsystems is limited to those components where there is a close match for existing component functionality – examples of which include databases, load balancers, compute resources, and storage resources.

1. Detailed application and service discovery
2. Mapping of existing system components to Cloud components
3. High-Level Design / Low-Level Design
4. Deployment of Target Systems Infrastructure
5. Application Stack Build
6. Integration / Replacement of System Components
7. Testing / Validation
8. Production Cut-Over
9. Transition to Target Operations/DevOps

This approach is well suited where the services being consumed are well understood and the cloud provider's offering can be adopted without wholesale change to the application. In this migration model, focus is on transitioning the environment, with transformation used for a subset of readily interchangeable components.

Risks associated with cloud provider lock-in may also be mitigated, with transformed components using the more ubiquitous service offerings from each of the cloud providers.

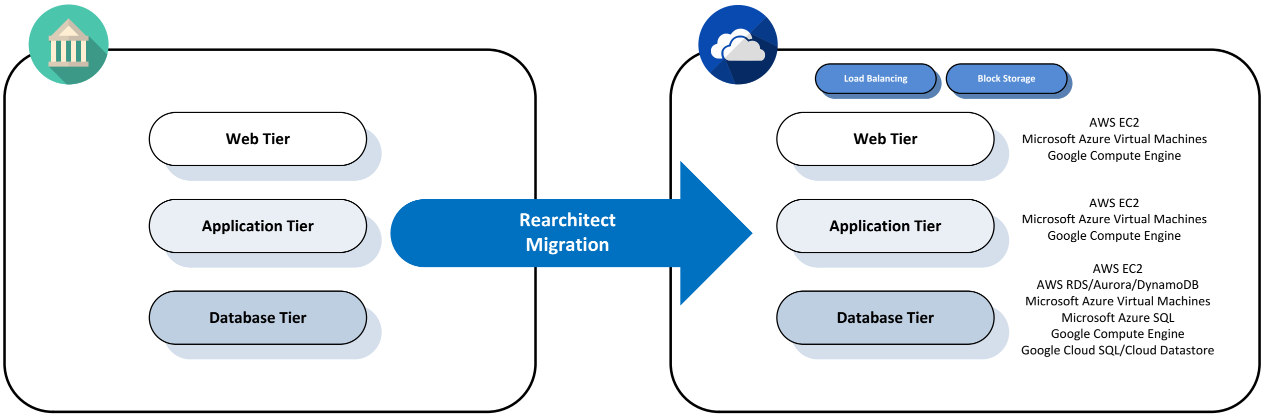


Figure 2 Parallel Build & Uplift (Replatforming)

2.3 Lift & Shift (Rehosting)

Lift & Shift migration tools typically deliver a full clone and migration and data synchronisation for source servers to public cloud virtual machine instances. Applications and services running on source server instances are effectively moved 'as-is' and transitioned wholesale to the target environment. The relative low-cost and low-complexity aspect of Lift & Shift results in a cloned source environment with server instances and applications moving 'as-is' to the target cloud environment.

1. Application to server instance discovery and interdependency analysis
2. Rightsizing factoring for compute, memory, storage (although most L&S moves are like for like).
3. Target VPC design/build
4. Migration move-group planning
5. Migration / Differential Synchronisation
6. Testing / Validation
7. Production Cut-Over

Migration can address the movement of one to many applications at a time, although this is typically limited by network bandwidth, change windows, and application testing/validation resource availability.

This method of migration is often considered when there is no desire to fundamentally re-architect the application design or deployment methodologies, or where the application and/or infrastructure components need to be maintained in a manner similar to the legacy environment.

