Hi, this is Julianne again. Welcome back to our course on Enterprise Database Migration.

Testing is important to ensure your migration is done correctly. Monitoring will help keep your migrated applications and databases running smoothly over time.

Now that you have your SQL Server instance in the cloud, let’s take a look at some things you can do to monitor and test it.

Learning objectives

- Use unit, integration, and regression testing techniques to ensure database migration success.
- Monitor your migration projects with Google tools.

Google Cloud

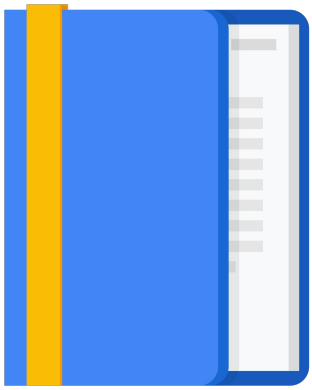
Taking a software development or DevOps approach to migration using unit, integration, and regression testing can ensure a successful migration process.

Aside from testing, There's also a need to -monitor- the entire migration, and Google has tools for that as well.

Agenda

Testing

Monitoring



Google Cloud

Thorough, automated testing is an important part of any IT project. Without effective testing there is no way to know if your database has been migrated correctly. There are many things you need to verify. Was the database schema migrated correctly? Has all the data been migrated? How about user logins? Can all the users still connect and can users only access the data they are permitted to access?

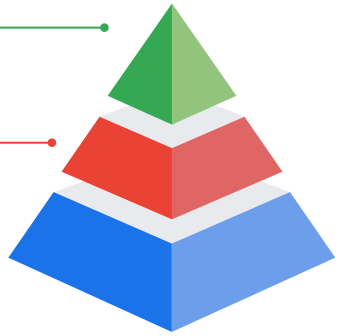
Let’s start off by exploring testing.

Thorough, automated testing can help uncover errors quickly and make migrations go faster

Ensure that the new server environment meets application demand.

Ensure that all applications continue to work with the new database.

Ensure that migration tools and scripts work correctly.

