Data DevOps Accelerator

What is Data DevOps

Data DevOps is a set of practices that combines deployment of big data development and IT operations. Its purpose is to shorten the systems development life cycle and provide continuous delivery with high software quality and standard environments. Using infrastructure as code allows environments to be quickly created in a consistent, secure manner. This sets a strong foundation for automated code delivery through development lifecycle.

Why You Need This

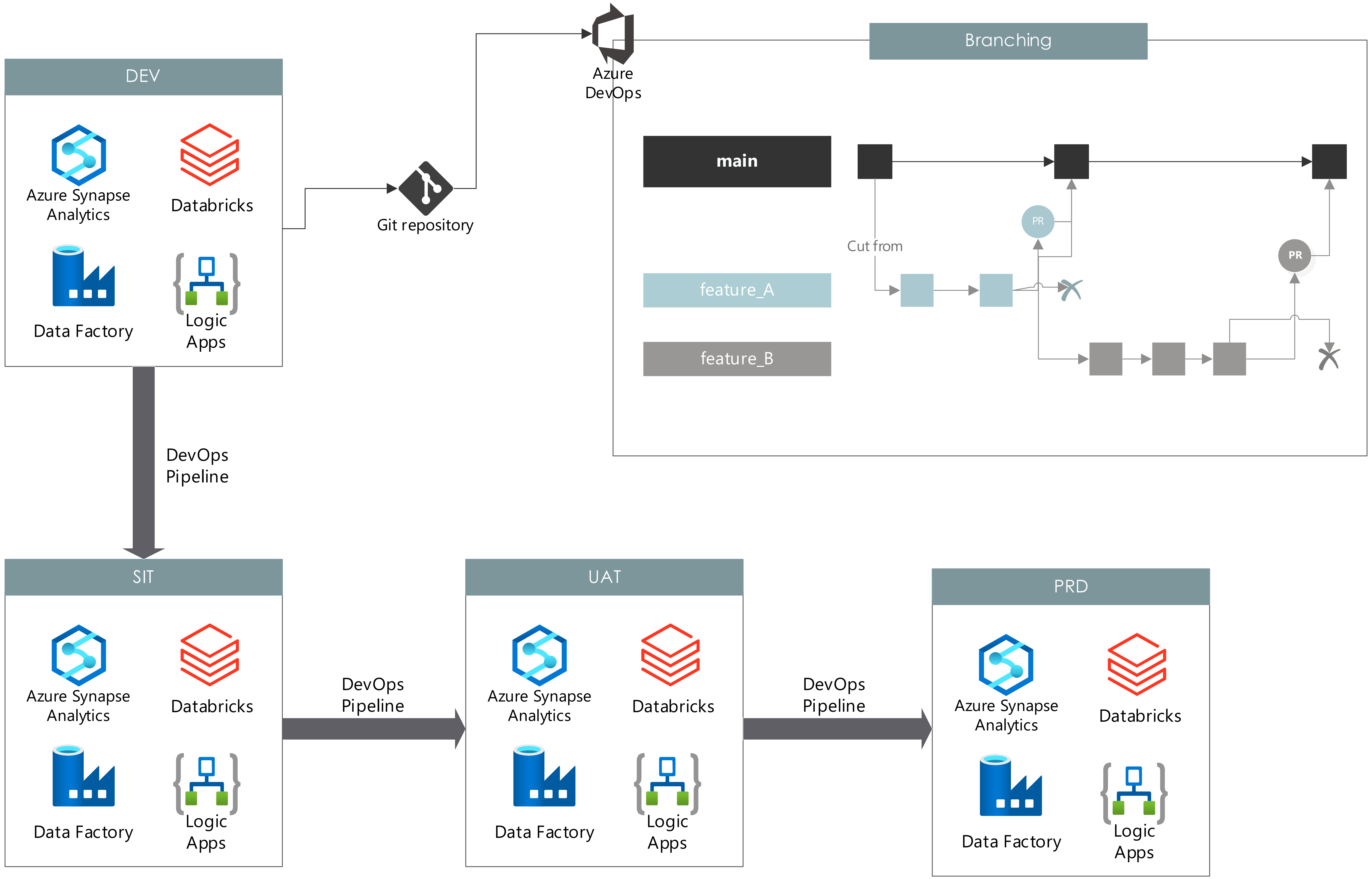
Cost savings – Automating repeatable tasks frees up employees to focus on higher-value areas.

