

3. Build and release pipelines

└ 3.4 Deployment strategy and execution

└ 3.4.3 Minimize downtime: VIP swap, rolling, slots

-
1. What is a VIP swap and when should it be used in Azure deployments?
 2. How do deployment slots help minimize downtime in Azure App Service?
 3. What is a rolling deployment and how is it executed in Azure?
 4. Compare the benefits and limitations of VIP swap versus rolling deployment.
 5. How do you perform a slot swap in Azure Pipelines?
 6. What considerations are required for zero-downtime deployments in multi-instance environments?
 7. How do health checks affect rolling deployments and slot swaps?
 8. What are common pitfalls to avoid when minimizing downtime during deployment?
 9. How do you roll back safely after a failed slot swap or rolling deployment?
 10. What best practices ensure minimal user impact during production deployments using these strategies?
-

1. What is a VIP swap and when should it be used in Azure deployments?

A VIP (Virtual IP) swap exchanges the public IP addresses of two *Azure* cloud service deployments (usually staging and production) instantly. It should be used to promote a pre-warmed, validated deployment to production with minimal downtime.

2. How do deployment slots help minimize downtime in Azure App Service?

Deployment slots allow you to host multiple app versions (e.g., staging, production) on the same App Service. You deploy to a staging slot, validate, then swap to production, ensuring fast cutover and rollback with minimal disruption.

3. What is a rolling deployment and how is it executed in Azure?

A rolling deployment updates subsets of instances incrementally, removing a portion from the load balancer, updating it, then proceeding to the next subset. In *Azure*, rolling deployments can be managed using *Virtual Machine Scale Sets* or *Kubernetes* for controlled, phased rollouts.

4. Compare the benefits and limitations of VIP swap versus rolling deployment.

- **VIP swap:** Near-instant cutover, easy rollback, but both environments must be identical.
 - **Rolling deployment:** Gradual updates, reduced risk of mass failure, allows monitoring during rollout, but may introduce temporary state inconsistencies and slightly longer total deployment time.
-

5. How do you perform a slot swap in Azure Pipelines?

Add an "Azure App Service deploy" task, specify the target deployment slot (e.g., staging), then use the "Swap Slots" action to exchange staging and production after validation. This can be automated with *Azure CLI* or built-in pipeline tasks.

6. What considerations are required for zero-downtime deployments in multi-instance environments?

Ensure at least one instance is always available by updating instances incrementally, using health probes, and configuring load balancers to direct traffic only to healthy, updated instances. Maintain session state externally to prevent user disruption.

7. How do health checks affect rolling deployments and slot swaps?

Health checks ensure only healthy instances receive traffic during and after deployment. In rolling deployments, unhealthy instances are removed from rotation. For slot swaps, pre-swap validation uses health probes to confirm readiness before traffic is switched.

8. What are common pitfalls to avoid when minimizing downtime during deployment?

- Skipping health checks
 - Deploying breaking schema changes without backward compatibility
 - Not externalizing session state
 - Failing to test slot settings or configuration differences
 - Insufficient monitoring during rollout
-

9. How do you roll back safely after a failed slot swap or rolling deployment?

For slot swaps, immediately swap back to the previous slot. For rolling deployments, stop the rollout and redeploy the previous stable version to affected instances. Always validate state and configuration before rollback.

10. What best practices ensure minimal user impact during production deployments using these strategies?

- Use staging slots and swap only after validation
- Deploy during low-traffic periods
- Implement incremental rollout with monitoring and automated rollback
- Externalize state/session management
- Automate health checks and alerts