

A decorative graphic on the left side of the slide. It features a large, light blue arrow pointing right, with a smaller, darker blue arrow pointing right inside it. Above the arrows are several light blue clouds. Below the arrows are several smaller, light blue arrows pointing in various directions. The background is a solid blue color.

Cloud Migration

STRATEGIES & BEST PRACTICES

WHY MIGRATE?

- Scalability to accommodate growing data requirements
- Business continuity and disaster recovery
- Faster development, iterations, and reduced provisioning time
- Improved IT resource management
- Improved cost management
- Collective IT Insights
- Data security
- Reduced carbon footprint

KEY CONSIDERATIONS

- Cloud vendor and type of cloud
- Cost structures - On-premise *versus* Cloud *versus* Hybrid
- Regulatory and compliance implications
- Security and data protection
- Backup and disaster recovery
- Portability: Ability to migrate or switch service providers easily
- Conformance with existing SLAs
- Skills required to build and operate a cloud environment

STRATEGIC ADVISORY SERVICE



Face-to-face
interviews with
business and IT



Architectural
assessment



Opportunities for cost
and capacity
optimization



Assessment of cloud
operational maturity



Adherence to
regulatory and
compliance
requirements - HIPAA,
PCI, SOC 1&2, GDPR



Roadmap toward
target state



Validation of business
case



Identification of issues
and possible risk
mitigation options

Migration to the cloud should be approached in a phased manner, ensuring minimal to no disruption to operations.

CLOUD MIGRATION STRATEGY



What to migrate

Prioritize the applications, processes, and infrastructure to be migrated



How to migrate

Formulate a migration strategy that aligns with needs and addresses potential risks



Where to migrate

Choose a reliable platform — private, public, or hybrid — as appropriate for the business

DISCOVERY

Validate current state and inventory. Gain a 360 degree view of infrastructure, applications, and their relationships

Inventory

- Servers: Physical and virtual, hypervisor, OS, CPU, RAM, disk
- Software: All software assets, including end of life, databases and websites
- Network devices: Switches, load balancers
- Storage: Devices and their logical partitioning

Dependency data

- Software dependencies — web, application, and database tiers, and clustered software configurations
- Containers and microservices
- Server-to-storage relationships to understand how the data flows
- Hybrid application deployments
- Host-to-edge network relationships
- Hardware and software load balancing
- Disaster recovery setups

DECIDING THE APPROACH

Rehost

- Lift and shift, minimizing changes, virtualization replatforming
- Physical to virtual to cloud

Refactor

- Lift, tinker, and shift. Initiate cloud optimizations.
- Revise applications to leverage common services refactor

Rebuild

- Replace application with SaaS service
- Build cloud native application with similar/improved characteristics and features

Revise

- Develop and productize common business and technical services
- Consolidate similar applications and services

Replatform

- Move to more cloud-aligned technology and platform services
- Integration with cloud operations and monitoring replatform

PUBLIC, PRIVATE, AND HYBRID CLOUDS



Public Cloud

Hardware, software, and other supporting infrastructure are owned and managed by the cloud provider.



Private Cloud

Services and infrastructure are maintained on a private network and the hardware and software are used exclusively.



Hybrid Cloud

Combine on-premises infrastructure or private clouds with public clouds so organizations can reap the advantages of both.

PUBLIC, PRIVATE, AND HYBRID CLOUDS: PROS & CONS

Public Cloud



- Lower cost
- Infinite space available
- Offers access to advanced technology



- No control over infrastructure
- Data is vulnerable to thefts
- Lack of customization

Private Cloud

- Assures a high level of security and privacy
- Improved efficiency and control
- Easy to meet compliance requirements

- Resource optimization is a challenge
- Hardware limitations lead to capacity ceiling
- High cost

Hybrid Cloud

- Better business continuity/disaster recovery
- Cost savings for specific projects
- Unique balance of control, performance, and scalability

- Security complexities
- Possible network bottleneck
- Difficult to maintain visibility

CHOOSING BETWEEN SINGLE AND MULTI-CLOUD

Single Cloud Provider

- Development teams have just one set of cloud APIs to learn
- Application can benefit from the features offered by the chosen cloud provider
- Downside: Vendor lock-in

Multi-cloud Approach

- Run applications across multiple cloud environments (choose cloud based on the capabilities of each provider)
- Downside: Increased complexity