Consistent and repeatable – Using an automated solution will allow teams to easily deploy code in a consistent manner.

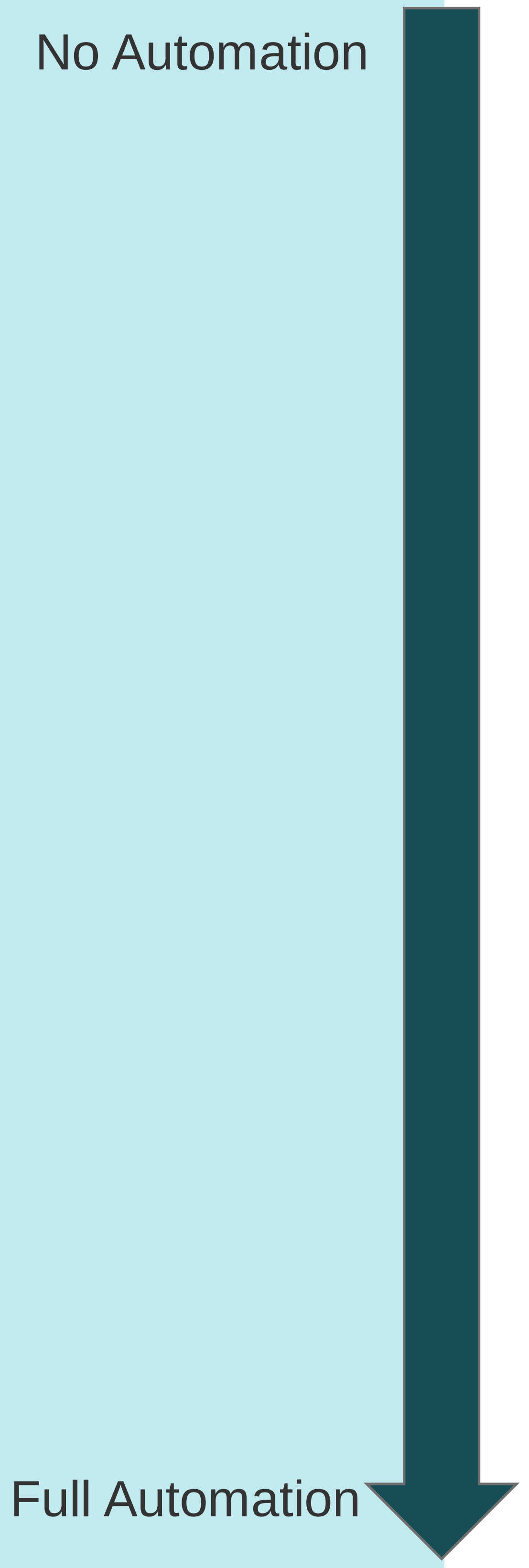
Enforcing standards – Enforcing standards such as coding, testing and naming will keep the solution supportable long-term.

Secure deployments – As workflows move to the cloud security is always a concern; all templates include security settings for firewall and access.

Environment Succession



Deployment Types



Manual Deployment – Manually copying files to resources across environments.

Automated Deployment – Scripts and pipelines to create resources and deploy code. Pipelines and scripts are run manually.

Continuous Integration – Code is merged to a main shared branch often (at least daily). Tests are run against a build that is automatically created by pipelines. This is used to avoid integration complications caused by big releases.

