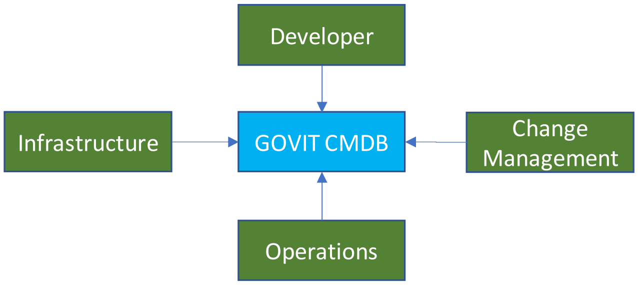
GOVIT ApS
Kalvehavevej 10
DK-3250 Gilleleje
CVR nr. 39 14 54 80
Phone: +45 26 30 40 40
Mail: info@govit.dk
Web: govit.dk

Contents

What is GOVIT?

GOVIT is a modular, centralized tool for enterprise IT organizations that assist with configuration management, automated provisioning, monitoring and operations of the business-critical applications across all environments.

GOVIT supports a process that reduces the gap between developers, operations, change management and infrastructure while preserving all enterprise compliance rules on auditability and segregation of duties.



GOVIT will reduce the enterprise costs for application provisioning, operations and monitoring by standardizing the interface for all these operations.

GOVIT and ITIL

GOVIT fits very well into the ITIL foundation Service Management as it provides an Active Configuration Management component that interacts with other Configuration Management systems in the enterprise.

GOVIT and Developers

GOVIT is centered around the developers having full access to perform all operations on their own projects in their own environments. Access to projects can be restricted to certain groups as needed to separate access.

Inside GOVIT, the developer performs configuration changes such as allocating servers, access to data bases and storing configuration files and properties.

The developer then performs changes in the source code, commits changes and performs continuous deployments / continuous delivery in their own environments by a simple integration to a GOVIT service API.

When the application is ready for next environment, GOVIT assists by promoting the configuration from one environment to the next to maintain a high similarity between environments. In most cases, the only differences between environments are end points and scalability.

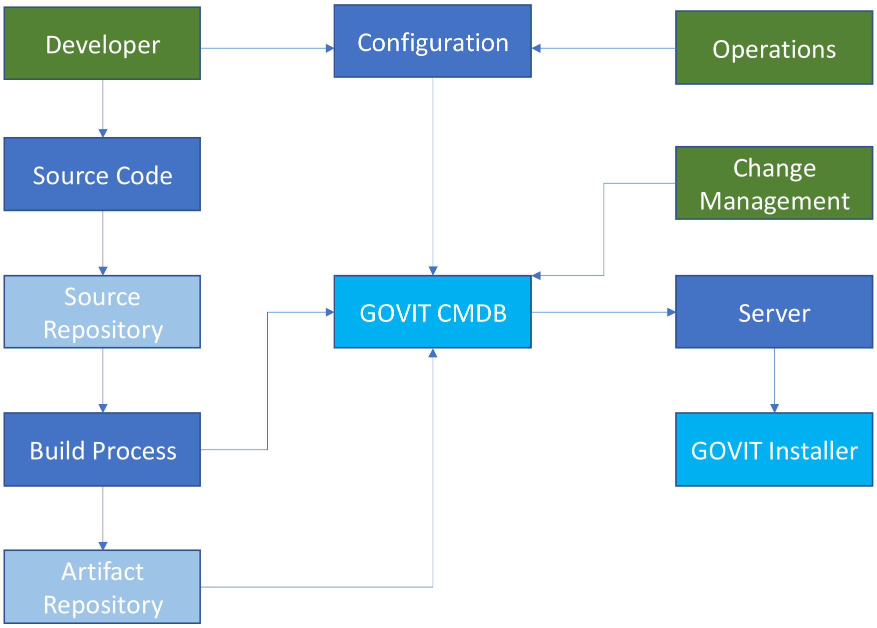
GOVIT keeps full track of the configuration versions in the database to allow for full auditing, comparison and rollback of a configuration to previously working version.

The audit trail also includes which application versions that have been deployed to which environment, when the deployment took place and by whom.