CLOUD NATIVE VS CLOUD AGNOSTIC

	Cloud Native	Cloud Agnostic
Attributes	<ul style="list-style-type: none">• Easy to set up• SaaS available for most tools• Easy integration	<ul style="list-style-type: none">• Can be implemented on and moved between cloud platforms• Application templating
Vendor lock-in	<ul style="list-style-type: none">• Vendor-specific• Managed services• Only one set of APIs	<ul style="list-style-type: none">• No vendor lock-in• No unified API for calling cloud computing services• Dependent on in-house skillset
Business value	<ul style="list-style-type: none">• Auto-provisioning• Auto-scaling• Auto-redundancy• Security	<ul style="list-style-type: none">• Single management interface for monitoring and reporting• Availability of cloud-neutral tools such as Terraform and Ansible

PRIORITIZE MIGRATION COMPONENTS

Migrate your entire application at once **OR**
Migrate component-by-component or service-by-service



First

Identify connections between services and interdependencies



For larger, complex applications

Use a monitoring application such as **New Relic APM** that offers service maps to generate dependency diagrams



Dependency diagram

Decide which components should be migrated and in what order



Start by migrating services that have fewer dependencies



Start with outward/customer-facing services/apps

ESTABLISH PERFORMANCE BASELINES

Baselining is the process of measuring the current (pre-migration) performance of your application or service in order to compare future (post-migration) performance.

- Set a baseline metric for each KPI that you've decided to measure
- Determine how long you will collect data to determine the baseline
- Choosing a short baseline period (such as a day) lets you move faster, but you risk not collecting a representative performance sample
- Choosing a longer period to baseline (such as a month) obviously takes more time, but can provide more representative data