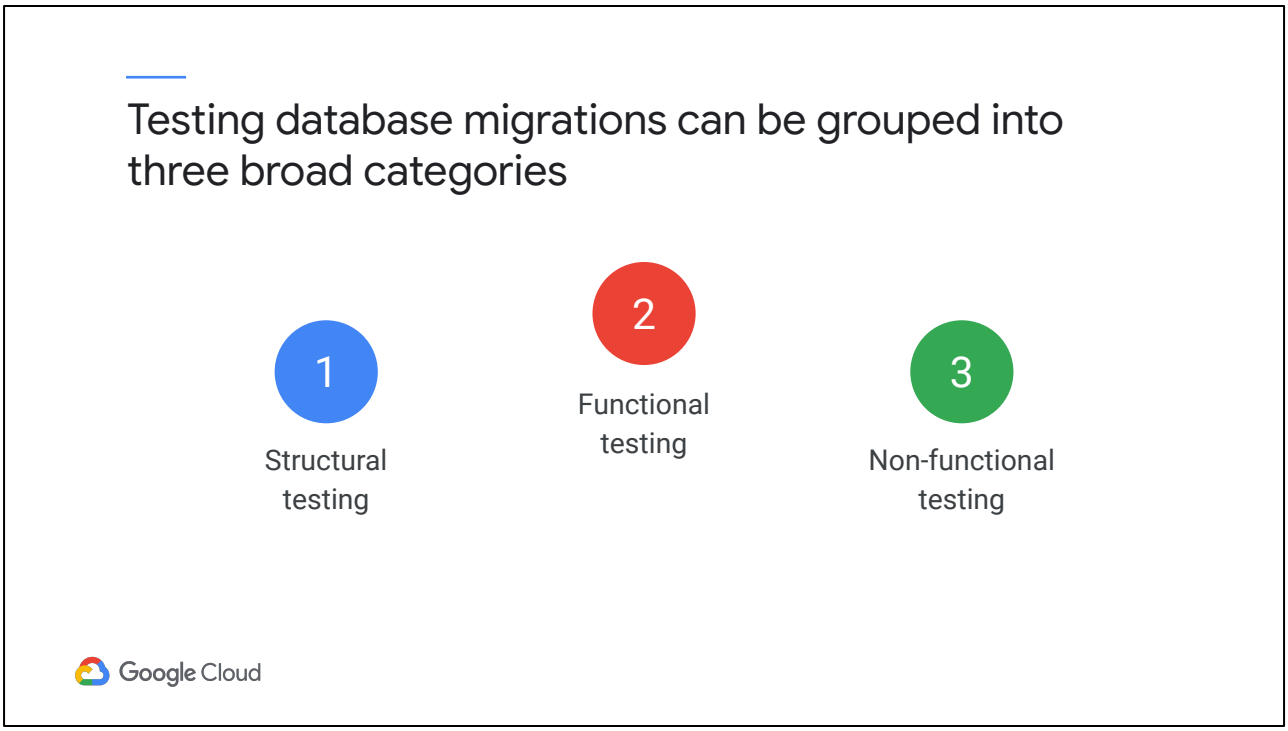


Google Cloud

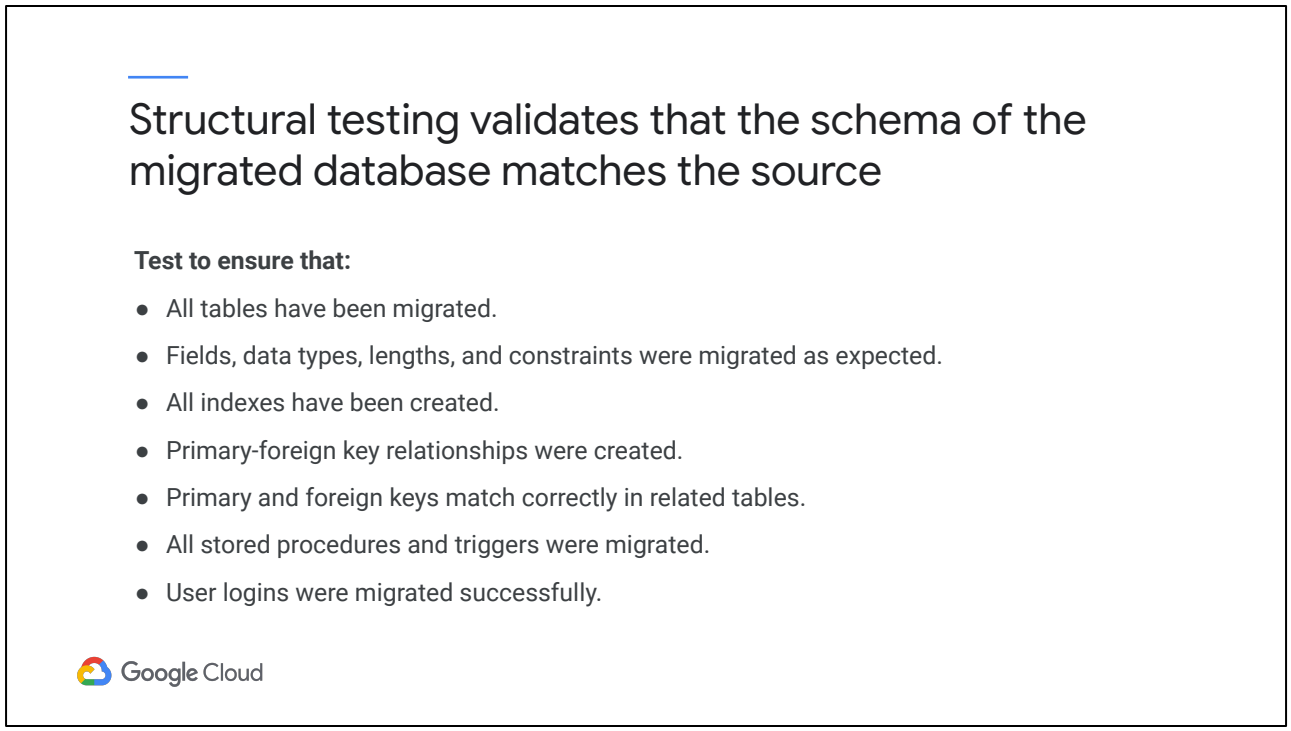
Complete testing is the surest way to facilitate a successful migration and can even speed up the process.

It is critical to ensure that applications continue to work on the new server and that the new server has sufficient resources to meet the demands of the application.