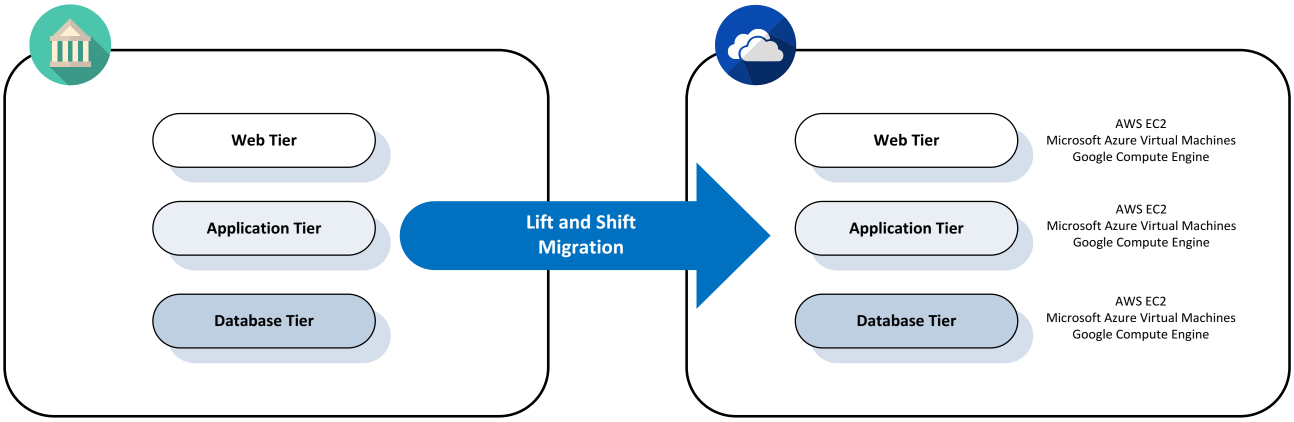


Figure 3 Lift & Shift (Rehosting)

2.4 Disaster Recovery

Disaster Recovery application builds using public cloud offer a relatively low-risk 'gateway' approach to moving applications to public cloud. The primary objective is disaster recovery, with a secondary benefit being the proof point of cutting over production workloads to a cloud based warm-standby system. Depending on the disaster recovery requirements, systems design in the target can follow 'like for like', or an uplifted design specification with the intent of eventually cutting production workloads over to the target/DR environment.

This approach requires significant effort, planning, and detailed design, with dual benefits of risk mitigation coupled with a gradual/phased migration of an application to public cloud.

1. Detailed application and service discovery
2. High Level Design / Low Level Design for Disaster Recovery
3. Deployment of Target Systems Infrastructure
4. Application Stack Build
5. Testing / Validation
6. Disaster Recovery Testing/Cut-Over
7. Production Testing / Cut-Over

In this migration approach, the movement of one application at a time is considered with Disaster Recovery as a phase-1 transition step. As with the lift and shift model, limited degrees of integration to core public cloud service layers are undertaken.

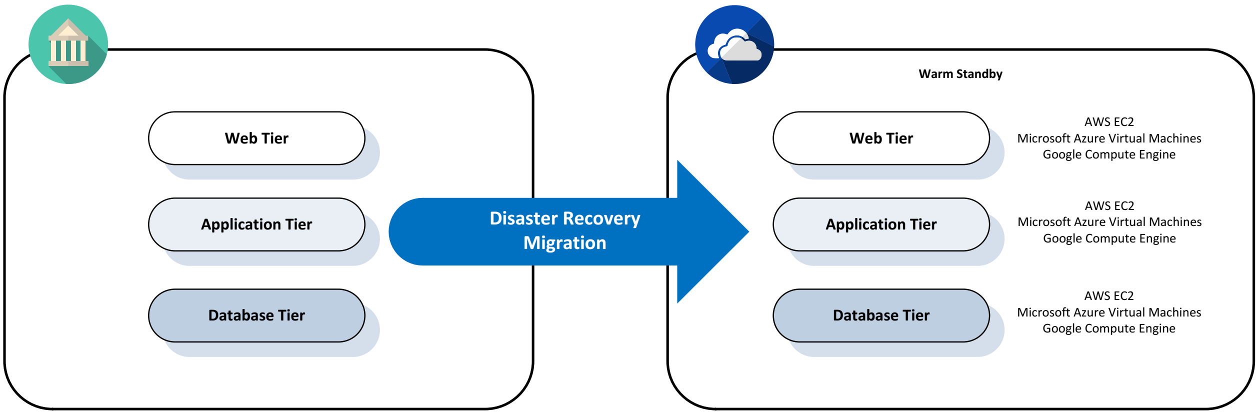


Figure 4 Disaster Recovery

2.5 Summary

In this summary, we outline the following benefits and drawbacks of each approach outlined above.

Approach	Pros	Cons
Parallel Design & Deploy (Rearchitect)	Fresh systems build, unlikely to carry-over any defects from source environment Allows for consideration of current Cloud design methodology and operations Efficiency and modernisation benefits likely	Significant effort and cost Application and systems acceptance testing required due to source build change High degree of customisation within the end-state environment that can lead to long term lock-in to cloud provider's service offerings Requirement to upskill operational support teams in re-architected application and infrastructure design
Parallel Build & Uplift (Replatform)	Fresh systems build, unlikely to carry-over any defects from source environment Allow for moderate consideration of current Cloud design methodology and operations Moderate efficiency and modernisation benefits likely	Moderate effort and cost Application and systems acceptance testing required due to source build change Moderate degree of customisation within the end-state environment that can lead to long term lock-in to cloud provider's service offerings – may be mitigated by using offerings common to majority of cloud providers Requirement to upskill operational support teams in re-architected application and infrastructure design
Lift & Shift Rehosting	Relatively quick and simple	System anomalies (known or unknown) may manifest later in target production