ESTABLISH CLOUD KPIS

User experience

- Page load time
- Lag
- Response time
- Session duration

Infrastructure

- CPU usage %
- Disk performance
- Memory usage
- Network throughput

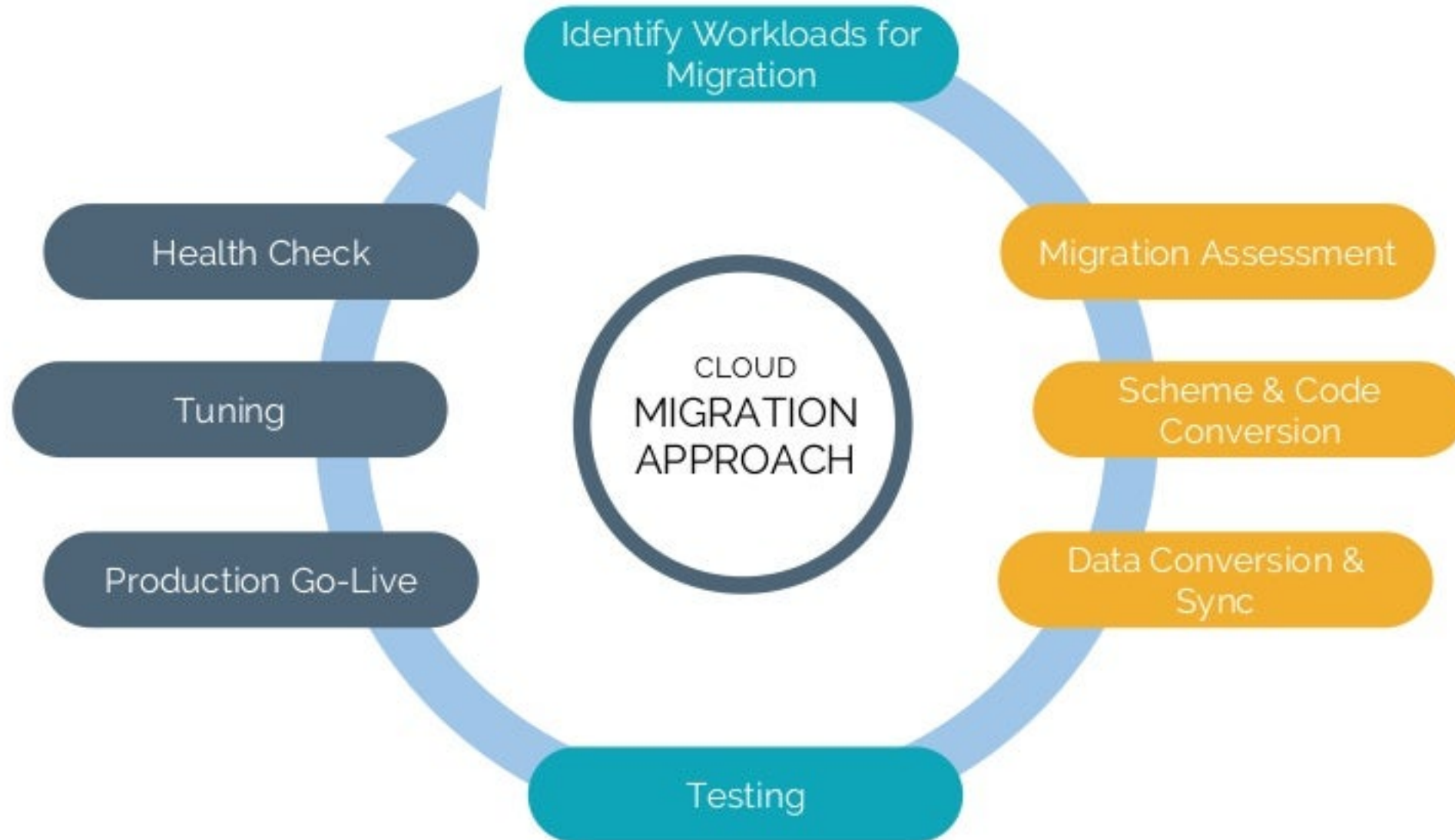
Application/component performance

- Error rates
- Throughput
- Availability
- Apdex

Business engagement

- Cart adds
- Conversions and conversion %
- Engagement rates

CLOUD MIGRATION APPROACH



CLOUD MIGRATION ROLLOUT



Beta Environment Setup

- Set up a beta environment of the existing environment
- Test/fine tune application using test data and test database
- Test scalability after loading production equivalent test data



Migration

- Build a new production environment
- Adopt DevOps practices
- Move production data and set up storage, DR, and fault tolerance
- Update DNS records and other configurations



Continuous Monitoring

- Set up monitoring tools
- Keep watching logs, performance, response time, and load

DATA MIGRATION BEST PRACTICES

- Use a **bi-directional syncing mechanism** between on-premise and cloud databases. Once all consumers of the data are moved to the cloud, remove the on-premise database.
- Use an **on-premise database with one-way synchronization** to a cloud-based database, and allow consumers to connect only to the on-premise version. When ready, disable access to the on-premise version so the cloud-based version becomes the main database, and enable access for cloud-based consumers to the new database.
- Use a **cloud data migration service**, such as those available from **Amazon Web Services** and **Microsoft Azure**.

REVIEW APPLICATION RESOURCE ALLOCATION

- Resource optimization by dynamic allocation
- Cost efficiencies through architectural and component optimization
- Optimize cloud solution based on real-world data
 - Based on requirement, add H/A functionality to protect against outages, or right-size instances to fulfill requirements at lowest cost
 - Start the monitor-optimize cycle
- Examine backup and disaster recovery policies
- Once all key goals have been met and cloud migration risks avoided, move the application to its steady state
- Runbooks ensure everyone has the information necessary to keep apps running and supported through the rest of their lifecycle
- Optimize customer experiences

SETTING UP A MIGRATION TEAM

Manager (Migrations)

- Knowledge of existing network, applications, and database management technologies.
- An experienced project manager with a sound understanding of how technology fits into the larger company strategy.

Architect (Migrations)

- Responsible for designing the cloud architecture - platforms, servers, storage, content delivery, and networks.
- An experienced technology professional with significant experience in cloud migrations

Cloud Developer

- Responsible for developing and deploying solutions on cloud platforms
- Significant understanding of core IaaS and PaaS platforms.

Cloud Security Specialists

- Responsible for configuring and managing security baselines
- Designs and manages a secure cloud environment
- Certified in cloud security management

- Set up a Cloud Center of Excellence (CCoE) in your organization with representation from various departments including Architecture, Development, Security, Operations, Leadership, and Finance.
- Governance and Communication framework: Dashboards to track progress (Jira).

CLOUD GOVERNANCE PLAN

Cloud governance plan involves:

- Decision making processes
- Criteria and policies involved in the planning, architecture, acquisition, deployment, operation and management of cloud computing capability
- Model formulation to cover security, financial, performance, collaboration & communication governance.



CLOUD GOVERNANCE FRAMEWORK



WHAT WE PROVIDE

- Expertise in a host of cloud services — Amazon Web Services, Microsoft Azure, and Google Cloud Platform
- Scalable and dynamic cloud architecture
- Multi-cloud security and tools experience
- Strong capabilities in pragmatic risk and compliance management
- Experience in cloud migrations





**14150 Newbrook Drive
Suite 115
Chantilly, VA 20151, USA**
www.qburst.com