GOVIT endorses separation of the application code and the environment specific configuration, leaving the developer to focus on delivering the code once, and having GOVIT control all the environment differences.

This will allow GOVIT to support the principle that access to critical production resources such as databases is only provided by authorized persons such as operations or DBAs.

The process looks like



1. The developer works on source code that is stored in a source code repository
2. A build process reads the source code, builds, perform unit tests and stores the build result binaries in an artifact repository
3. The developer and/or operations updates the configuration in GOVIT including allocating servers, virtual machines, ports and additional detailed configuration
4. GOVIT deploys the entire application with configuration and application binaries by pushing all files to the relevant servers and invoking an installer

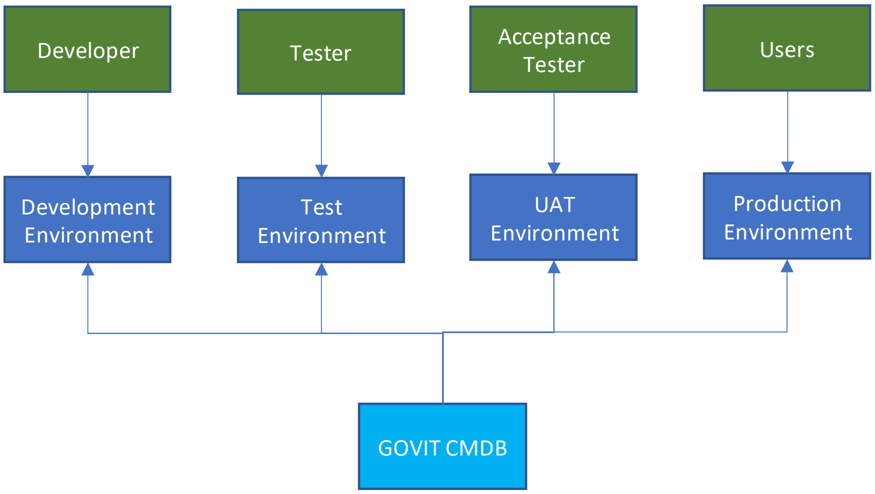
5. The Installer is a component specific to the target middleware product that build the application environment and deploys the project binaries.
   Examples of middleware products are Oracle Weblogic and Apache Tomcat

The deployment process is thus fully automated, where all target environment changes are tracked to the GOVIT based configuration. This allows for highly reliable and cost-effective deployment processes and very simple disaster procedures.

The deployments can be triggered as part of the build process, from the user interface or from the enterprise change management system. All this is performed with full audit and trace logs and always confirming to the enterprise governance rules with segregation of duties and four-eye principles.

GOVIT and Testers

GOVIT supports the concept of separating environments for different purposes and with different accesses.



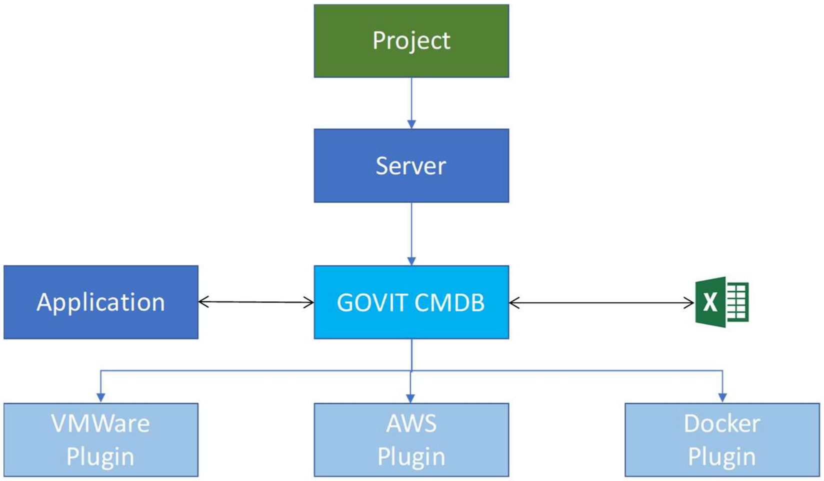
A typical scenario as

1. The developer has full access to the development environments, where initial development and testing is performed
2. The testers work with test plans and test cases that depend on the test manager having control of which application versions are deployed to their environments and if the applications are running satisfactory. Integration testing is usually performed here
3. When integration testing has completed, operations and infrastructure may require a separate User Acceptance Test environment to validate changes in the infrastructure such as server patches, network, firewalls and databases.
   In many cases, this environment is production-like in terms of scalability making this the right environment for load testing.
4. When user acceptance test or a load test has been accepted, the application is promoted to separate production environment.

