Ganymede Architecture

Overview

This page presents the high-level architecture of Ganymede.

Requirements

Ganymede requires:

- Database access to the Cloud Control database in all locations where data is to be retrieved from.
- A server for running the Ganymede API.
- A server for running the Ganymede database.
- A server for holding data files sent from the locations.
- A server/cluster for running MongoDB.
- A server for running the Ganymede daemon.
- A server, provided by Glassbeam, for transferring data files on a periodic basis and loaded into the Glassbeam reporting application.

⚠ A single server may fulfill multiple requirements, where necessary or practical. For example, a single server may run the Gaymede daemon and act as the central location for holding all the Cloud Control database dump files.

Components

Ganymede has several moving parts. Each component that makes up the product is described below.

Agents

An agent is a script that gets run once an hour on a server in a GEO. This script is responsible for the following actions:

1. Initializing a transaction in the Ganymede system.
2. Extracting data from the Cloud Control database.
3. Compressing and encrypting the extracted data.
4. Transferring the data file to a central location.
5. Signaling the end of the transaction.

The agent will send periodic status messages to the Ganymede system so operators can monitor the progress of the activity.

All communication with the Ganymede system is accomplished via an API.

Jupiter

Jupiter is a long-running process that coordinates actions, based on updates sent from the agents. Actionable events trigger the launching of a worker process. These worker processes include:

1. Decrypting and uncompressing a data file.
2. Loading a data file into a temporary MySQL schema.
3. Performing extract/transform on the data in the temporary schema and loading it into MongoDB.
4. Removing the temporary schema and data file(s) after the ETL process is complete.
5. Generating a data feed for transfer to Glassbeam.
6. Transferring the data file(s) to Glassbeam.

Activities 1-4 happen for each GEO configured. Activities 5 & 6 happen once data for all GEOs has been loaded into MongoDB. Jupiter utilizes the Ganymede database and the MongoDB warehouse and has no communication with any other outside resources.

Workers

A worker is a short-lived task that accomplishes a single step in the process. Tasks that workers perform are listed under Jupiter. Workers utilize the Ganymede database and MongoDB, and Glassbeam's transfer server. There is no other communications with other outside resources.

Archive

An archive process runs independent of the Agents, Workers and Jupiter once an hour. It's function is to dump the Ganymede schema and then copy all MySQL dumps (the Ganymede schema dump and the dumps sent by the Agents) to the Cloud Files service as an off-server backup of Ganymede. The Cloud Files service provides long-term storage of all archives, enabling the restore of any period of data in the event of disaster.

Ganymede Database

The Ganymede database is a MySQL instance (5.6 or greater) that is used for holding configuration data of Ganymede components, a transaction log of the agents and workers activities, temporary schemas for loading Cloud Control data during the transform and load tasks.

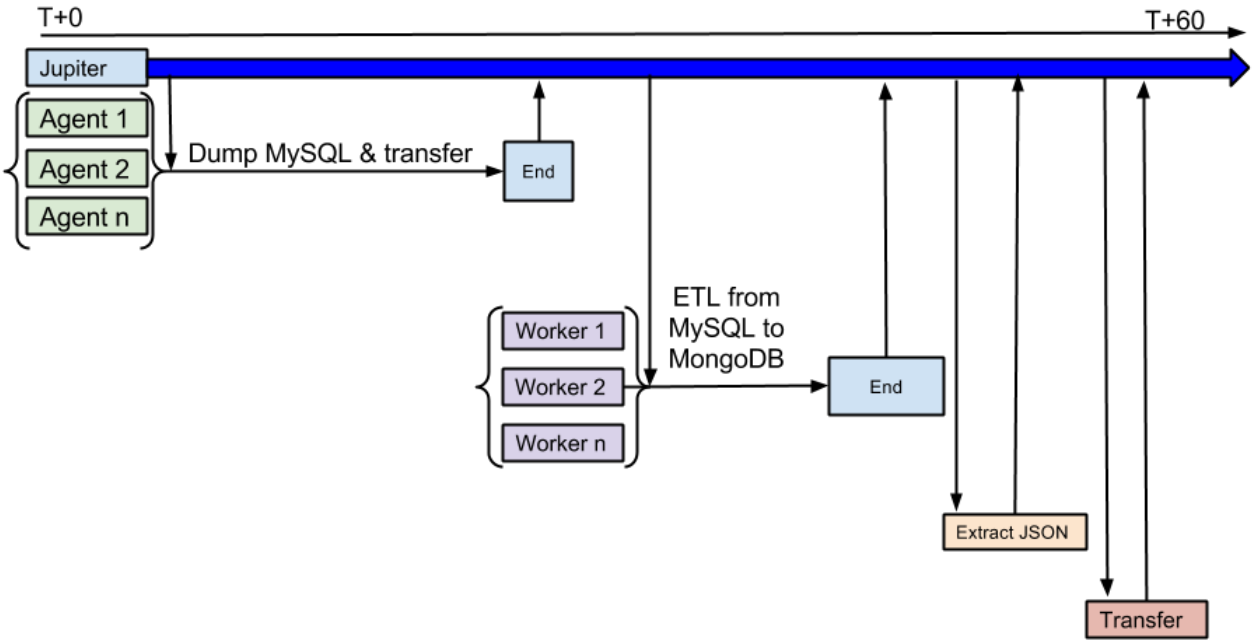
MongoDB

The MongoDB cluster is used for long-term storage of Cloud Control data.

API

The API is a web application that implements an API for managing Ganymede configuration data, getting transaction log information and for the Ganymede components to report status information.

Process Flow


T+0
T+60
Jupiter
Agent 1
Agent 2
Agent n
Dump MySQL & transfer
End
Worker 1
Worker 2
Worker n
ETL from MySQL to MongoDB
End
Extract JSON
Transfer