Before starting a long migration process, you want to make sure that the tools and scripts work properly.



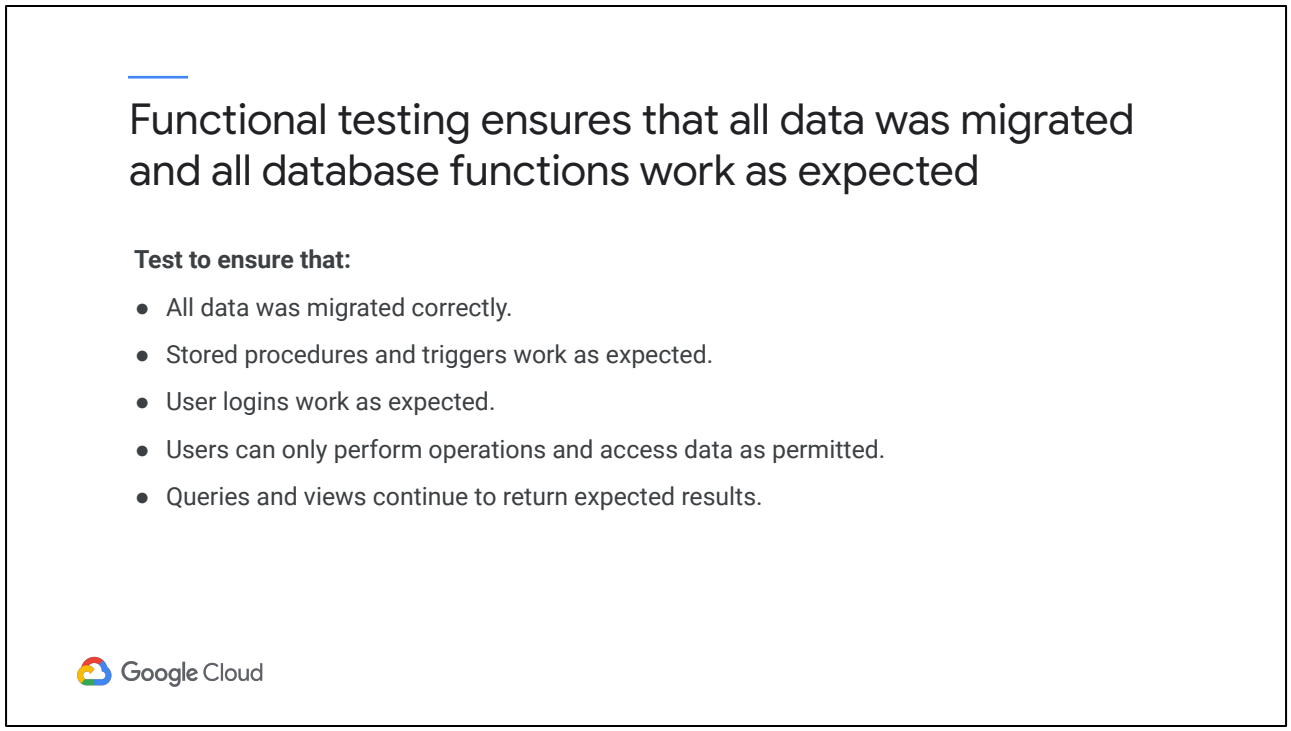
There are basically three categories of testing that need to be considered: structural, functional, and non-functional.



Structural testing validates that everything that should have been moved has successfully been moved. It checks to make sure that every table, procedure, index, foreign key, trigger, login, and more has been brought over and that the new environment is structurally the same as the original.

Write automated structural tests to ensure that:

- All tables have been migrated.
- Fields, data types, lengths, and constraints were migrated as expected.
- All indexes have been created.
- Primary-foreign key relationships were created.
- Primary and foreign keys match correctly in related tables.
- All stored procedures and triggers were migrated.
- And lastly that User logins were migrated successfully.



Functional testing goes to the next level and makes sure that all of the data has come across correctly. It also makes sure that all of the views and stored procedures work as intended and that users can log in and see what they are supposed to, but not what they shouldn't.

Write automated functional tests to ensure that:

- All data was migrated correctly.
- Stored procedures and triggers work as expected.
- User logins work as expected.
- Users can only perform operations and access data as permitted.
- And that Queries and views continue to return expected results.

