Establishment of a standardized and integrated data platform
Part 4: DevOps TEG Case Study

Through their integrated operating model, TEG (Ticketek) bring the best live content, ticketing and technology to their partners and create memories for fans that last a lifetime.

A key differentiator in their approach is being a truly digital company first, enabling data and analytics to help shape the customer experience.

Introduction

TEG are expanding globally and required an efficient and reliable process for rolling out their data platform solution into new regions. In this part of the series, we'll explore how Altis and TEG built effective and robust Continuous Integration / Continuous Deployment (CI/CD) practices to facilitate this.

Source Control

TEG uses Gitlab for their code repositories & source control. New repos were created for the various components of the data platform solution. This allows for fine grained control of the permissions granted to developers working on different parts of the solution.

Gitlab pipelines and runners play an important part of the CI/CD process. Pipelines are written in YAML and contain the jobs, steps and commands associated with the deployment of a solution component. Gitlab runners provide the compute power to execute the pipelines.

The Gitlab UI provides a front-end for engineers to do required configuration changes at deployment time (e.g., specifying the target deployment region) and to monitor pipeline executions.

Pipeline execution works in conjunction with the Git branching strategy. For example, an approved merge request to the develop branch would trigger a pipeline run for the development environment.

Database Schema Management

Altis leveraged an open-source tool called Liquibase to track, version and deploy database changes to Snowflake. Data engineers used VSCode integrated with Gitlab to manage the SQL code.

The SQL files are named and stored in a thought-out folder structure to provide easy navigation of the solution. Liquibase requires the developer to use 'changesets' to organise the code into executable blocks which are tracked during each 'liquibase update' command. Liquibase maintains changelog tables in Snowflake to keep track of changeset deployments and modifications. Changesets give the developer the freedom to allow control over some SQL being executed once-off (e.g., table creation) versus some SQL being re-runnable if a change is detected (e.g. CREATE or ALTER VIEW...)

- **Data**
- **Information**
- **Analytics**
- **Outcomes**

# Establishment of a standardized and integrated data platform
Part 4: DevOps TEG Case Study
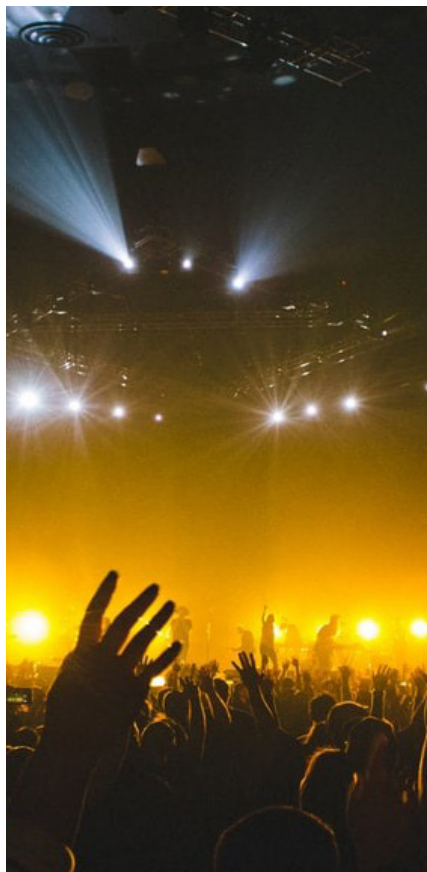
Infrastructure as Code (IaC)

AWS Cloudformation templates were used to provision several AWS resources such as S3 buckets, IAM roles, SQS Queues, Lambda functions etc.

A noteworthy example of a solution component being automated is that of the Tableau server deployment. Cloudformation was used to provision the EC2 instance for Tableau server. Python scripts were used to call the Tableau server REST API in order to provision resources such as projects, permissions, data sources, workbooks etc. A master template of Tableau resources was defined in JSON and ensured that a consistent Tableau environment and user experience was rolled out for each region.
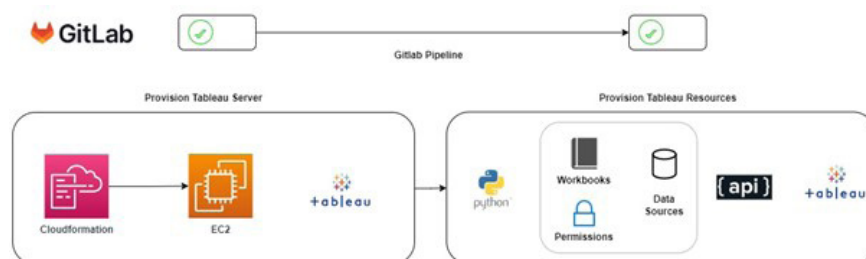
Do you want to find out more about establishing an integrated data platform?

Connect with Altis today to find out how we can help maximise your business performance.

Call Altis today to find out how we can help maximise your business performance.
Visit www.altisconsulting.com