Oracle® Fusion Middleware

Understanding Oracle GoldenGate

12*c* (12.3.0.1)
E80974-02
December 2017

Oracle Fusion Middleware Understanding Oracle GoldenGate, 12*c* (12.3.0.1)

E80974-02

Contents

Preface

Audience v
Documentation Accessibility v
Conventions v
Related Information vi

1 Introduction to Oracle GoldenGate

1.1 What is Oracle GoldenGate? 1-1
1.2 Why Do You Need Oracle GoldenGate? 1-1
1.3 When Do You Use Oracle GoldenGate? 1-2
1.4 How Do You Use Oracle GoldenGate? 1-3
1.5 Oracle GoldenGate Product Family 1-3

2 Getting Started with Oracle GoldenGate Microservices Architecture

2.1 Oracle GoldenGate Architectures Overview 2-1
2.2 Common Data Replication Processes 2-3
2.2.1 What is an Extract? 2-3
2.2.2 What is a Trail? 2-4
2.2.3 What is a Replicat? 2-5
2.3 Oracle GoldenGate Key Terms and Concepts 2-6
2.4 Components of Oracle GoldenGate Classic Architecture 2-8
2.4.1 What is a Data Pump? 2-10
2.4.2 What is a Collector? 2-11
2.4.3 What is a Manager? 2-11
2.4.4 What is GGSCI? 2-12
2.5 Components of Oracle GoldenGate Microservices Architecture 2-12
2.5.1 What is a Service Manager? 2-15
2.5.2 What is an Administration Server? 2-15
2.5.3 What is a Receiver Server? 2-16
2.5.4 What is a Distribution Server? 2-18
2.5.5 What is a Performance Metrics Server? 2-19

2.5.6 What is the Admin Client? 2-19
2.5.7 Microservices Architecture (MA) Security, Authentication, and Authorization 2-20
2.5.7.1 Microservices Architecture Authentication 2-21
2.5.7.2 Microservices Architecture Security 2-23
2.5.8 What are the Key Microservices Architecture Directories and Variables? 2-26

3 Roadmap for Implementing the Microservices Architecture

Preface

Understanding Oracle GoldenGate12*c* (12.3.0.1) describes data replication concepts, Oracle GoldenGate Classic Architecture, Oracle GoldenGate Microservices Architecture, and the architecture components. Using these concepts you can implement a data replication solution using Oracle GoldenGate.

- Audience
- Documentation Accessibility
- Conventions
- Related Information

Audience

This guide is intended for system administrators or application developers who are installing and configuring Oracle GoldenGate Services Architecture. It is assumed that readers are familiar with web technologies and have a general understanding of Windows and UNIX platforms.

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc`.

Accessible Access to Oracle Support

Oracle customers who have purchased support have access to electronic support through My Oracle Support. For information, visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info` or visit `http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs` if you are hearing impaired.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Related Information

The Oracle GoldenGate Product Documentation Libraries are found at

Oracle GoldenGate

Oracle GoldenGate Application Adapters

Oracle GoldenGate for Big Data

Oracle GoldenGate Plug-in for EMCC

Oracle GoldenGate Monitor

Oracle GoldenGate for HP NonStop (Guardian)

Oracle GoldenGate Veridata

Oracle GoldenGate Studio

Additional Oracle GoldenGate information, including best practices, articles, and solutions, is found at:

Oracle GoldenGate A-Team Chronicles

1
Introduction to Oracle GoldenGate

Learn what is Oracle GoldenGate, why and when should you use it, and get familiar with some of the basic terminology and keywords associated with Oracle GoldenGate.

Topics:

- What is Oracle GoldenGate?
  Oracle GoldenGate is a software product that allows you to replicate, filter, and transform data from one database to another database.
- Why Do You Need Oracle GoldenGate?
  Enterprise data is typically distributed across the enterprise in heterogeneous databases. To get data between different data sources, you can use Oracle GoldenGate to load, distribute, and filter transactions within your enterprise in real-time and enable migrations between different databases in near zero-downtime.
- When Do You Use Oracle GoldenGate?
  Oracle GoldenGate meets almost any data movement requirements you might have. Some of the most common use cases are described in this section.
- How Do You Use Oracle GoldenGate?
  After installation, Oracle GoldenGatecan be configured to meet your organization's business needs.
- Oracle GoldenGate Product Family
  There are numerous products in the Oracle GoldenGate product family.

1.1 What is Oracle GoldenGate?

Oracle GoldenGate is a software product that allows you to replicate, filter, and transform data from one database to another database.

Using Oracle GoldenGate, you can move committed transactions across multiple heterogeneous systems in your enterprise. Oracle GoldenGate enables you to replicate data between Oracle Databases, to other supported heterogeneous database, and between heterogeneous databases. In addition, you can replicate to Java Messaging Queues, Flat Files, and to Big Data targets in combination with Oracle GoldenGate for Big Data.

1.2 Why Do You Need Oracle GoldenGate?

Enterprise data is typically distributed across the enterprise in heterogeneous databases. To get data between different data sources, you can use Oracle GoldenGate to load, distribute, and filter transactions within your enterprise in real-time and enable migrations between different databases in near zero-downtime.

To do this, you need a means to effectively move data from one system to another in real-time and with zero-downtime. Oracle GoldenGate is Oracle’s solution to replicate and integrate data.

Oracle GoldenGate has the following key features:

- Data movement is in real-time, reducing latency.
- Only committed transactions are moved, enabling consistency and improving performance.
- Different versions and releases of Oracle Database are supported along with a wide range of heterogeneous databases running on a variety of operating systems. You can replicate data from an Oracle Database to a different heterogeneous database.
- Simple architecture and easy configuration.
- High performance with minimal overhead on the underlying databases and infrastructure.

1.3 When Do You Use Oracle GoldenGate?

Oracle GoldenGate meets almost any data movement requirements you might have. Some of the most common use cases are described in this section.

You can use Oracle GoldenGate to meet the following business requirements:

Business Continuity and High Availability

Business Continuity is the ability of an enterprise to provide its functions and services without any lapse in its operations. High Availability is the highest possible level of fault tolerance. To achieve business continuity, systems are designed with multiple servers, multiple storage, and multiple data centers to provide high enough availability to support the true continuity of the business. To establish and maintain such an environment, data needs to be moved between these multiple servers and data centers, which is easily done using Oracle GoldenGate.

Consider a scenario where you are working in a multinational bank that has its headquarters in London, UK. You work in one of the banks' branches in Bangalore, India. This bank uses a specific account for its financial application that is used globally at all the branches. You have been asked by your manager to daily synchronize the transactions that have happened for this account in the database in the Bangalore branch with the centralized database situated at the UK. The volume of transactions is massive, and even the slightest delay can greatly impact the business. This same process is required at multiple destinations for every database in all the branches of the bank worldwide. This process has to be monitored continuously, preferably through some sort of GUI-based tool for the ease of management. Additionally, the bank has several other, non-critical applications used at all the branches. These applications are based on heterogeneous databases, such as MySQL, but the transactions done over these databases also must be loaded into an Oracle Database located at the headquarters. The replication technology used must support both Oracle and heterogeneous databases so that they can talk to each other. Oracle GoldenGate is an apt solution in such a scenario.

Initial Load and Database Migration

Initial load is a process of extracting data records from a source database and loading those records onto a target database. Initial load is a data migration process that is performed only once. Oracle GoldenGate allows you to perform initial load data migrations without taking your systems offline.

Data Integration

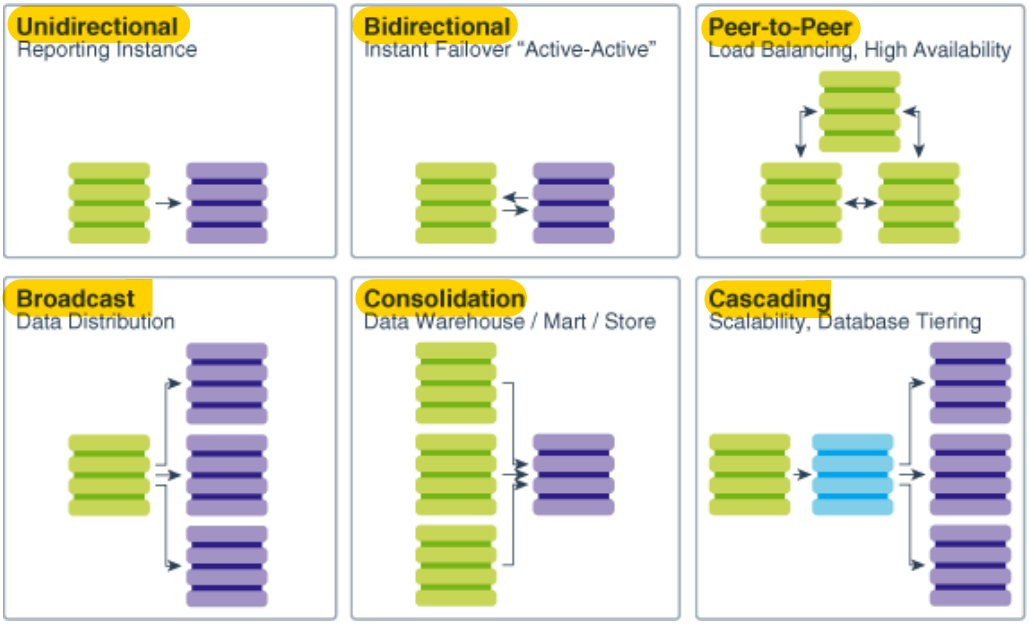
Data integration involves combining data from several disparate sources, which are stored using various technologies, and provide a unified view of the data. Oracle GoldenGate provides real-time data integration.

1.4 How Do You Use Oracle GoldenGate?

After installation, Oracle GoldenGatecan be configured to meet your organization's business needs.

There are many different architectures that can be configured; which range from a simple uni-directional architecture to the more complex peer-to-peer. No matter the architecture, Oracle GoldenGate provides similarities between them, making administration easier.

Figure 1-1 Oracle GoldenGate Supported Topologies



For full information about processing methodology, supported topologies and functionality, and configuration requirements, see the Oracle GoldenGate documentation for your database.

1.5 Oracle GoldenGate Product Family

There are numerous products in the Oracle GoldenGate product family.

- **Oracle GoldenGate Veridata** : Oracle GoldenGate Veridata compares one set of data to another and identifies data that is out-of-sync, and allows you to repair any data that is found out-of-sync.
- **Oracle GoldenGate Plug-in for EMCC**: The Enterprise Manager Plug-in for Oracle GoldenGate extends the Oracle Enterprise Manager Cloud Control and

provides visual support for monitoring and managing Oracle GoldenGateprocesses.

- **Oracle GoldenGate Monitor**: Oracle GoldenGate Monitor is a real-time, Web-based monitoring console that delivers an at-a-glance, graphical view of all of the Oracle GoldenGate instances and their associated databases within your enterprise.
- **Oracle GoldenGate for Big Data**: Oracle GoldenGate for Big Data contains built-in support to write operation data from Oracle GoldenGate trail records into various Big Data targets (such as, HDFS, HBase, Kafka, Flume, JDBC, Cassandra, and MongoDB).
- **Oracle GoldenGate Application Adapters**: Oracle GoldenGate Application Adapters integrate with installations of the Oracle GoldenGatecore product to bring in Java Message Service (JMS) information or to deliver information as JMS messages or files.
- **Oracle GoldenGate for HP NonStop (Guardian)**: Oracle GoldenGate for HP NonStop enables you to manage business data at a transactional level by extracting and replicating selected data records and transactional changes across a variety of heterogeneous applications and platforms.
- **Oracle GoldenGate Studio**: Oracle GoldenGate Studio enables you to design and deploy high-volume, real-time replication by automatically handling table and column mappings, allowing drag and drop custom mappings, generating best practice configurations from templates, and contains context sensitive help.

2

Getting Started with Oracle GoldenGate Microservices Architecture

The Microservices Architecture (MA) for Oracle GoldenGate is a new REST API Microservices-based architecture that allows you to install, configure, monitor, and manage Oracle GoldenGate services using a web-based UI.

You can use MA to deploy, monitor, manage, and perform Extract and Replicat operations on trail data within your MA implementation. To know more about MA see Components of Oracle GoldenGate Microservices Architecture. Classic Architecture illustrates the processes and files required to effectively move data across a variety of topologies. To know more about Classic Architecture, see Components of Classic Architecture and the Oracle GoldenGateconfiguration guide for your database.

- Oracle GoldenGate Architectures Overview
  Oracle GoldenGate has two separate architectures that you can choose to implement according to your requirements.
- Common Data Replication Processes
  There are a number of data replication processes that are common to both Oracle GoldenGate architectures.
- Oracle GoldenGate Key Terms and Concepts
  Apart from the two architectures and their components, there are some key terms that you should get familiar with.
- Components of Oracle GoldenGate Classic Architecture
  You can use the Oracle GoldenGate Classic Architecture to configure and manage your data replications from the command line.
- Components of Oracle GoldenGate Microservices Architecture
  You can use Oracle GoldenGate MA to configure and manage your data replication using an HTML user interface.

2.1 Oracle GoldenGate Architectures Overview

Oracle GoldenGate has two separate architectures that you can choose to implement according to your requirements.

The following table describes the two Oracle GoldenGate architectures and when you should each of the architectures.

	Classic Architecture	Microservices Architecture
What is it?	Oracle GoldenGate classic architecture provides the processes and files required to effectively move data across a variety of topologies. These processes and files form the main components of the classic architecture and was the product design until this release.	Oracle GoldenGate Microservices Architecture is a new administration architecture that provides REST-enabled services as part of the Oracle GoldenGate environment. The REST-enabled services provide remote configuration, administration, and monitoring through HTML5 web pages, command line, and APIs.
When should I use it?	Oracle GoldenGate can be installed and configured to use the Oracle GoldenGate classic architecture for the following purposes: • A static extraction of data records from one database and the loading of those records to another database. • Continuous extraction and replication of transactional Data Manipulation Language (DML) operations and Data Definition Language (DDL) changes (for supported databases) to keep source and target data consistent. • Extraction from a database and replication to a file outside the database. • Capture from heterogeneous database sources.	Oracle GoldenGate can be installed and configured to use the Oracle GoldenGateMicroservices Architecture for the following purposes: • Large scale and cloud deployments with fully-secure HTTPS interfaces and Secure WebSockets for streaming data. • Simpler management of multiple implementations of Oracle GoldenGate environments and control user access for the different aspects of Oracle GoldenGate setup and monitoring. • Support system managed database sharding to deliver fine-grained, multi-master replication where all shards are writable, and each shard can be partially replicated to other shards within a shardgroup. • Support the following features: – Thin and browser-based clients – Network security – User Authorization – Distributed deployments – Remote administration – Performance monitoring and orchestration – Coordination with other systems and services in an Oracle Database environment. – Custom embedding of Oracle GoldenGate into applications or to use secure, remote HTML5 applications.

2.2 Common Data Replication Processes

There are a number of data replication processes that are common to both Oracle GoldenGate architectures.

Topics:

- What is an Extract?
  Extract is a process that is configured to run against the source mining database or configured to run on a downstream database (Oracle-Database only) with capturing data generated in the true source database located somewhere else. This process is the extraction or the data capture mechanism of Oracle GoldenGate.
- What is a Trail?
  A trail is a series of files on disk where Oracle GoldenGate stores the captured changes to support the continuous extraction and replication of database changes.
- What is a Replicat?
  Replicat is a process that delivers data to a target database. It reads the trail file on the target database, reconstructs the DML or DDL operations, and applies them to the target database.

2.2.1 What is an Extract?

Extract is a process that is configured to run against the source mining database or configured to run on a downstream database (Oracle-Database only) with capturing data generated in the true source database located somewhere else. This process is the extraction or the data capture mechanism of Oracle GoldenGate.

You can configure an Extract for the following two use cases:

- **Initial Loads**: When you set up Oracle GoldenGate for initial loads, the Extract process captures the current, static set of data directly from the source objects.
- **Change Synchronization**: When you set up Oracle GoldenGateto keep the source data synchronized with another set of data, the Extract process captures the DML and DDL operations performed on the configured objects after the initial synchronization has taken place. Extracts can run locally on the same server as the database or on another server using the downstream Integrated Extract for reduced overhead. It stores these operations until it receives commit records or rollbacks for the transactions that contain them. If it receives a rollback, it discards the operations for that transaction. If it receives a commit, it persists the transaction to disk in a series of files called a trail, where it is queued for propagation to the target system. All of the operations in each transaction are written to the trail as a sequentially organized transaction unit. This design ensures both speed and data integrity.

> **Note:**
>
> Extract ignores operations on objects that are not in the Extract configuration, even though a transaction may also include operations on objects that are in the Extract configuration.

The Extract process can be configured to extract data from three types of data sources:

- Source tables: This source type is used for initial loads.
- Database recovery logs or transaction logs: While capturing from the logs, the actual method varies depending on the database type. Some examples of this source type are the Oracle Database redo logs or SQL/MX audit trails.
- Third-party capture modules: This method provides a communication layer that passes data and metadata from an external API to the Extract API. The database vendor or a third-party vendor provides the components that extract the data operations and pass them to Extract.

2.2.2 What is a Trail?

A trail is a series of files on disk where Oracle GoldenGate stores the captured changes to support the continuous extraction and replication of database changes.

A trail can exist on the source system, an intermediary system, the target system, or any combination of those systems, depending on how you configure Oracle GoldenGate. On the local system, it is known as an Extract trail (or local trail). On a remote system, it is known as a remote trail. By using a trail for storage, Oracle GoldenGate supports data accuracy and fault tolerance. The use of a trail also allows extraction and replication activities to occur independently of each other. With these processes separated, you have more choices for how data is processed and delivered. For example, instead of extracting and replicating changes continuously, you could extract changes continuously and store them in the trail for replication to the target later, whenever the target application needs them.

In addition, trails allow Oracle Database to operate in heterogeneous environment. The data is stored in a trail file in a consistent format, so it can be read by Replicat process for all supported databases.

Processes that write to the trail file:

The Extract and the data pump processes write to the trail. Only one Extract process can write to a given local trail. All local trails must have different full-path names though you can use the same trail names in different paths.

Multiple data pump processes can each write to a trail of the same name, but the physical trails themselves must reside on different remote systems, such as in a data-distribution topology. For example, a data pump named 1pump and a data pump named 2pump can both reside on sys01 and write to a remote trail named aa. Data pump 1pump can write to trail aa on sys02, while data pump 2pump can write to trail aa on sys03.

Processes that read from the trail file:

The data pump and Replicat processes read from the trail files. The data pump extracts DML and DDL operations from a local trail that is linked to an Extract process, performs further processing if needed, and transfers the data to a trail that is read by the next Oracle GoldenGate process downstream (typically Replicat, but could be another data pump if required).

The Replicat process reads the trail and applies the replicated DML and DDL operations to the target database.

Trail file creation and maintenance:

The trail files are created as needed during processing. You specify a two-character name for the trail when you add it to the Oracle GoldenGate configuration with the `ADD RMTTRAIL` or `ADD EXTTRAIL` command. By default, trails are stored in the `dirdat` sub-directory of the Oracle GoldenGate directory. You can specify a six or nine digit sequence number using the `TRAIL_SEQLEN_9D | TRAIL_SEQLEN_6D GLOBALS` parameter; `TRAIL_SEQLEN_9D` is set by default.

Trail files age automatically to allow processing to continue without interruption for file maintenance. As each new file is created, it inherits the two-character trail name appended with a unique nine digit sequence number from 000000000 through 999999999 (for example `c:\ggs\dirdat\tr000000001`). When the sequence number reaches 999999999, the numbering starts over at 000000000, and previous trail files are overwritten. trail files can be purged on a routine basis by using the Manager parameter `PURGEOLDEXTRACTS`.

You can create more than one trail to separate the data from different objects or applications. You link the objects that are specified in a `TABLE` or `SEQUENCE` parameter to a trail that is specified with an `EXTTRAIL` or `RMTTRAIL` parameter in the Extract parameter file. To maximize throughput, and to minimize I/O load on the system, extracted data is sent into and out of a trail in large blocks. Transactional order is preserved.

> **Note:**
>
> **Extract Files**: You can configure Oracle GoldenGate to store extracted data in an extract file instead of a trail. The extract file can be a single file, or it can be configured to roll over into multiple files in anticipation of limitations on file size that are imposed by the operating system. It is similar to a trail, except that checkpoints are not recorded. The file or files are created automatically during the run. The same versioning features that apply to trails also apply to extract files.

2.2.3 What is a Replicat?

Replicat is a process that delivers data to a target database. It reads the trail file on the target database, reconstructs the DML or DDL operations, and applies them to the target database.

The Replicat process uses dynamic SQL to compile a SQL statement once and then executes it many times with different bind variables. You can configure the Replicat process so that it waits a specific amount of time before applying the replicated operations to the target database. For example, a delay may be desirable to prevent the propagation of errant SQL, to control data arrival across different time zones, or to allow time for other planned events to occur.

For the two common uses cases of Oracle GoldenGate, the function of the Replicat process is as follows:

- **Initial Loads**: When you set up Oracle GoldenGate for initial loads, the Replicat process applies a static data copy to target objects or routes the data to a high-speed bulk-load utility.
- **Change Synchronization**: When you set up Oracle GoldenGate to keep the target database synchronized with the source database, the Replicat process

applies the source operations to the target objects using a native database interface or ODBC, depending on the database type.

You can configure multiple Replicat processes with one or more Extract processes and Data Pumps in parallel to increase throughput. To preserve data integrity, each set of processes handles a different set of objects. To differentiate among Replicat processes, you assign each one a group name

If you don't want to use multiple Replicat processes, you can configure a single Replicat process in coordinated or integrated mode.

- Coordinated mode is supported on all databases that Oracle GoldenGate supports. In coordinated mode, the Replicat process is threaded. One coordinator thread spawns and coordinates one or more threads that execute replicated SQL operations in parallel. A coordinated Replicat process uses one parameter file and is monitored and managed as one unit.
- Integrated mode is supported for Oracle Database releases 11.2.0.4 or later. In integrated mode, the Replicat process leverages the apply processing functionality that is available within the Oracle Database. Within a single Replicat configuration, multiple inbound server child processes known as apply servers apply transactions in parallel while preserving the original transaction atomicity.

2.3 Oracle GoldenGate Key Terms and Concepts

Apart from the two architectures and their components, there are some key terms that you should get familiar with.

What are Checkpoints?

Checkpoints store the current read and write positions of a process to disk for recovery purposes. Checkpoints ensure that data changes that are marked for synchronization actually are captured by Extract and applied to the target by Replicat, and they prevent redundant processing. They provide fault tolerance by preventing the loss of data should the system, the network, or an Oracle GoldenGate process need to be restarted. For complex synchronization configurations, checkpoints enable multiple Extract or Replicat processes to read from the same set of trails.

Checkpoints work with inter-process acknowledgments to prevent messages from being lost in the network. Oracle GoldenGate has a proprietary guaranteed-message delivery technology.

Extract creates checkpoints for its positions in the data source and in the trail. Because Extract only captures committed transactions, it keeps track of the operations in all open transactions, in the event that any of them are committed. This requires Extract to record a checkpoint where it is currently reading in a transaction log, plus the position of the start of the oldest open transaction, which can be in the current or any preceding log.

For Oracle Databases, you can control the amount of transaction log that must be re-processed after an outage in an Oracle Database. Extract also persists the current state and data of processing to disk at specific intervals, including the state and data (if any) of long-running transactions. If Extract stops after one of these intervals, it can recover from a position within the previous interval or at the last checkpoint, instead of having to return to the log position where the oldest open long-running transaction first appeared.

Replicat creates checkpoints for its position in the trail. Replicat stores these checkpoints in a checkpoint file on disk, and optionally, in a checkpoint table in the target database. The checkpoint table is stored with a user-specified name and location. The checkpoint file is stored in the `dirchk` sub-directory of the Oracle GoldenGate directory.

At the completion of each transaction, Replicat writes information about that transaction to a row in the checkpoint table, linking the transaction with a unique position in a specific trail file. Replicat also writes a checkpoint to the checkpoint file when it completes a transaction. At regular intervals, Replicat also writes its current read position to the checkpoint file. These positions are typically not at a transaction boundary, but at some point within a transaction. The interval length is controlled by the `CHECKPOINTSECS` parameter.

Because the checkpoint table is part of the database, and benefits from the database recovery system, it provides a more efficient recovery point for Replicat. The last checkpoint in the checkpoint file may not be the most recent transaction boundary. It could be the middle of a transaction not yet applied by Replicat or an earlier transaction that was already applied. The checkpoint table ensures that Replicat starts at the correct transaction boundary, so that each transaction is applied only once. The information in the checkpoint file can be used for recovery in some cases, but is primarily used for purposes, such as for the `INFO` commands in `GGSCI`.

Regular backups of the Oracle GoldenGate environment, including the trails, should match your database backup, recovery, and retention policies. Restoring the database (and with it the checkpoint table) to an earlier period of time causes Replicat to reposition to an earlier checkpoint that matches that time. If the required trail files for this time period are already aged off the system, they must be restored from backup.

Checkpoints are not required for non-continuous types of configurations, such as a batch load or initial load. If there is a failure, these processes can be started again from the original start point.

What are Process Types?

Depending on the requirement, Oracle GoldenGate can be configured with the following processing types.

- An ***online*** Extract or Replicat process runs until stopped by a user. Online processes maintain recovery checkpoints in the trail so that processing can resume after interruptions. You use online processes to continuously extract and replicate DML and DDL operations (where supported) to keep source and target objects synchronized.
- A ***source-is-table*** (also known as in initial-load Extract) Extract process extracts a current set of static data directly from the source objects in preparation for an initial load to another database. This process type does not use checkpoints.
- A ***special-run*** Replicat process applies data within known begin and end points. You use a special Replicat run for initial data loads, and it also can be used with an online Extract to apply data changes from the trail in batches, such as once a day rather than continuously. This process type does not maintain checkpoints, because the run can be started over with the same begin and end points.
- A ***remote task*** is a special type of initial-load process in which Extract communicates directly with Replicat over TCP/IP. Neither a Collector process nor temporary disk storage in a trail or file is used.

What are Groups?

To differentiate among multiple Extract or Replicat processes on a system, you define processing groups. For example, to replicate different sets of data in parallel, you would create two Replicat groups.

A processing group consists of a process (either Extract or Replicat), its parameter file, its checkpoint file, and any other files associated with the process. For Replicat, a group may also include an associated checkpoint table. You define groups by using the `ADD EXTRACT` and `ADD REPLICAT` commands in one of the Oracle GoldenGate command line interfaces: `GGSCI` or `adminclient`.

All files and checkpoints relating to a group share the name that is assigned to the group itself. Any time that you issue a command to control or view processing, you supply a group name or multiple group names by means of a wildcard.

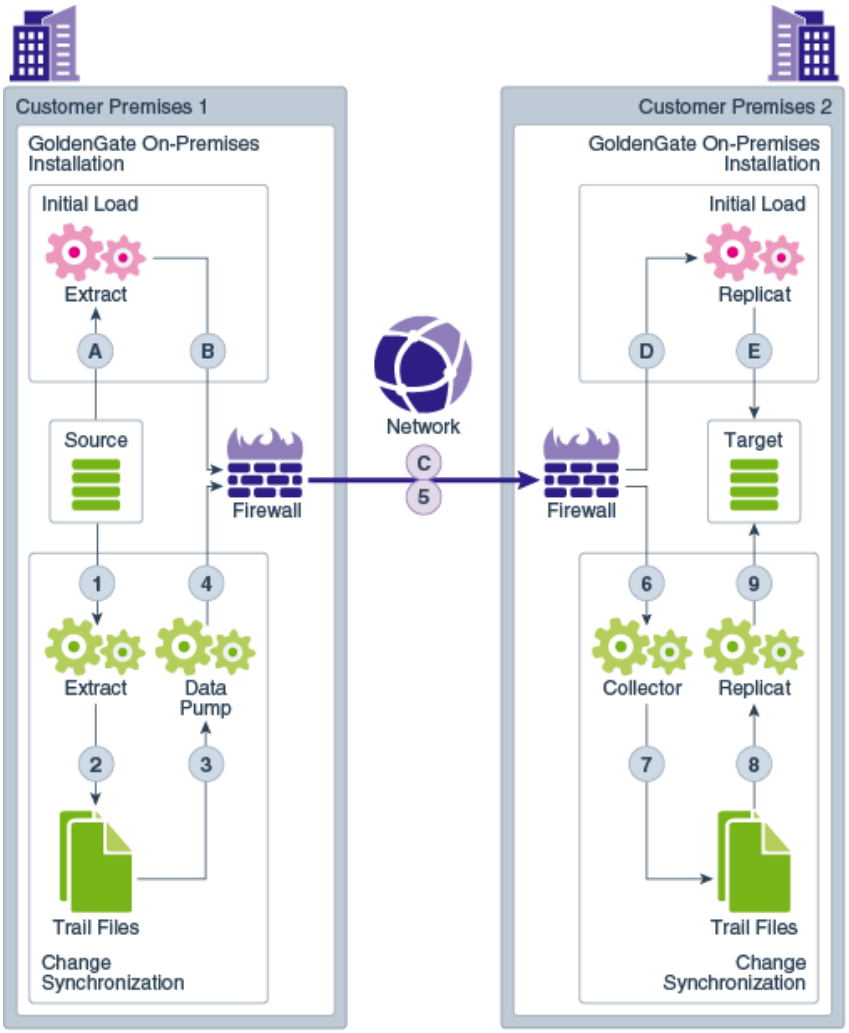
What is a Commit Sequence Number?