Non-functional testing ensures that the new database system meets its business requirements

Types of non-functional tests include:

- Performance testing to ensure that queries and transactions complete quickly enough to meet user expectations
- Load testing to ensure that the application performs at peak demand
- Stress testing to find the load at which the database breaks

Google Cloud

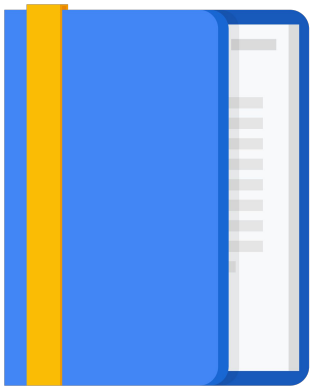
After you know that everything works as intended, the next step is to make sure it performs well and can withstand peak usage periods. Load and stress testing will push the server to its limit to see whether it can handle the intended workload and just how far it will go before it breaks.

This helps determine whether more resources need to be allocated to the server.

Agenda

Testing

Monitoring



Google Cloud

After a functional server is up and running, you want to monitor its activity to make sure there are no surprises.

Cloud Operations monitoring collects and displays information about your resources

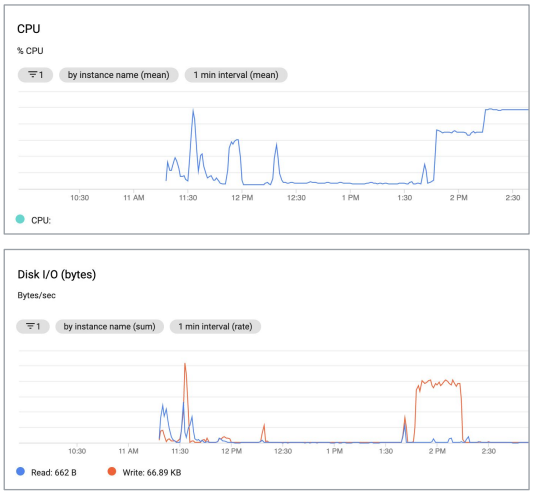
- Default monitoring is built into the web-based console for every service.
- You can create additional dashboards with custom charts.
- You can monitor resources in both AWS and Google Cloud.

Google Cloud

Google provides a web-based monitoring tool with a default console for each service. You can create your own custom dashboards and monitor resources in both Google Cloud and AWS.

Compute Engine VMs offer basic monitoring by default

- Metrics include:
  - CPU usage
  - Network traffic
  - Disk I/O

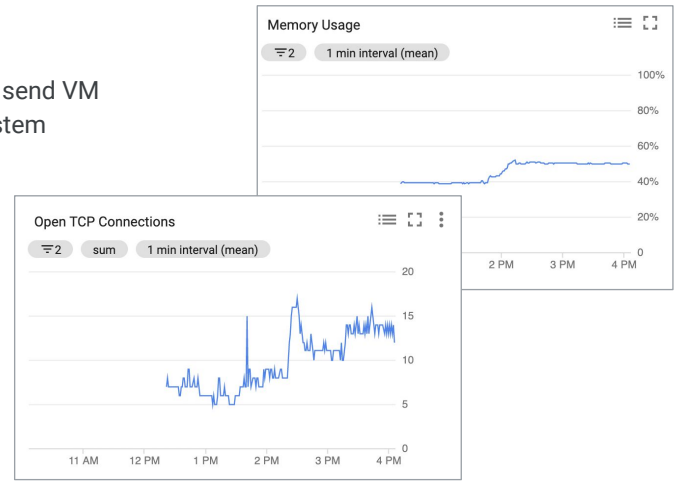


Google Cloud

There is basic VM monitoring by default, but that just shows “big-picture” stats such as CPU usage, network traffic, and Disk I/O.

Install the Monitoring Agent to monitor additional metrics in a VM

- Uses a `collectd` daemon to send VM metrics to the Monitoring system
- Additional metrics include:
  - Memory usage
  - Swap usage
  - TCP connections



Google Cloud

To really understand what's going on inside of a database server, you typically need to look at more metrics such as memory usage and swap file usage. You can install the Monitoring Agent to collect those metrics from the VM and send them to the Monitoring system.

The Monitoring Agent uses a "collectd" daemon to send VM metrics to the Monitoring system.

