Architecture Overview

The GeoSystems Monitor architecture was specifically designed for ease of deployment and to allow customization to fit your unique monitoring needs. The core architecture elements include:

- Service architecture – GeoMonitor provides real-time monitoring for the growing service-based architecture that is present in today's GIS.

- Agentless – all monitoring occurs without requiring the installation of agents on monitored servers or services. This provides a monitoring solution that is easy to implement and manage over time.

- Customized monitoring – GeoMonitor includes an extensive library of pre-configured monitoring end points. Custom monitoring scripts can be developed using Python® (the core programming language used in GeoMonitor checks), in addition to other languages to fit unique monitoring needs.

- Flexible interface – The GeoMonitor dashboard interface is designed to allow user-defined organization with tabbed windows and "containers", which are user-specified groups of monitoring end points.

The GeoMonitor system architecture is divided into three main components:

1. A web-based **User Interface**, including:

- The main Dashboard, which provides a centralized view of GeoMonitor Servers and the monitoring "points" (the individual services, applications, etc. that are monitored within GeoMonitor). The Dashboard is a HTML5 and JavaScript libraries.-based web application.

- The Editor, designed for modifying existing monitoring points, with a focus on performing batch editing. The editor is developed with HTML5 and JavaScript libraries.

- Charts and Reports that provide visual summarizations of the monitoring information that is collected and stored within the GeoMonitor application. The charts and reports are developed with HTML5, and JavaScript libraries.

2. The GeoMonitor **Engine**: A Windows® Service that executes the monitoring point scripts. The scripts are executable functions that communicate with the monitoring points, submitting requests, analyzing the results, and triggering alerts when issues occur. The scripts are developed in Python.

3. The GeoMonitor **Databases**, comprised of:

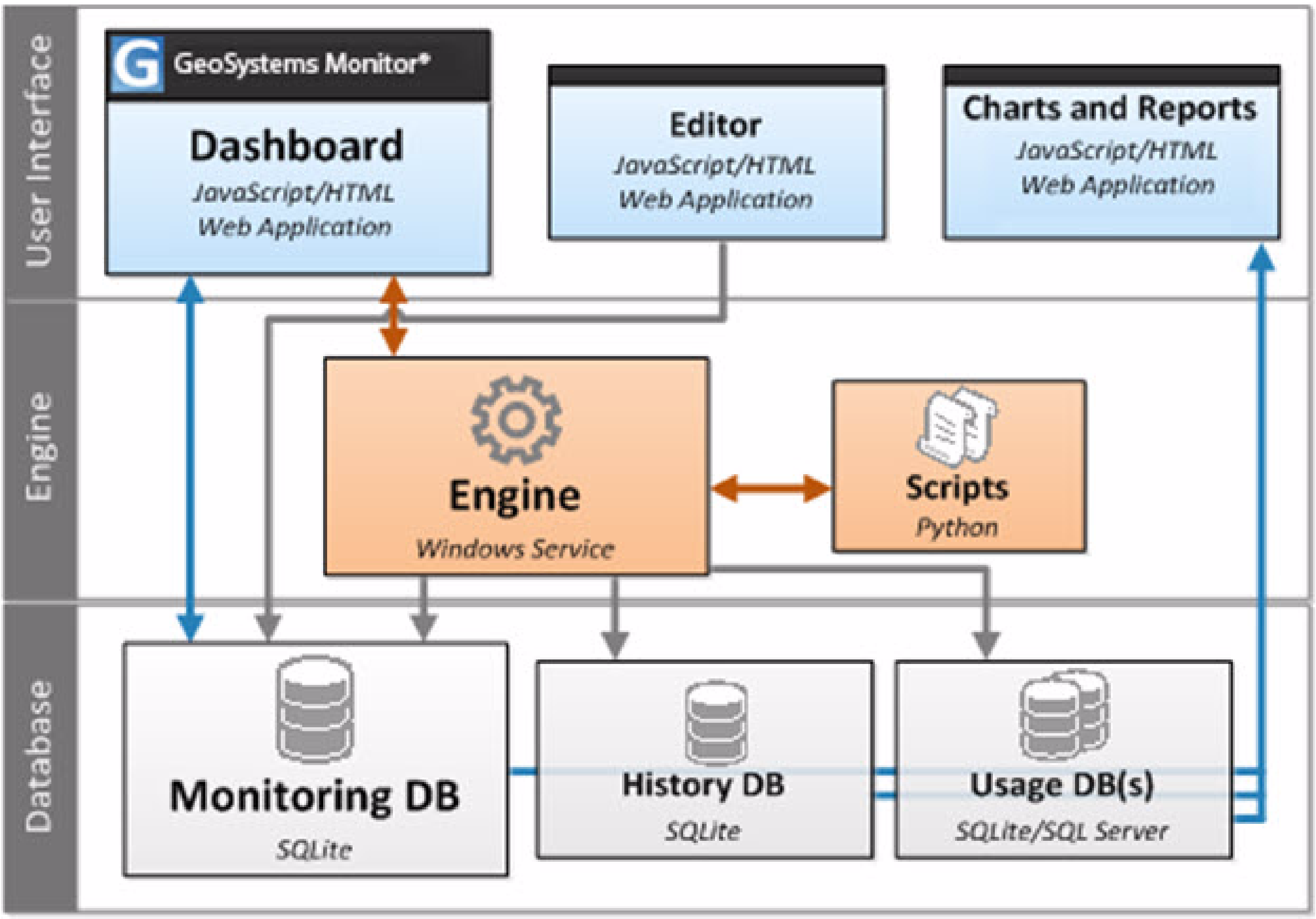
- The Monitoring Database, the main database for the application that stores both the GeoMonitor system configuration, and results of monitoring checks. This is a SQLite format database.

- A History Database that stores the cumulative history of monitoring checks performance for use in the Charts and Reports. This is a SQLite format database.

- Usage Database, one or more databases that store usage information. Note that this is associated with an optional feature that stores ArcSDE® and ArcGIS® Server usage information for reporting. Usage Databases can exist within a SQLite or Microsoft SQL

Server® (Express or Standard/Business Intelligence/Enterprise editions). Note that due to the volume of data that is generated in a typical usage database, it is recommended to utilize SQL Server databases instead of SQLite. The Usage database by default utilizes SQLite; SQL Server may be more appropriate if a large amount of data is generated.

Diagram of the GeoSystems Monitor Architecture and interactions between the components:



Article ID: 344

Last updated: 26 Dec, 2019

Revision: 1

GeoSystems Monitor Enterprise -> Product Guide v4.1 - 4.2.1 -> Introduction -> Architecture Overview

http://www.vestra-docs.com/index.php?View=entry&EntryID=344