When working with Oracle GoldenGate, you might need to refer to a Commit Sequence Number, or CSN. A CSN is an identifier that Oracle GoldenGate constructs to identify a transaction for the purpose of maintaining transactional consistency and data integrity. It uniquely identifies a point in time in which a transaction commits to the database.

The CSN can be required to position Extract in the transaction log, to reposition Replicat in the trail, or for other purposes. It is returned by some conversion functions and is included in reports.

2.4 Components of Oracle GoldenGate Classic Architecture

You can use the Oracle GoldenGate Classic Architecture to configure and manage your data replications from the command line.



> **Note:**
>
> This is the basic configuration. Depending on your business needs and use case, you can configure different variations of this model.

Topics:

- What is a Data Pump?
  Data Pump is a secondary Extract group within the source Oracle GoldenGate configuration.
- What is a Collector?
  Collector is a process that runs in the background on the target system when continuous, online change synchronization is active.
- What is a Manager?
  Manager is the control process of Oracle GoldenGate. Manager must be running on each system in the Oracle GoldenGate configuration before the Extract or Replicat processes can be started.

- What is GGSCI?
  You can use the Oracle GoldenGate Software Command Interface (GGSCI) commands to create data replications. This is the command interface between you and Oracle GoldenGate functional components.

2.4.1 What is a Data Pump?

Data Pump is a secondary Extract group within the source Oracle GoldenGate configuration.

If you configure a Data Pump, the Extract process writes all the captured operations to a trail file on the source database. The Data Pump reads the trail file on the source database and sends the data operations over the network to the remote trail file on the target database. Though configuring a Data Pump is optional, it is highly recommended for most configurations. If a data pump is not used, the Extract process must streams all the captured operations to a trail file on the remote target database.

The Data Pump can be configured in two ways:

- Perform data manipulation: Data Pump can be configured to perform data filtering, mapping, and conversion.
- Perform no data manipulation: Data Pump can be configured in pass-through mode, where data is passively transferred as-is, without manipulation. Pass-through mode increases the throughput of the Data Pump, because all of the functionality that looks up object definitions is bypassed.

Though configuring a data pump is optional, Oracle recommends it for most configurations.. Some reasons for using a data pump include the following:

- **Protection against network and target failures:** In a basic Oracle GoldenGate configuration, with only a trail on the target system, there is nowhere on the source system to store the data operations that Extract continuously extracts into memory. If the network or the target system becomes unavailable, Extract could run out of memory and abend. However, with a trail and data pump on the source system, captured data can be moved to disk, preventing the abend of the primary Extract. When connectivity is restored, the data pump captures the data from the source trail and sends it to the target system(s).
- **You are implementing several phases of data filtering or transformation.** When using complex filtering or data transformation configurations, you can configure a data pump to perform the first transformation either on the source system or on the target system, or even on an intermediary system, and then use another data pump or the Replicat group to perform the second transformation.
- **Consolidating data from many sources to a central target.** When synchronizing multiple source databases with a central target database, you can store extracted data operations on each source system and use data pumps on each of those systems to send the data to a trail on the target system. Dividing the storage load between the source and target systems reduces the need for massive amounts of space on the target system to accommodate data arriving from multiple sources.
- **Synchronizing one source with multiple targets.** When sending data to multiple target systems, you can configure data pumps on the source system for each target. If network connectivity to any of the targets fails, data can still be sent to the other targets.

2.4.2 What is a Collector?

Collector is a process that runs in the background on the target system when continuous, online change synchronization is active.

When the Manager receives a connection request from an Extract process, the Collector scans and binds to an available port and sends the port number to the Manager for assignment to the requesting Extract process. The Collector also receives the captured data that is sent by the Extract process and writes them to the remote trail file.

Collector is started automatically by the Manager when a network connection is required, so Oracle GoldenGate users do not interact with it. Collector can receive information from only one Extract process, so there is one Collector for each Extract that you use. Collector terminates when the associated Extract process terminates.

> **Note:**
>
> Collector can be run manually, if needed. This is known as a static Collector (as opposed to the regular, dynamic Collector). Several Extract processes can share one static Collector; however, a one-to-one ratio is optimal. A static Collector can be used to ensure that the process runs on a specific port.

By default, Extract initiates TCP/IP connections from the source system to Collector on the target, but Oracle GoldenGate can be configured so that Collector initiates connections from the target. Initiating connections from the target might be required if, for example, the target is in a trusted network zone, but the source is in a less trusted zone.

2.4.3 What is a Manager?

Manager is the control process of Oracle GoldenGate. Manager must be running on each system in the Oracle GoldenGate configuration before the Extract or Replicat processes can be started.

Manager must also remain running while the Extract and Replicat processes are running so that resource management functions are performed. One Manager process can control many Extract or Replicat processes.

Manager performs the following functions:

- Starts Oracle GoldenGate processes.
- Starts dynamic processes.
- Maintains port numbers for processes.
- Purges Trail files based on retention rules.
- Creates event, error, and threshold reports.

2.4.4 What is GGSCI?

You can use the Oracle GoldenGate Software Command Interface (GGSCI) commands to create data replications. This is the command interface between you and Oracle GoldenGate functional components.

GGSCI is the Oracle GoldenGate command-line interface. You can use GGSCI to issue the complete range of commands that configure, control, and monitor Oracle GoldenGate.

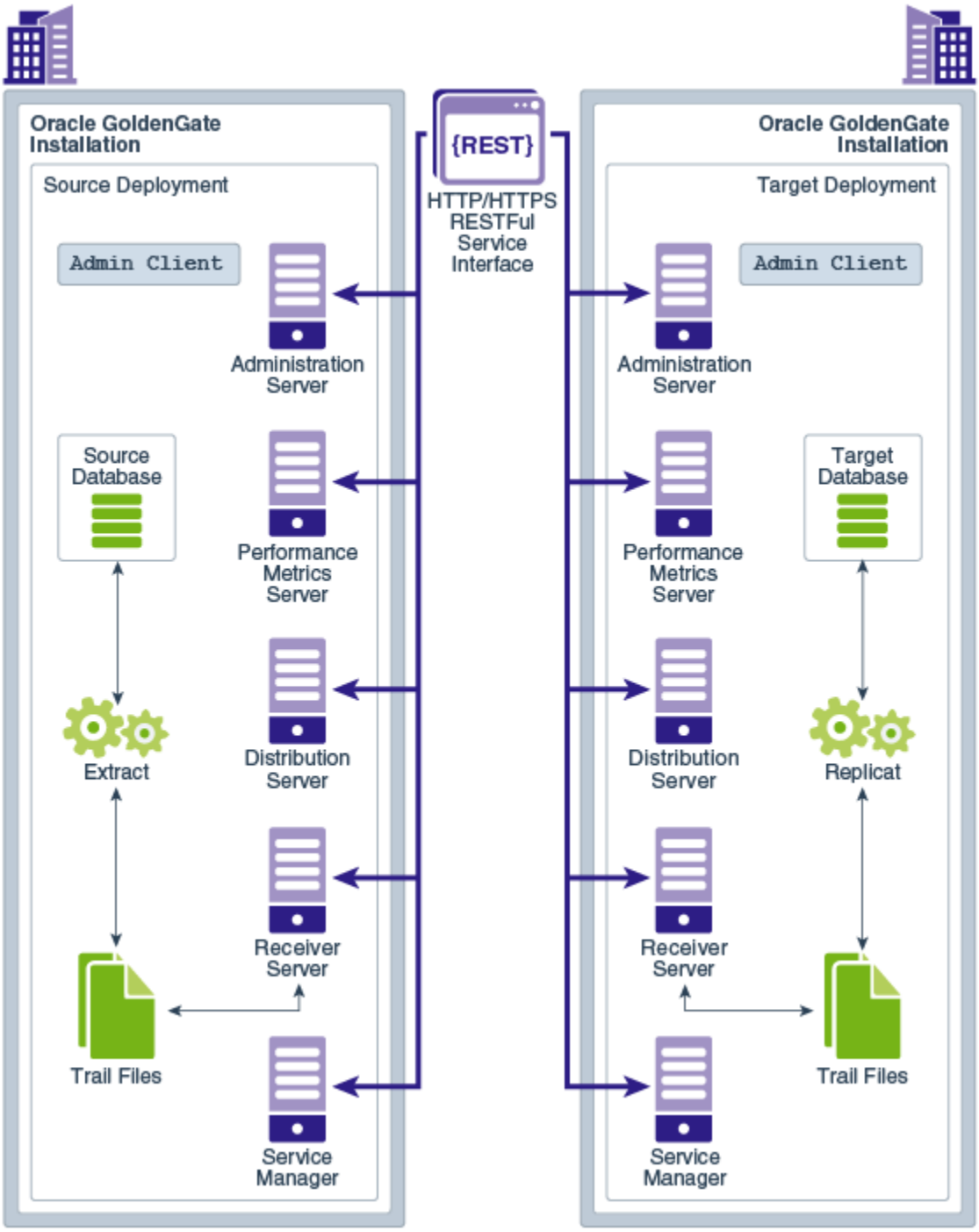
To start GGSCI, change directories to the Oracle GoldenGate installation directory, and then run the `ggsci` executable file.

For more information about Oracle GoldenGate commands, see *Reference for Oracle GoldenGate*.

2.5 Components of Oracle GoldenGate Microservices Architecture

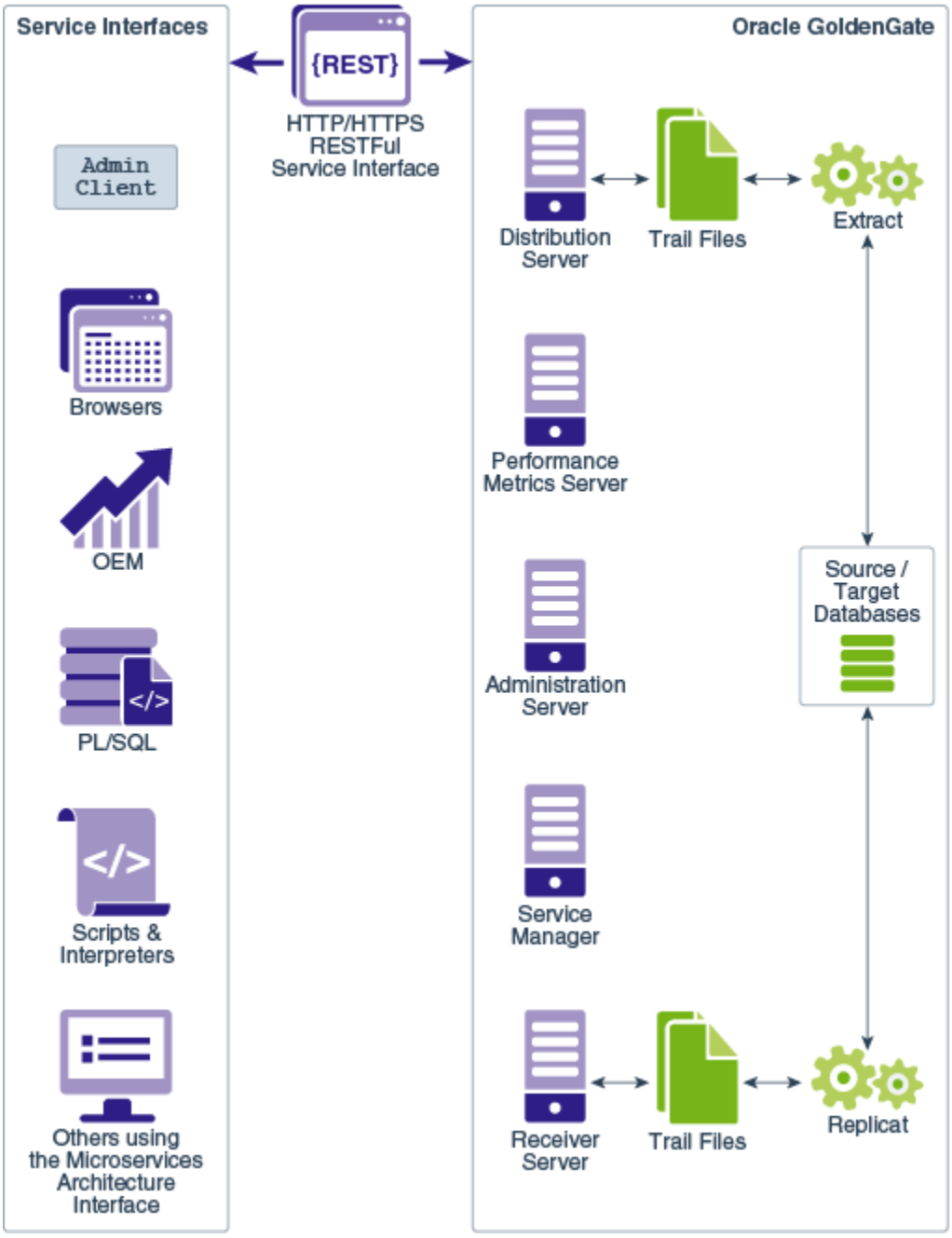
You can use Oracle GoldenGate MA to configure and manage your data replication using an HTML user interface.

There are five main components of the Microservices Architecture. The following diagram depicts the contrast between Oracle GoldenGate MA and the Classic Architecture components, and illustrates how replication processes operate with the secure REST API interfaces.



The Oracle GoldenGateMA provides all the tools you need to configure, monitor, and administer your deployments. It is designed with the industry-standard HTTP communication protocol and the JavaScript Object Notation (JSON) data interchange format. In addition, the architecture provides you with the ability to verify the identity of clients with basic authentication or Secure Sockets Layer client certificates.

The following diagram shows the variety of clients (Oracle products, command line, browsers, and programmatic REST API interfaces) that you can use to administer your deployments using the service interfaces.



Note:

For using any command line utility, you must set up the `OGG_HOME`, `OGG_VAR_HOME`, and `OGG_ETC_HOME` variables correctly in the environment.

Topics:

- What is a Service Manager?
  A Service Manager acts as a watchdog for other services available with Microservices Architecture.

- What is an Administration Server?
  An Administration Server supervises, administers, manages, and monitors processes operating within an Oracle GoldenGate deployment for both active and inactive processes.
- What is a Receiver Server?
  A Receiver Server is the central control service that handles all incoming trail files. It interoperates with the Distribution Server and provides compatibility with the classic architecture pump for remote classic deployments.
- What is a Distribution Server?
  A Distribution Server is an application that functions as a networked data distribution agent in support of conveying and processing data and commands in a distributed networked deployment. It is a high performance application that is able to handle multiple commands and data streams from multiple source trail files, concurrently.
- What is a Performance Metrics Server?
  The Performance Metrics Server uses the metrics service to collect and store instance deployment performance results. This metrics collection and repository is separate from the administration layer information collection.
- What is the Admin Client?
  The Admin Client is a command line utility (similar to the classic GGSCI utility). It uses the REST API published by the Microservices Architecture(MA) Servers to accomplish control and configuration tasks in an Oracle GoldenGate deployment.
- Microservices Architecture (MA) Security, Authentication, and Authorization
  The MA security and authorization model declares and defines how communication security (Confidentiality and Integrity) and Authorization (Authentication and Permissions) are configured and implemented. All the security and authorization configurations and services are common to MA-based servers.These servers authenticate, authorize, and secure access to command and control, monitoring, data conveyance, and information service interfaces for the MA.
- What are the Key Microservices Architecture Directories and Variables?
  The Microservices Architecture is designed with a simplified installation and deployment directory structure.

2.5.1 What is a Service Manager?

A Service Manager acts as a watchdog for other services available with Microservices Architecture.

A Service Manager allows you to manage one or multiple Oracle GoldenGate deployments on a local host.

Service Manager is run as a system service and maintains inventory and configuration information about your deployments and allows you to maintain multiple local deployments. Using the Service Manager, you can start and stop instances, and query deployments and the other services.

2.5.2 What is an Administration Server?

An Administration Server supervises, administers, manages, and monitors processes operating within an Oracle GoldenGate deployment for both active and inactive processes.

The Administration Server operates as the central control entity for managing the replication components in your Oracle GoldenGate deployments. You use it to create and manage your local Extract and Replicat processes without having to have access to the server where Oracle GoldenGate is installed. The key feature of the Administration Server is the REST API Service Interface that can be accessed from any HTTP or HTTPS client, such as the Microservices Architecture service interfaces or other clients like Perl and Python.

In addition, the Admin Client can be used to make REST API calls to communicate directly with the Administration Server, see What is the Admin Client?.

The Administration Server is responsible for coordinating and orchestrating Extracts, Replicats, and paths to support greater automation and operational managements. Its operation and behavior is controlled through published query and service interfaces. These interfaces allow clients to issue commands and control instructions to the Administration Server using REST JSON-RPC invocations that support REST API interfaces.

The Administration Server includes an embedded web application that you can use directly with any web browser and does not require any client software installation.

Use the Administration Server to create and manage:

- Extract and Replicat processes
  - Add, alter, and delete
  - Register and unregister
  - Start and stop
  - Review process information, statistics, reports, and status including LAG and checkpoints
  - Retrieve the report and discard files
- Configuration (parameter) files
- Checkpoint, trace, and heartbeat tables
- Supplemental logging for procedural replication, schema, and tables
- Tasks both custom and standard, such as auto-restart and purge trails
- Credential stores
- Encryption keys (`MASTERKEY`)
- Add users and assign their roles

2.5.3 What is a Receiver Server?

A Receiver Server is the central control service that handles all incoming trail files. It interoperates with the Distribution Server and provides compatibility with the classic architecture pump for remote classic deployments.

A Receiver Server replaces multiple discrete target-side Collectors with a single instance service.

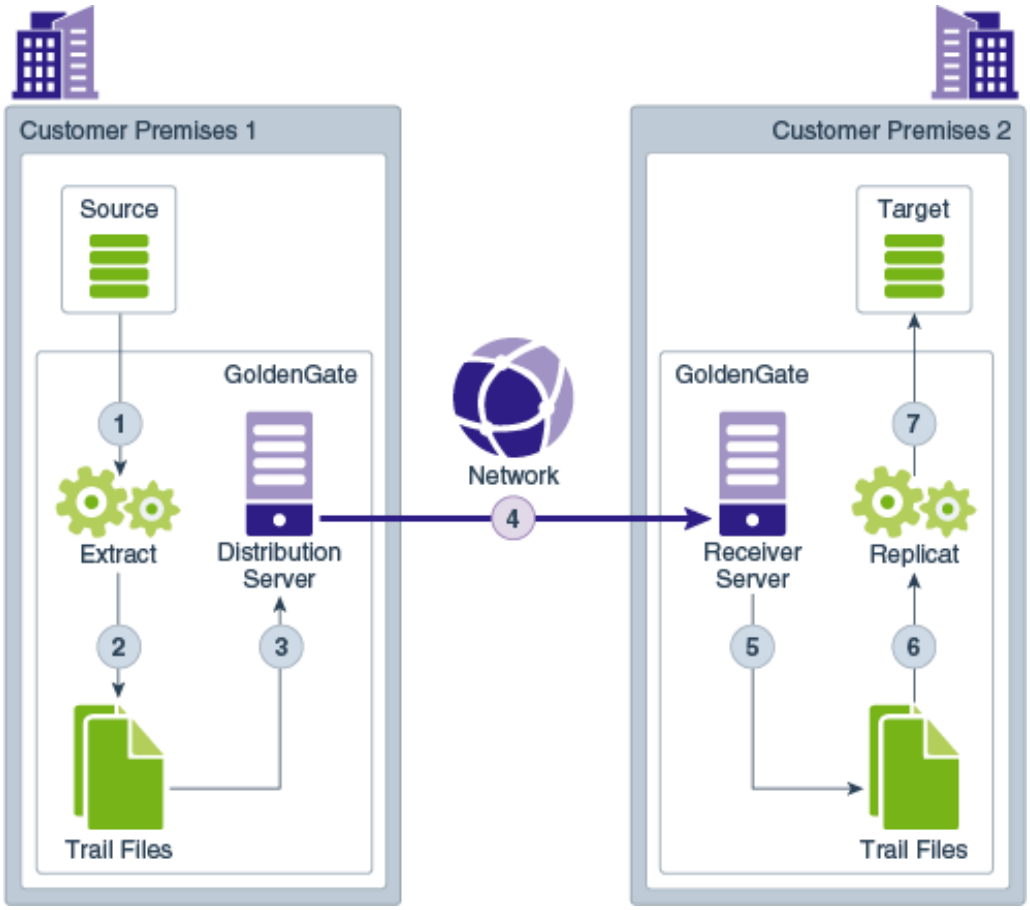
Use Receiver Server to:

- Monitor path events
- Query the status of incoming paths

- View the statistics of incoming paths
- Diagnose path issues

WebSockets is the default HTTPS initiated full-duplex streaming protocol used by the Receiver Server. It enables you to fully secure your data using SSL security. The Receiver Server seamlessly traverses through HTTP forward and reverse proxy servers as illustrated in Figure 2-1

Figure 2-1 Receiver Server Communication



Additionally, the Receiver Server supports the following protocols:

- UDT—UDP-based protocol for wide area networks. For more information, see http://udt.sourceforge.net/.
- Classic Oracle GoldenGate protocol—For classic deployments so that the Distribution Service communicates with the Collector and the Data Pump communicates with the Receiver Service.

> **Note:**
>
> Microservices Architecture does not support TCP encryption. For an MA deployment, a Distribution Server cannot be configured to use TCP encryption, to communicate with the Server Collector running in a deployment. The Receiver Server also cannot accept a connection request from a Pump configured with `RMTHOST ... ENCRYPT` parameter running in a deployment.

2.5.4 What is a Distribution Server?

A Distribution Server is an application that functions as a networked data distribution agent in support of conveying and processing data and commands in a distributed networked deployment. It is a high performance application that is able to handle multiple commands and data streams from multiple source trail files, concurrently.

Distribution Server replaces the classic multiple source-side data pumps with a single instance service. This server distributes one or more trails to one or more destinations and provides lightweight filtering only (no transformations).

Multiple communication protocols can be used, which provide you the ability to tune network parameters on a per path basis. These protocols include:

- Oracle GoldenGate protocol for communication between the Distribution Server and the Collector in a non services-based (classic) target. It is used for inter-operability.

  > **Note:**
  >
  > Microservices Architecture does not support TCP encryption. For an MA deployment, a Distribution Server cannot be configured to use TCP encryption, to communicate with the Server Collector running in a deployment. The Receiver Server also cannot accept a connection request from a Pump configured with `RMTHOST ... ENCRYPT` parameter running in a deployment.

- WebSockets for HTTPS-based streaming, which relies on SSL security.
- UDT for wide area networks.
- Proxy support for cloud environments:
  - SOCKS5 for any network protocol.
  - HTTP for HTTP-type protocols only, including WebSocket.
- Passive Distribution Server to initiate path creation from a remote site. **Paths** are source-to-destination replication configurations though are not included in this release.

> **Note:**
>
> There is no content transformation by the server.

2.5.5 What is a Performance Metrics Server?

The Performance Metrics Server uses the metrics service to collect and store instance deployment performance results. This metrics collection and repository is separate from the administration layer information collection.

You can monitor performance metrics using other embedded web applications and use the data to tune your deployments for maximum performance. All Oracle GoldenGate processes send metrics to the Performance Metrics Server. You can use the Performance Metrics Server in both Microservices Architecture and Classic Architecture.

Use the Performance Metrics Server to:

- Query for various metrics and receive responses in the services JSON format or the classic XML format
- Integrate third party metrics tools
- View error logs
- View active process status
- Monitor system resource utilization

2.5.6 What is the Admin Client?

The Admin Client is a command line utility (similar to the classic GGSCI utility). It uses the REST API published by the Microservices Architecture(MA) Servers to accomplish control and configuration tasks in an Oracle GoldenGate deployment.

The Admin Client is a standalone application used to create processes, rather than using MA. It's not used by MA Servers. For example, you can use the Admin Client to execute all the commands necessary to create a new Extract, create a custom Extract application using the REST API, or use the Administration Server available with MA to configure an Extract.

> **Note:**
>
> Ensure that the `OGG_HOME`, `OGG_VAR_HOME`, and `OGG_ETC_HOME` are set up correctly in the environment.

For more information on environment variables, see Setting Environment Variables.

The way that you use the Admin Client while similar is different in some ways in support of the MA design:

Table 2-1 Admin Client Operation versus GGSCI Operation

GGSCI	Admin Client
Connects to local deployment	Connects to any MA deployment
Requires local machine access, typically SSH	Requires HTTP or HTTPS access
Application logic executed locally	Application logic executed remotely
Requires connection to DBMS	No connection to DBMS required
Uses operating system security	Uses MA security
Authenticated and authorized once	Authenticated and authorized for each operation
No special connect semantics	Requires a `CONNECT` command
Supports `USERID`, `PASSWORD`, and `USERIDALIAS`	Supports `USERIDALIAS` only
`REGISTER EXTRACT` before `ADD EXTRACT`	`REGISTER EXTRACT` after `ADD EXTRACT`
Non-secure communications	Encrypted communications using SSL
Uses pump processes	Use Distribution Server

The Admin Client was designed with GGSCI as the basis the following table describes the new, deleted, and deprecated commands in the Admin Client:

Table 2-2 Admin Client Commands

New Commands	Deleted Commands and Processes:	Deprecated Commands

`CONNECT`
`DISCONNECT`
`[START | STATUS | STOP] SERVICE`
`[ADD | ALTER | DELETE | INFO |`
`[KILL START | STATS | STOP]`
`[EDIT | VIEW] GLOBALS`
`CD`
	`* MGR`
	`* JAGENT`
	`* CREATE DATASTORE`
	`SUBDIRS`
	`FC`
	`DUMPDDL`
	`INFO MARKER`
		`ADD CREDENTIALSTORE`
		`[CREATE | OPEN] WALLET`

2.5.7 Microservices Architecture (MA) Security, Authentication, and Authorization

The MA security and authorization model declares and defines how communication security (Confidentiality and Integrity) and Authorization (Authentication and Permissions) are configured and implemented. All the security and authorization configurations and services are common to MA-based servers.These servers authenticate, authorize, and secure access to command and control, monitoring, data conveyance, and information service interfaces for the MA.