	Excellent for 'last-minute' migrations where detailed design/plan is not feasible Suitable for workloads that must remain largely as is, in terms of infrastructure deployment Low complexity and cost Lower degree of lock-in to cloud provider's service offerings	Limited to no optimisation in target cloud, outside of shifting footprint Limited change to resiliency or availability of application – the status quo model would be retained in end-state
Disaster Recovery	Low Risk, limited risk of rollback Gradual or phased migration to public cloud production Improves DR risk profile for application Target DR system can be replicated into another region or availability zone for DR/Production in public cloud Less applicable to large scale application migration initiatives due to cost of on premise and target cloud operating costs	Extra planning and testing associated with DR plus production cutover DR designs can require uplift for full production workloads Target DR system design is less likely to have deep integration with target cloud service layers and may require additional uplift to this end

Table 1 Migration Approach Summary

3 Cloud Provider Migration Solutions

Next, we profile the Cloud Service Provider programs and technologies as promoted by AWS, Azure, and Google for customer enablement and migration.

3.1 Amazon Web Services

Amazon recently announced the Migration Hub as an integrated gateway for AWS developed and third-party migration technologies. The AWS Discovery Service and Server Migration Service (SMS) are provided at no-cost, and together allow customers to inventory, group, and migrate workloads to AWS. Note, that SMS currently supports vSphere source environments only. Additional third party migration tooling is available from CloudEndure and Racemi, allowing customers to address a wider range of source environments and articulate more fine-grained controls over data replication vs. the SMS VMware snapshotting methodology.

While the AWS Migration Competency program includes numerous vetted and approved third party Independent Software Vendors (ISV) for migration discovery, planning, workload mobility, and application testing, a smaller subset is initially integrated with the Migration Hub, presumably due to the API links to application grouping and migration status tracking. We expect this integration to include more third parties, however note the leading placement of native AWS services in this customer facing model.

Note: The AWS Migration Hub is documented with general API access points

3.2 Microsoft Azure

Microsoft originally acquired Inmage in 2014 to compliment the Site Recovery Manager portfolio. At the time Inmage was a data-mover client/server based application, providing data replication for block data stores to supported target clouds. Inmage is now enveloped into a single onboarding capability for Azure workloads: Site Recovery Manager. Like Amazon SMS, Site Recovery Manager is subsidised as a no-cost offer for Azure onboarding, with the aim to remove commercial and technical friction for customers migrating workloads from on premises to Azure.

As this capability emerges along with AzureStack, we anticipate a highly integrated experience for HyperV / AzureStack workload mobility to (and presumably from) Azure.

In addition to Site Recovery Manager, Microsoft Azure also highlights key ISV partnerships for TCO/Costing and migration discovery, but has limited focus on a wider ecosystem of partners and ISVs across the migration lifecycle.