GOVIT fully supports this process with full audit logs, version logs, activity logging, change logs and allowing different users to operate on different environments with the same processes.

GOVIT and Infrastructure

A key element of GOVIT is a central register of servers, their location in the network and data centers, and which servers are using which servers for which purpose.



GOVIT easily imports from and exports to Excel files making it very easy to register all existing servers as a base for registering their usage for the applications and projects.

Existing applications can also be reverse-engineered from their current definitions and imported into GOVIT making it easier to start on-boarding applications to GOVIT.

GOVIT can integrate with current and future infrastructure elements such as VMWare Hypervisor, AWS Cloud, Microsoft Azure and OpenShift / Docker and hiding infrastructure differences from the users. The developers and operations should focus on applications and less on infrastructure details.

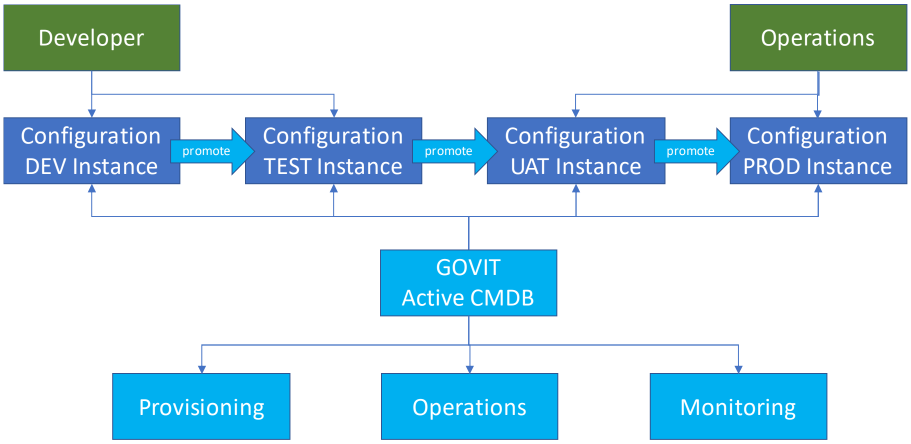
GOVIT can also integrate with resource acquisition, so when a project requires new servers, GOVIT relays the request to the appropriate provider, creates the servers in the CMDB, relates them to the project once they have been provisioned.

GOVIT can also install and upgrade base software packages such as JDK and middleware products on relevant servers. This will allow for a close relation between the projects and their required product versions, also automating software rollout and patching.

It will also allow a centralized governance of current software levels to ensure the proper security patches have been tested and rolled out with full project integration.

GOVIT and Operations

The GOVIT concept makes it easy and cost effective to move applications and their configuration to a new environment as it is typically only new servers, endpoints and passwords that require adjustment. All other details are the same as in previous environment.


Developer
Operations
Configuration DEV Instance
promote
Configuration TEST Instance
promote
Configuration UAT Instance
promote
Configuration PROD Instance
GOVIT Active CMDB
Provisioning
Operations
Monitoring

The GOVIT CMDB thus contains the entire configuration from all environments, and it the user's roles that determine what can be changed in which environment.

GOVIT makes it easy and transparent to provision application changes, monitor the running application and perform required operational steps such as restarting application components and retrieving a log file from a running application. All this is performed regardless of the underlying technologies such as server operating system, network location and middleware product.

GOVIT can integrate with the enterprise change management and incident system to fetch or raise changes, incidents and scheduled downtime events. GOVIT also automatically puts a project in planned downtime status when a deployment or an operation is performed to keep unwanted errors from being propagated to the enterprise operations dashboard.

This makes GOVIT the principal component for operations across all projects, environments and platforms.

GOVIT also supports DevOps, where different operations groups can be assigned to different projects. This is done by assigning users to either global or project level roles

GOVIT and Technologies

GOVIT supports all server based technologies that can be configured using scripts, command lines or APIs.