Continuous Delivery - Deployment pipelines deploy code to multiple environments automatically while executing tests to ensure code is deployed as expected.

Continuous Deployment - Changes that pass all stages of the deployment pipeline are released to production without human interaction. Automated testing is used to verify deployments to ensure quality.

Resource Deployment

Terraform - Terraform is a third-party tool for building, changing, and versioning infrastructure safely and efficiently. Terraform can manage existing and popular service providers (Azure, AWS) as well as custom in-house solutions.

Bicep – Bicep is a Microsoft Domain Specific Language for deploying Azure resources. It aims to drastically simplify the authoring experience with a cleaner syntax and better support for modularity and code re-use. Bicep is a transparent abstraction over ARM and ARM templates, which means anything that can be done in an ARM template can be done in Bicep.

Templated Resource Deployment Benefits

Resource templates streamline deployments and allow for consistency the first time. All the resources in your environment can live in a single file which allows you to spin up and down quickly. Each template includes firewall and security settings, so they are not an after-thought. Ensuring you have all the components you need for each environment saves time for developers and eliminates ad-hoc resource provisioning requests.

Branching Strategies

With automation it is important to have a good branching strategy that everyone can follow; SDK has two possible solutions depending on the project size and complexity. This ensures code is properly tested and not overwritten by mistake. These strategies are built for an agile CI/CD workflow.

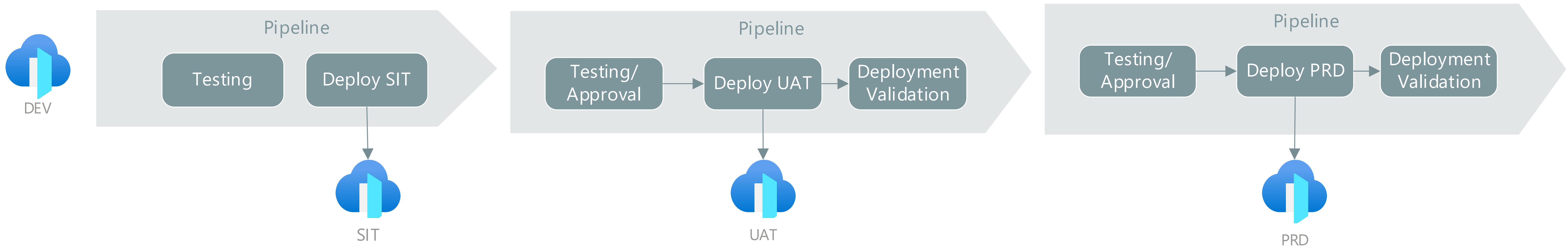
GitHub Flow – Simple branching strategy that supports small teams in isolated environments.

Party Flow – Complex branching strategy that supports multiple projects in a shared environment.

Code Deployment

Data DevOps Pipelines – Azure Pipelines combines continuous integration (CI) and continuous delivery (CD) to constantly and consistently test and build your code and ship it to any target. This solution also allows for approvals to integrate with your current change control process. The solution can support multiple environments.

Pipeline Flow



SDK Azure Templates and Pipelines

Resource Templates	Code Deployment Pipelines
Key Vault Azure SQL Synapse Virtual Machines Network Security Groups API Management Function Apps Log Analytics Event Hub Azure Blob Storage Azure Data Lake Store (Gen 2) Logic Apps Data Factory Databricks	Azure SQL Synapse Logic Apps Data Factory Databricks

Reporting

(Available Soon)

Azure costing – Keep an eye on Azure costs with a Power BI report that can easily tell you what resources are costing.

Security management – Global view of security across resources and data lake, easily audit who has access to what.

Pipeline deployment tracking – Power BI report that shows the history of deployments. Allows you to see who deployed, when and if it was successful.

Why Invest?

