



CCS switches on AWS to support digital transformation of the construction industry

CCS is a multinational company that offers specialised software solutions for the construction, engineering, and contracting industries. This includes integrated solutions for project management, cost estimation, and lifecycle control.

CCS aims to develop technologies that enhance productivity in the built environment, and that digitise and transform the industry.

Customers have used CCS software to deliver iconic projects, including the Gautrain Rail Link, the Cape Town Stadium, Mall of Africa, and the Sasol head office. Its software is trusted by global industry professionals in over 80 countries, including South Africa, Australia, New Zealand, Europe, UK, and Asia.

Business challenge

CCS's software applications were running on Microsoft Azure, with data stored in a SQL Server database. This was costly, especially when CCS required a new database instance and had to pay additional licensing fees for more storage capacity.

CCS needed to find a cost-effective way to reach more customers in different regions of the world,

which it believed it could achieve by implementing an agile environment that supported quick code deployment and code and process integration.

It partnered with iOCO, who brought valuable expertise, knowledge, and experience in the cloud-based solutions and approaches that would help CCS to scale its reach and operations.



CCS needed to find a cost-effective way to reach more customers in different regions of the world, which it believed it could achieve by implementing an agile environment that supported quick code deployment and code and process integration.

It partnered with iOCO, who brought valuable expertise, knowledge, and experience in the cloud-based solutions and approaches that would help CCS to scale its reach and operations.

iOCO worked with CCS to:

- Streamline business processes through the agile implementation of Amazon Web Services (AWS) cloud-based technologies.
- Reduce the cost of business processes through the ability to rapidly scale its AWS capacity to meet real-time demand.
- Drive business success by designing and implementing the right cloud-based solution.

Through iOCO's strategic guidance, CCS developed an agile roadmap to deliver pay-per-use AWS infrastructure and to design, implement, and deploy a tailored solution in a multi-cloud environment.

Approach

iOCO designed and implemented an AWS architecture stack that moved CCS's entire data management pipeline from Azure to AWS. Amazon Aurora, a MySQL- and PostgreSQL-compatible relational database, was used as a SQL Server replacement.



Built for the cloud, the relational database provides the security, availability, and reliability of commercial databases at a fraction of the cost. It's distributed, fault-tolerant, self-healing, and has the ability to auto-scale, delivering high performance and availability, low latency, point-in-time recovery, continuous backup to Amazon S3, and replication across three Availability Zones.



This technology stack would help CCS to meet its goals to reduce costs and find cost-effective ways to reach more customers in more regions. Upscaling is robust and uncomplicated, with a reduced risk of the database or any part of the AWS stack failing over.



Apart from the deployment of an Amazon Aurora database, iOCO also implemented CloudFront, a multi-region web service that speeds up the distribution of web content to users via a global network of edge locations or data centres. CloudFront allows users to choose what region they would like to connect to, and automatically routes to the correct underlying services.



The AWS architecture supports distributed services across geographic regions, allowing CCS clients to access the software with reduced latency.



The default CCS architecture has five user accounts: the main account, developer account, QA account, Alpha account and Production account. Metrics are generated on each of these accounts based on the services implemented.



iOCO developed a consolidated dashboard on the main account with all the data from the other accounts being channelled through to this dashboard, allowing for the comparison of services running on different accounts. Additionally, green, amber, or red alerts displayed on the main dashboard help DevOps teams to quickly identify problems.

Within each user account, CCS now has the following architecture:

A request for data comes in through the DNS entry service, across to the load balancer, and into the Fargate service. In this use case, the software application code is not stored in Fargate; it is stored in an application container within Fargate. Both the application code and the X-Ray sidecar container interface to the Aurora database.

The request is processed in the application container, which uses ECR docker containers and code deployment, X-Ray, CloudWatch, and SNS services to ensure that monitoring and autoscaling happen simultaneously.

The Availability Zone is split into 3 subnets:

- The public interface, where the request comes in from the DNS entry service to the load balancer
- The business layer that stores the software application code
- The data layer where the Aurora database sits.

The business layer (Fargate) and the data layer (Aurora database and Redis Cache) are private, ensuring that the data is well-protected.

The AWS architecture also has multiple Lambdas running in the background, continuously performing calculations on the database data.

Finally, there is no ETL in this solution. It is a straightforward API exposure with multi-region reach.

Business outcomes

There are three benefits to the implementation of this AWS architecture:



The migration from Azure to AWS reduced CCS's running and scaling costs substantially, increasing the software's user base.



A cost-effective way was found to reach existing and new customers in different regions. Because of AWS's ability to operate across multi-region ecosystems, one of the fundamental benefits is that users can access their data in their home region, from wherever they are in the world.



The development of an agile environment provided the opportunity for quick code deployment and integration as well as straightforward and robust process integration.