Examples of products are:

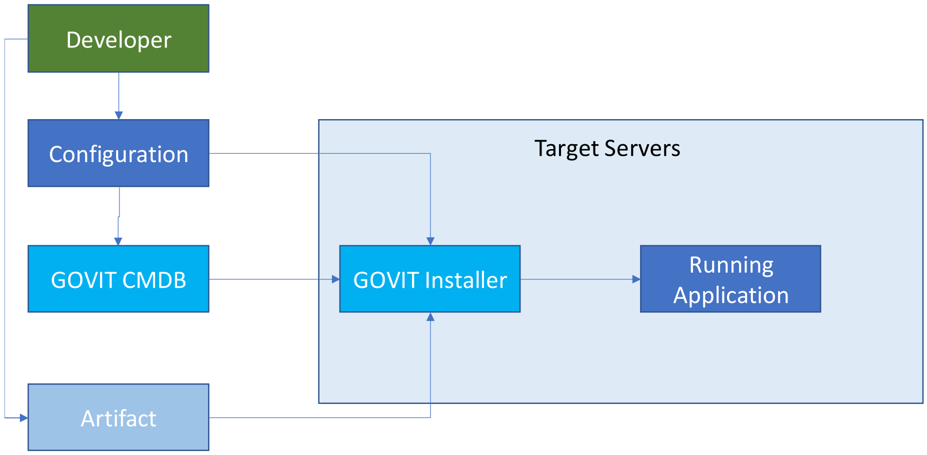
- Oracle WebLogic
- Oracle SOA Suite
- IBM WebSphere Application Server
- RedHat JBOSS
- Apache Tomcat
- SpringBoot Microservices
- .Net Server Applications

GOVIT is based on a shared configuration model applicable to most products that hold elements as servers, Virtual Machines, Clusters, Data Sources, Deployments, files and security elements. In addition, product specific configuration elements can be added

To support a specific platform, the following components are required:

- A product specific installer that creates the server configuration and deployments the applications.
- A product specific validation component that verifies that the central configuration is consistent
- Product specific configuration elements such as Weblogic JMS definitions.

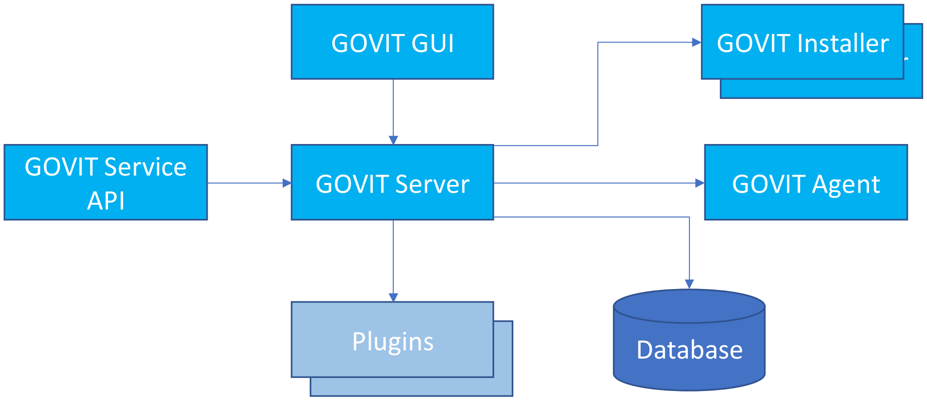
The process is:



The GOVIT Installer is the component that is launched from GOVIT, reads the configuration, creates the application context and deploys the applications.