MA defines a model and infrastructure for building service-aware applications. This model is not a generalized model, but one targeted at the current and future Oracle GoldenGate products that need to operate and integrate into global, cloud-based deployment environments. Oracle GoldenGate server programs are implemented

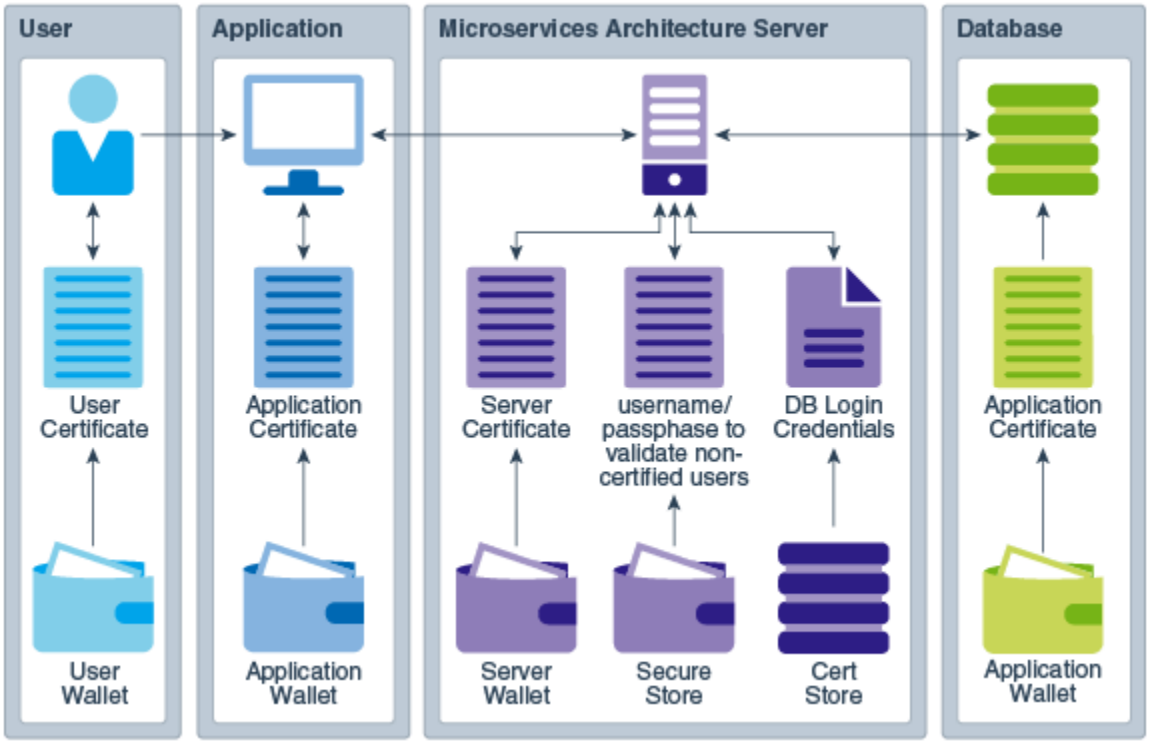
using the MAinfrastructure. All security and configuration implementations are provided by the MA infrastructure as common services.

- Microservices Architecture Authentication
  The goal of the authenticated identity design is to establish identity authentication between users, an MA server or application, and an MA server. The authentication design relies on either the validity of a certificate or of a user credential (username/passphrase pair).
- Microservices Architecture Security
  Communications security in the MA infrastructure, deals with the confidentiality of information sent over communication channels such as networks based on TCP/IP. All MA servers can be configured as secured and unsecured.

2.5.7.1 Microservices Architecture Authentication

The goal of the authenticated identity design is to establish identity authentication between users, an MA server or application, and an MA server. The authentication design relies on either the validity of a certificate or of a user credential (username/passphrase pair).

MA servers publish RESTful service interfaces, which enable user and applications to request services including operational control over one or more MA deployments, service administration, status and performance monitoring. The following illustration depicts the relationship between the user, application, server, and database.



The following types of certificates are used for authentication:

- **Application Certificate**: An Application Certificate is a certificate issued to a specific application. The Application Certificate is stored by the application. Oracle GoldenGate client applications store the Application Certificate in an application

Oracle Wallet designated by the Application configuration. The default location of the application Oracle Wallet is under the installation home directory.

- **User Certificate**: A User Certificate is a certificate issued to a specific user. Oracle GoldenGate client applications store the User Certificate in a user Oracle Wallet. The default location of the user Oracle Wallet is under the user's home directory. Service requests issued with User Certificates include the user name and group information acquired from the host environment. This information identifies the real user executing the application.
- **Server Certificate**: A Server Certificate is a certificate issued to a specific MA server. The Server Certificate is stored by the MA server in the server's Oracle Wallet. The default location of the server Oracle Wallet is under the server's installation directory. An MA server is authenticated to applications as the identity described in the Server Certificate.
- **User's or Application's Database Authentication**: MA servers support Service Interface request whose fulfillment requires logging into a source or target database. MA Server database actions are limited to specific operations required to fulfill service request requirements. The following table describes the type of authentication that are supported by MA servers:

Type of Authentication	Description
MA server database authentication	This configuration sets the MA server to establish connections to the database using its own credentials as the only authenticated user. All service requests requiring database access use the MA server database session. Database operations are logged as originating from the MA
MA server database authentication with database proxy support	This type sets the MA server to establish connections to the database using its own credentials but support proxy user sessions, through an MA server authenticated connection. Proxy support is configured using: User Name or Distinguished Name.
Pass-thru database authentication	This configuration sets the MA server to establish a session or connection to the database using the client provided user name and password.
User-alias database authentication	This configuration sets the MA server to establish a session or connection to the database using a client provided Alias ID that is mapped to a credential, held by the MA server, to establish a session or connection to the database.

Oracle UTL_HTTP Authentication

The User and Application authentication model also applies to database packages that support issuing RESTful Server Interface requests to MA servers. Depending on the security configuration of the MA server, packages or procedures that use the UTL_HTTP database package may need to configure the client database security environment to enable the use of Client-side certificates for authentication in UTL_HTTP.

To enable UTL_HTTP to use client-side certificates:

1. Configure the database client Oracle Wallet.
2. Configure UTL_HTTP with TLS(SSL) for Client-side authentication

Certificate Revocation List (CRL) Authentication Support

MA servers supports CRL checks as part of the authentication process. Although MA servers do not automatically query for updated CRLs, the MA infrastructure supports updating server CRL information at runtime without requiring the MA servers to restart.

User Authentication Delegation

OAuth 2.0 is used to provide delegated authentication and is an established standard for such trusted delegation.

2.5.7.2 Microservices Architecture Security

Communications security in the MA infrastructure, deals with the confidentiality of information sent over communication channels such as networks based on TCP/IP. All MA servers can be configured as secured and unsecured.

The MA service interfaces use the RESTful architectural style, within an HTTP environment. As REST is a style that uses HTTP and not a distinct transfer implementation, all the security related concerns and solutions applied to HTTP apply equally to REST interfaces. This includes ensuring general security related to HTTP-based requests, responses, sessions, cookies, headers and content as well as addressing issues such as Cross Site Request Forgery, UI Redressing and delegated authentication. TLS/SSL when enabled, ensures confidentiality and optionally integrity, although typical configurations do not ensure bi-lateral integrity. Negotiating security configurations can further specify identity validation, renegotiation, and revocation requirements as allowed by Oracle security standards.

Communications Transport

All RESTful Service Interfaces and Data Conveyances may be conducted over the following network transport:

- TCP is used for network communication.
- UDT is an additional protocol used for data conveyance. It is a high-performance, UDP-based data transfer protocol, which transfers large datasets over high-speed WAN.
- WebSockets 2.0 is a not a transport protocol but a pseudo-transport that enables a server to send content to client without client solicitation, thereby enabling bi-directional messaging over a persistent connection. It operates over HTTPS ports simplifying network security management.

Communications Security

An MA server is the originator of all the response messages sent to the client when a request is sent to the server. An MA server neither serves as a proxy nor supports tunneling of response messages generated by other applications. Secured network communications use Oracle approved TLS (Transport Layer Security) or DTLS (Datagram Transport Layer Security) libraries. MA Oracle platforms uses the Oracle SSL toolkit (NZ), which includes Oracle Wallet integration.

For non-Oracle platforms, the Oracle SSL toolkit is used where available. Where the Oracle SSL Toolkit is not available, an alternate SSL toolkit is used.

All MA servers implement client and server authentication. However, client and server authentication is only available when network security is configured and enabled. MA servers can be configured with network security enabled but without using server or client authentication.

Inbound and Outbound Security Configuration

Security configuration can be inbound or outbound. Inbound configuration implies configuring specific behavior associated with a server. A server receives requests and responds with information or messages. Outbound security configuration assumes that the specific behavior is associated with a client.

A client issues requests and receives the response information from the server. An MA server acts as both. For example, in MA, the Distribution Server accepts service requests from clients through inbound configured secured connections, while it connects and sends trail data to Receiver Server through secure connections with Outbound configuration.

MA Security Authentication Modes

The following is the list of supported security authentication modes that establish the authenticity of the entity presenting the authorization information. These are the available values that may be used when setting the `/config/securityDetails/network/common/authMode` security setting. This mode is set when configuring an Oracle GoldenGate MA deployment.

Authorization Mode ID	Notes
`server_only`	Only validate Server certificates. The Server certificates are required. The Client certificates are ignored.
`client_server`	Validate both Client and Server certificates. Both certificates are required.
`clientOptional_server`	This is the default. Validate the client certificate if it is present, as it is optional. Validate the server certificate (it's mandatory).

MA User Privileges

The following is the list of supported security roles.

> **Note:**
>
> These are authorization privileges and are not directly related to authentication.

Role ID	Privilege Level
User	Allows information-only service requests, which do not alter or effect the operation of either the MA. Examples of Query/Read-Only information include performance metric information and

	resource status and monitoring information.
Operator	Allows users to perform only operational actions, such as creating, starting and stopping resources. Operators cannot alter the operational parameters or profiles of the MA server.
Administrator	Grants full access to the user, including the ability to alter general, non-security related operational parameters and profiles of the server.
Security	Grants administration of security related objects and invoke security related service requests. This role has full privileges.

Other Security Features

Some of the other security considerations in MA include the following:

- **Connection Filtering**: This is responsible for qualifying and filtering a candidate connection based on connection policy specifications.
- **Certificate Filtering**: Similar to connection filtering, this feature enables qualifying certificates as part of accepting or denying a connection request.
- **Fall-back Constraints**: Network security configuration within MA servers enables users to configure and constrain the protocol version negotiation fall-back behavior allowing them to control if and how the protocol versions are negotiated.
- **IPv6 Support**: Oracle GoldenGatenetwork implementations support native IPv6 addressing standards.
- **Session Management**: MA Service Interfaces requests are RESTful and stateless, which implies that no client application context it stored on the server between requests. The application session state is entirely held by the client.
- **User Credential Storage**: MA implementations address this by using Oracle Wallets and related identity management services to store security information. Approved encryption technologies are configured to secure both stored and in-flight user data. Stored data typically refers to file system files like capture data trail files while in-flight data typically refers to data transmitted between peers over a non-persistent communications channel.
- **SPAs and WebApp Security**: If the initial connection to the Service Manager uses the HTTPS protocol, then the browser connects using SSL/TLS. If the server is configured to require the client to present a certificate, the browser needs to be configured to present the appropriate client certificate.

> **Note:**
>
> All security related operations including the administration of server, application and user certificates, Certificate Revocation Lists, and security related operational parameters can only be performed through direct login. No security related operational or configuration parameters are presented or exposed through the Service Interfaces.

Cipher-suites

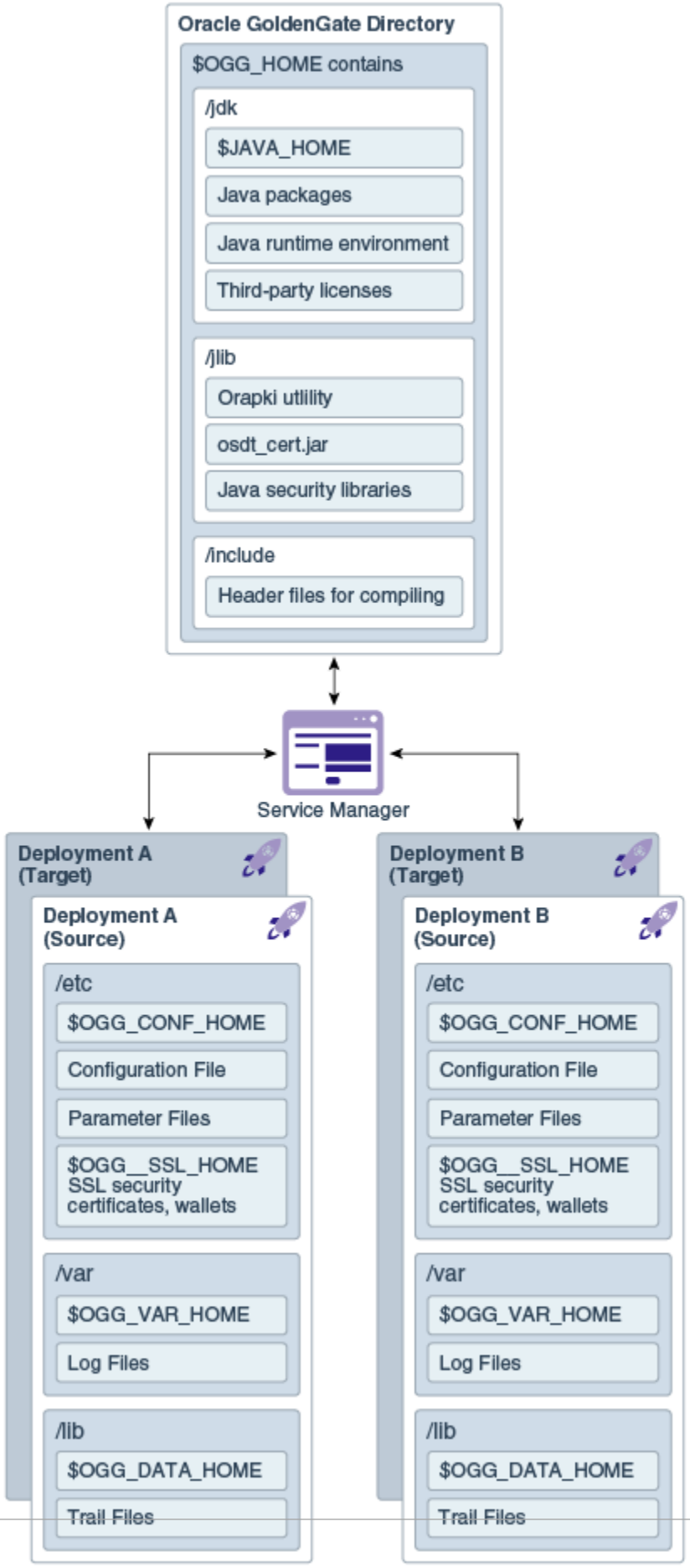
The cipher-suites for MA are configured during deployment. The value of the cipher-suite can be updated by the user through the Server Manager RESTful interfaces for each server or updated using either the MA boostrap configuration override option or the command-line configuration override options. The list of cipher-suites available to a user, differs based on the environment. This ensures that there is sufficient overlap to allow secure communication at the required security level.

Both client and server platforms generally support more than one cipher-suite. This increases the probability that the client and server can negotiate and agree on a cipher-suite to use. The set of available cipher-suites on the server is dictated by the NZ Toolkit (or alternate TLS/SSL toolkit). There are several cipher-suites set as the default set. The default set attempts to specify the most common cipher-suites with the highest security protection and highest performance. However, in practice you need to choose between high security and high performance as these are competing attributes and there is a trade-off between security and performance.

2.5.8 What are the Key Microservices Architecture Directories and Variables?

The Microservices Architecture is designed with a simplified installation and deployment directory structure.

This directory structure is based on the Linux Foundation Filesystem Hierarchy Standard. Additional flexibility has been added to allow parts of the deployment subdirectories to be placed at other locations in the file system or on other devices, including shared network devices. The design is comprised of a read only home directory where you install Oracle GoldenGate and create a custom deployment specific directories as in the following:


Oracle GoldenGate Directory
$OGG_HOME contains
/jdk
$JAVA_HOME
Java packages
Java runtime environment
Third-party licenses
/lib
Orapki utlility
osdt_cert.jar
Java security libraries
/include
Header files for compiling
Service Manager
Deployment A (Target)
Deployment A (Source)
/etc
$OGG_CONF_HOME
Configuration File
Parameter Files
$OGG__SSL_HOME SSL security certificates, wallets
/var
$OGG_VAR_HOME
Log Files
/lib
$OGG_DATA_HOME
Trail Files
Deployment B (Target)
Deployment B (Source)
/etc
$OGG_CONF_HOME
Configuration File
Parameter Files
$OGG__SSL_HOME SSL security certificates, wallets
/var
$OGG_VAR_HOME
Log Files
/lib
$OGG_DATA_HOME
Trail Files

The following table describes the key MA directories and the variables that are used when referring to those directories in an Oracle GoldenGate installation. When you see these variables in an example or procedure, replace the variable with the full path to the corresponding directory path in your enterprise topology.

Table 2-3 Directories in an MA Installation and Deployment

Directory Name	Variable	Description	Default Directory Path
Oracle Database home	*ORACLE_HOME*	The Oracle GoldenGate home that is created on a host computer is the directory that you choose to install the product. This read-only directory contains binary, executable, and library files for the product.	*/database_install_location*
Oracle GoldenGate home	*OGG_HOME*	The Oracle GoldenGate home that is created on a host computer is the directory that you choose to install the product. This read-only directory contains binary, executable, and library files for the product.	*/ogg_install_location*
Deployment configuration home	*OGG_CONF_HOME*	The location in which each deployment information and configuration artifacts are stored.	*/ogg_deployment_location*/etc/conf
Deployment security home	*OGG_SSL_HOME*	The location in which each deployment security artifacts (certificates, wallets) are stored.	*/ogg_deployment_location*/etc/ssl
Deployment data home	*OGG_DATA_HOME*	The location in which each deployment data artifacts (trail files) are stored.	*/ogg_deployment_location*/var/lib/data
Deployment variable home	*OGG_VAR_HOME*	The location in which each deployment logging and reporting processing artifacts are stored.	*/ogg_deployment_location*/var
Deployment etc home	*OGG_ETC_HOME*	The location in which your deployment configuration files are stored including parameter files.	*/ogg_deployment_location*/etc

You can change the default location of all of these to customize where you want to store these files.

In a configuration where the `OGG_VAR_HOME` is a local directory and the `OGG_HOME` is a shared read-only remote directory, many deployments with local `OGG_VAR_HOME` can share one read-only shared `OGG_HOME`.

This directory design facilitates a simple manual upgrade. You simply switch a deployment to use a new Oracle GoldenGate release by changing the `OGG_HOME` directory path in your Service Manager to a new Oracle GoldenGate home directory, which completes the upgrade. You then must restart the MA servers, Extract processes, and Replicat processes.

3

Roadmap for Implementing the Microservices Architecture

Microservices Architecture is based on the REST API. Its services are deployed and accessible through a web interface, once the installation completes successfully. You must make sure that the database is set up correctly, so that MA users can connect and use it effectively.

This topic describes the roadmap for implementing Microservices Architecture components and clients.

Table 3-1 Roadmap for Implementing Oracle GoldenGate Oracle GoldenGateMicroservices Architecture

Task	More Information
Installing the MA	*Installing the Oracle GoldenGate Microservices Architecture*
Starting the Service Manager	How to Connect to a Service Manager
Starting the Servers	Quick Tour of the Service Manager Home Page
(Optional) Starting the Admin Client	How to Use the Admin Client