The Monitoring Agent also monitors applications on virtual machines

- Many applications are supported on Linux, including:
  - Apache Web server
  - MySQL
  - Nginx
  - JVM
- On Windows, the monitoring agent supports IIS and SQL Server

- SQL Server metrics include:
  - User connections
  - Transactions
  - Write transactions

Google Cloud

The Monitoring Agent is flexible enough to monitor a variety of applications such as MySQL, Apache web server, and JVM. When you are running a Windows VM, it can monitor IIS and SQL Server.

This allows you to tap into the SQL Server-specific metrics usually provided with the SQL Server Performance Monitor, such as number of SQL connections, transactions, and cache hit ratio.

On Windows, download and install the Monitoring Agent using Powershell

```
cd $env:UserProfile;
(New-Object
Net.WebClient).DownloadFile("https://repo.stackdriver.com/windows/StackdriverMonitoring-GCM-46.exe", ".\StackdriverMonitoring-GCM-46.exe")
.\StackdriverMonitoring-GCM-46.exe
```

Google Cloud

This Powershell command downloads and installs the Monitoring Agent on Windows. You can run it after logging on to the machine, or better still, make it part of the machine's startup script.

For more information, search Google Cloud documentation for the Monitoring agent.

From the documentation you can copy and paste the script shown.

On Linux, download and install the Monitoring Agent using Bash

```
curl -sSO https://dl.google.com/cloudagents/add-monitoring-agent-repo.sh
sudo bash add-monitoring-agent-repo.sh
sudo apt-get update
sudo apt-get install stackdriver-agent
sudo service stackdriver-agent start
```

Google Cloud

If you're running SQL Server on Linux instead, the bash commands shown here will install the monitoring agent. Make these commands part of your machine startup script.

Again, see the documentation to copy-paste the code.

Logging allows you to store, search, analyze, monitor, and alert on log data

- Log data is collected and stored for each service.
- Logs can be filtered for analysis
- Log data can be exported to Cloud Storage, BigQuery, or Pub/Sub.
- Log metrics can be displayed in custom charts.
- You can create alerts from log metrics.

Google Cloud

The agents collect the data from the OS and the SQL Server and store it in the Google Cloud logging service. Once there, it can be searched, filtered, and analyzed. You can export the log data to Cloud Storage, BigQuery, or Pub/Sub and even create alerts if the metrics meet a certain condition. Of course, you can also display custom charts of the metrics to visualize the data. You can even create log metrics and trigger alerts when there are anomalies in the log stream.

Install the Logging Agent to monitor additional metrics in a VM

- Based on `fluentd`
- Streams logs from third-party applications and system software
- Linux and Windows

Google Cloud

You should install the Logging Agent if you are running your database on a virtual machine. The Logging Agent is based on fluentd and captures machine logs and the logs of third-party software, including SQL Server.

This works on both Windows and Linux.

On Windows, install the Logging Agent using Powershell

```
cd $env:UserProfile;
(New-Object
Net.WebClient).DownloadFile("https://dl.google.com/cloudagents/windows/Sta
ckdriverLogging-v1-11.exe", ".\StackdriverLogging-v1-11.exe")
.\StackdriverLogging-v1-11.exe
```

Here's the Powershell script to install the Logging Agent on Windows.

For more information, search Google Cloud documentation for the Logging agent.

On Linux, install the Logging Agent using Bash

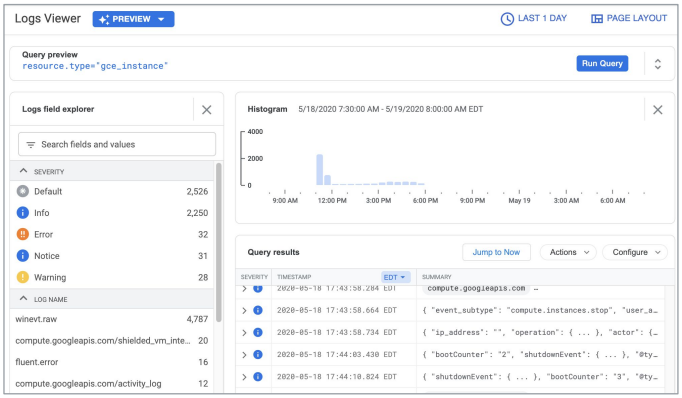
```
curl -sSO https://dl.google.com/cloudagents/add-logging-agent-repo.sh
sudo bash add-logging-agent-repo.sh
sudo apt-get update
sudo apt-get install google-fluentd
sudo service google-fluentd start
```

Google Cloud

Of course, you can also install the logging agent on Linux. Shown here are the equivalent bash commands. As before, copy and paste the code from the documentation.