GOVIT Components

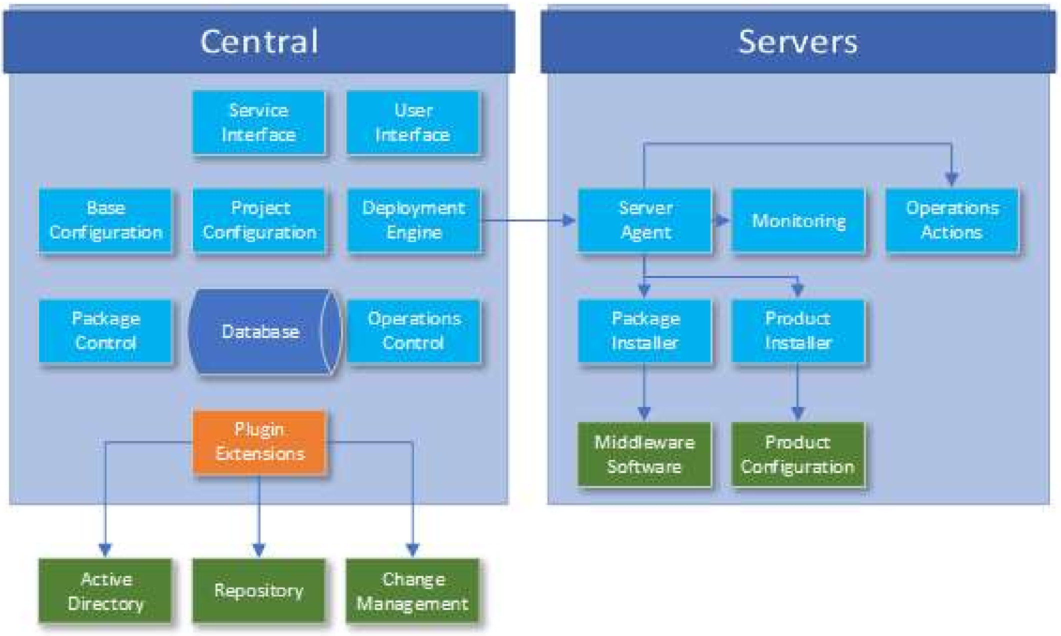
GOVIT comprises of the following components:



- A database is required, either Oracle or Mysql
- The central GOVIT server is a JEE based application that can deployed to all JEE servers such as Apache Tomcat and Oracle Weblogic where the database access is configured.
  The actual setup depends on the scalability and availability requirements.
- The GUI is the browser based interface to the GOVIT server, and can be used to all operations and configurations.
  The GUI can be deployed with the GOVIT server and security can be AD integrated and work with single sign-on products.
- The GOVIT Service API can be used from applications to GOVIT ranging from the continuous deployment interface to full access to all configuration elements.
- Plugins can be created and configured to modify the default behavior and to perform customer specific integrations to customer specific end points such as repositories, change management, server providers and background tasks
- The GOVIT agent is used to perform remote execution during deployments and operations and to perform monitoring of the server and the server based applications
- GOVIT installers are created for each product as described above

GOVIT Modules

GOVIT can be adjusted to each enterprise requirements by selecting the GOVIT components as:



Module	Contents
Core	Core configuration of infrastructure, projects and environments, including the GUI The core component can be launched on a cloud solution for an immediate start. When the required configuration has been loaded, the configuration can be transferred to the customer's infrastructure or to a cloud solution as needed.
Provisioning	Provisioning is the automatic configuration and deployment of customer specific applications. Access to the enterprise infrastructure is required.
Agent	The agent is used for remote execution during provisioning and operations, and is required for Monitoring
Service API	The Service API comes in two parts • Access to continuous deployment from eg. Build servers • Full access to the entire CMDB from external systems
AD Integration	AD Integration can be selected to move the user administration to the central AD, including users, passwords, group memberships and E-mail adresses
Monitoring	Since the GOVIT CMDB knows the applications and environments, monitoring of all these elements can be created with no additional steps. Monitoring routes alerts to email and configured custom management systems.
Operations	Operations is supported by allowing stop/starts of application components such as virtual machines and clusters as well as retrieving remote log files. All of this is done from the central GUI regardless of server network location

Product support	Specific products require a dedicated installer component, possibly plugins and configuration items
Base Software	GOVIT can handle the roll-out and patching of middleware software packages specific to each project

In addition to the standard modules, specific requirements are handled by developing customer specific plugins.