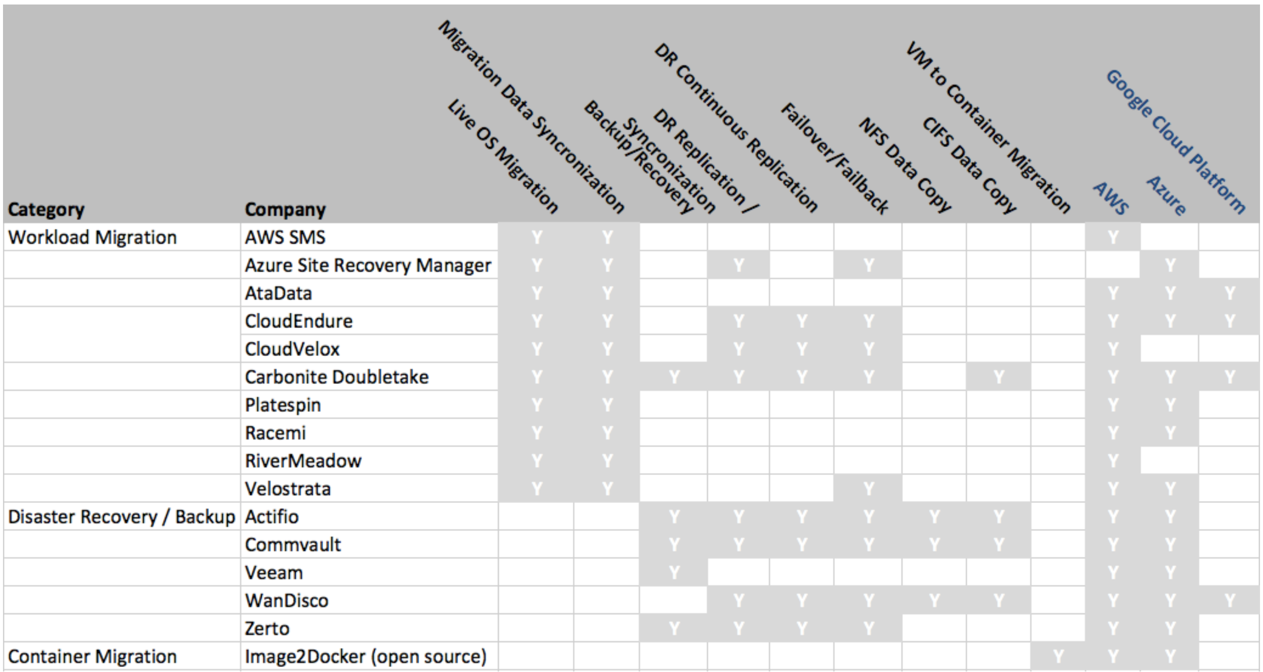
3.3 Google Cloud Platform

Google in 2017 announced a third party capability via CloudEndure providing subsidised migration software for customers migrating to Google Cloud Platform (GCP). Noting Google's development culture, we see this as an initial step towards customer enablement, but note the partner ecosystem and ISV tooling for migration lifecycle activities is yet to be developed. The Google approach also stands out to AWS and Azure's internally developed platforms and multi-party ISV/ecosystem elements.

4 Third Party Migration Technologies

We see a diverse mix of technologies all competing directly or indirectly for the migration market. Our view is that whilst migration tools are extremely valuable, these however cannot be treated as stand-alone silver-bullets to an enterprise public cloud migration. In reality, enterprise migration projects will incur a number of migration approaches, along with specific migration tools to satisfy migration use-cases, operational constraints, and business requirements.

Per the below matrix, we have summarised a number of third party ISV's participating in the public cloud migration ecosystem. This information summarises technology functions, and does not include product architecture, ease of deployment, ease of use, and solution flexibility.



Category	Company	Live OS Migration	Migration Data Syncronization	Backup/Recovery	DR Replication / Syncronization	DR Continuous Replication	Failover/Failback	NFS Data Copy	CIFS Data Copy	VM to Container Migration	AWS	Azure	Google Cloud Platform
Workload Migration	AWS SMS	Y	Y								Y		
	Azure Site Recovery Manager	Y	Y		Y		Y					Y	
	AtaData	Y	Y								Y	Y	Y
	CloudEndure	Y	Y		Y	Y	Y				Y	Y	Y
	CloudVelox	Y	Y		Y	Y	Y				Y		
	Carbonite Doubletake	Y	Y	Y	Y	Y	Y		Y		Y	Y	Y
	Platespin	Y	Y								Y	Y	
	Racemi	Y	Y								Y	Y	
	RiverMeadow	Y	Y								Y		
	Velostrata	Y	Y				Y				Y	Y	
Disaster Recovery / Backup	Actifio			Y	Y	Y	Y	Y	Y		Y	Y	
	Commvault			Y	Y	Y	Y	Y	Y		Y	Y	
	Veeam			Y							Y	Y	
	WanDisco				Y	Y	Y	Y	Y		Y	Y	Y
	Zerto			Y	Y	Y	Y				Y	Y	
Container Migration	Image2Docker (open source)									Y	Y	Y	

Figure 5 - Migration Technologies

Workload migration tools move operating system instances, with installed applications and user/block data transitioning from source to target. The technical implementation of the copy function varies from one to the next, as does the product architecture and ease of implementation/use. Disaster recovery and Data Protection vendors provide image copy and migration capability; however implementations and complexity vary widely to accomplish this result.

Value added features including target provisioning, instance right-sizing, and API end-points are major differentiators to reduce the amount of pre/post migration manual effort. We advise customers perform trial migrations against defined use-cases, or engage consultancies who have embedded third party tooling into migration delivery frameworks.

5 Getting Started

For moving existing on premises applications to public cloud, we advise piloting initial applications to establish early proof-points, and in concert take a careful eye to operations and governance as the deployment to public cloud gains momentum. In this whitepaper we outlined common approaches, cloud provider programs, and third-party technologies, which collectively provide a framework for moving applications to cloud, with varying degrees of risk, complexity, and cost-benefit.

We help customers simplify and accelerate application migrations to public cloud. Glasshouse offers Cloud Transition portfolio services, including Cloud Compute Costing, Cloud Accelerate, Application/Workload Migration, and Application Managed Services.

For questions or a technical requirements review, please contact us at: info@glasshouse.io