Use the Logs Viewer to filter and analyze the logs

- Filter logs by service, resource, date/time range, severity, etc.
- Export logs to external sink (Cloud Storage, BigQuery, Pub/Sub).

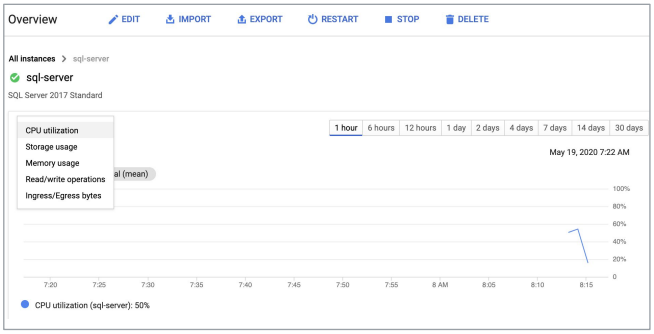


Google Cloud

The Logs Viewer provides a convenient web-based interface to filter the logged data, or you can export it and explore it using any other tools. To export logs, create a filter that collects the log information you are interested in, and then create an export sink. The export destination can be Cloud Storage, if you just want to save the log data, or BigQuery, if you want to use SQL for log analysis; or send the data to Pub/Sub for real-time log analysis.

Cloud SQL provides monitoring by default

- No need to install the agents
- Metrics include:
  - CPU utilization
  - Storage usage
  - Memory usage
  - Read/write operations
  - Ingress/egress bytes
- Available from the Console and Monitor Dashboards



Google Cloud

If you use Cloud SQL instead of building your own VMs, you get monitoring by default.

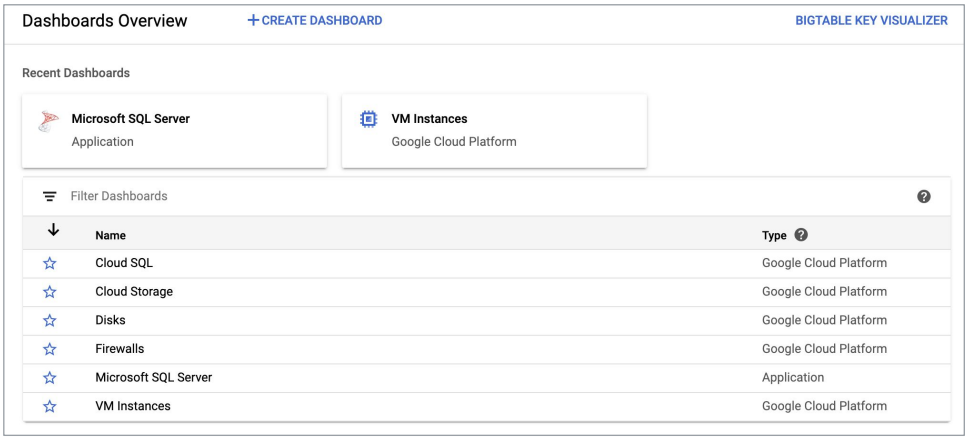
There's no need to install any agents. You get multiple standard metrics and can view this from the Console and the Monitor Dashboards.

The built-in metrics include:

CPU utilization
Storage usage
Memory usage
Read/write operations
and both ingress and egress bytes

These are the metrics you would probably want to monitor when managing a database.

The Monitoring system automatically creates dashboards for resources in your projects



Google Cloud

The Monitoring system automatically creates dashboards for resources in your projects. It analyzes what resources have been created in a project and builds dashboards accordingly. This is a great way to get started. Often, the hardest part of monitoring is figuring out what you want to monitor.

After you get some experience, you can take this a step further by creating your own custom dashboards.

Module review

Google Cloud

In this module, you learned to use automated unit, integration, and regression testing techniques to ensure database migration success, and monitor your migration projects with Google tools.

Google Cloud