GOVIT FAQ

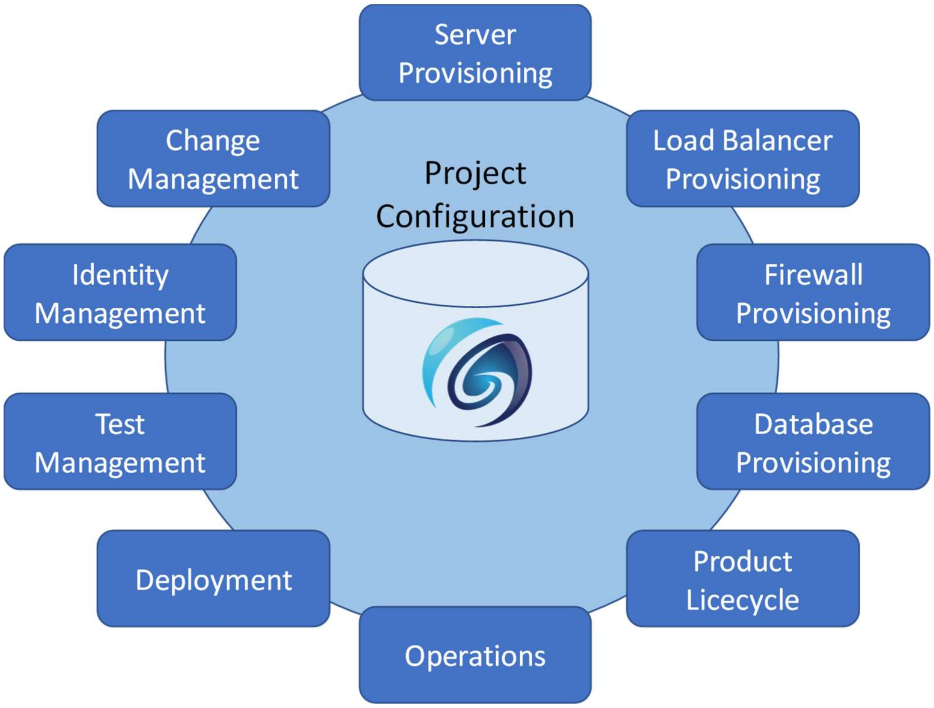
What about

GOVIT and	Answer
DevOps	GOVIT supports full DevOps where roles and accesses can be configured freely. Access can be delegated to specific projects or globally. It is only a matter of enterprise configuration that determines what rules and validations that must be enforced.
Ansible	GOVIT differs from the playbook concept by focusing on storing application specific configuration in the database and letting the installer perform the required actions based on this configuration. The configuration is centrally stored and accessible and is easily transferred to next environment. If needed, the installer can be defined as dynamically configured Anslible playbook
Puppet	Puppet is focused on base OS configuration and base software delivery. GOVIT builds on top of this to configure application containers and deploying project artifacts without the need of creating a new package each time.
System Center	Much of the same considerations as Puppet
Cloud	GOVIT easily support cloud vendors such as AWS by configuring an AWS specific plugin to mask the vendor specific details. GOVIT can also run as a cloud solution
Containers	GOVIT can deploy to container based products such as Docker. Application deployable artifacts can easily be Docker images where the configuration and governance rules are performed by GOVIT.
Change Management	GOVIT is not a change management system. GOVIT can integrate with the enterprise change and incident management system to retrieve, update and react on changes and incidents.
Other CMDBs	In the ITIL context, several CMDBs can exist as long they interact with a clearly defined ownership of which Configuration Items. GOVIT can integrate with other CMDBs to fetch server information or to propagate GOVIT owned information such as port assignments, database schema usage and end points

GOVIT Vision

The GOVIT vision does not end here.

Since GOVIT is an active CMDB with detailed knowledge of the applications and their surroundings, it makes it an ideal candidate for further automation and self-service.



You can imagine GOVIT integrating into provisioning of areas such as

- User management
- Changes
- Server provisioning
- Load balancer provisioning
- Key and certificate management
- Firewall provisioning
- Database provisioning
- Database migration
- Middleware product lifecycle
- Operations
- Deployments
- Test management

GOVIT History

GOVIT is based on many years experiences with development, deployment and operations of complex, business critical application and using best practices from all these areas.

The GOVIT product is launched in 2017, but the product design, technology selecting is based on the developer's solid experience with the product areas.

Contact

Mail us at info@govit.dk

Call us at +45 26 30 40 40

See also at linkedin:

www.linkedin.com/in/niels-gylling-3934111

www.linkedin.com/company/govit/