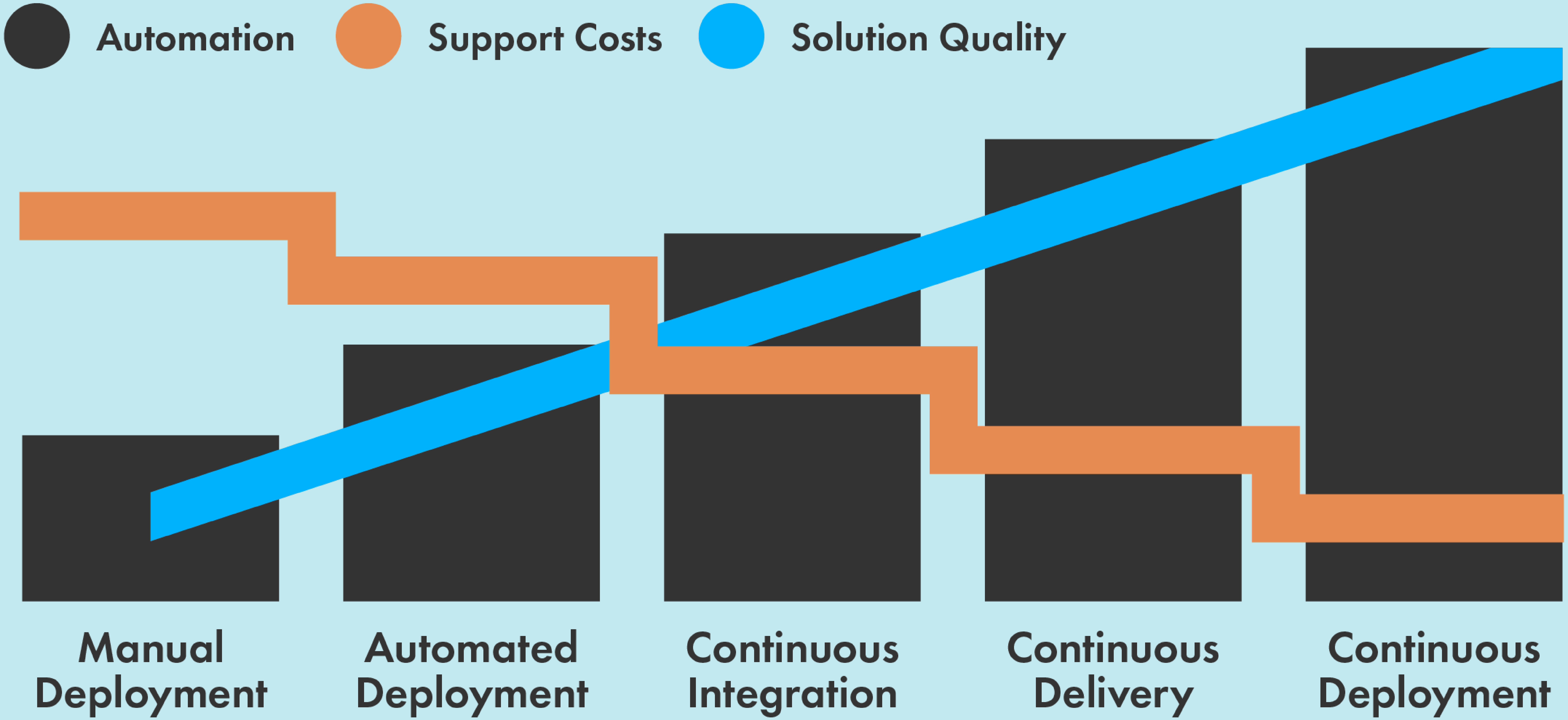
Stable foundation – Quickly build a strong, secure foundation with consistent naming standards and resources.

Consistency – Consistent resources and code through all environments for easy testing and successful code delivery.

Supportability and sustainment – Easier to support with less training required allows for long-term sustainability.

Cost reduction – Reduced effort due to automation and more stable/consistent code all drive lower development and support costs.

Data DevOps Maturity



Thank You.

Please visit SDK for more Information: