A scenic view of the Golden Gate Bridge in San Francisco, California, spanning across the water with the city skyline in the background under a blue sky with light clouds.

# New and Exciting!

## Available Now! 已可用!

- GoldenGate V21.3 发布
- GoldenGate 微服务架构是地球上最现代高性能的数据集成软件

Microservices architecture is the most modern, high performance data integration software on the planet

- V21.3 Microservice enabled builds
  - Linux 86\_64 Oracle (unified build)
  - Linux 86\_64 DB2 zOS
  - Linux 86\_64 MySQL
  - Linux 86\_64 PostgreSQL
  - Linux 86\_64 SQLServer



ORACLE



# Goldengate 微服务架构介绍

## GoldenGate Microservices Architecture Introduction

**Werner He**

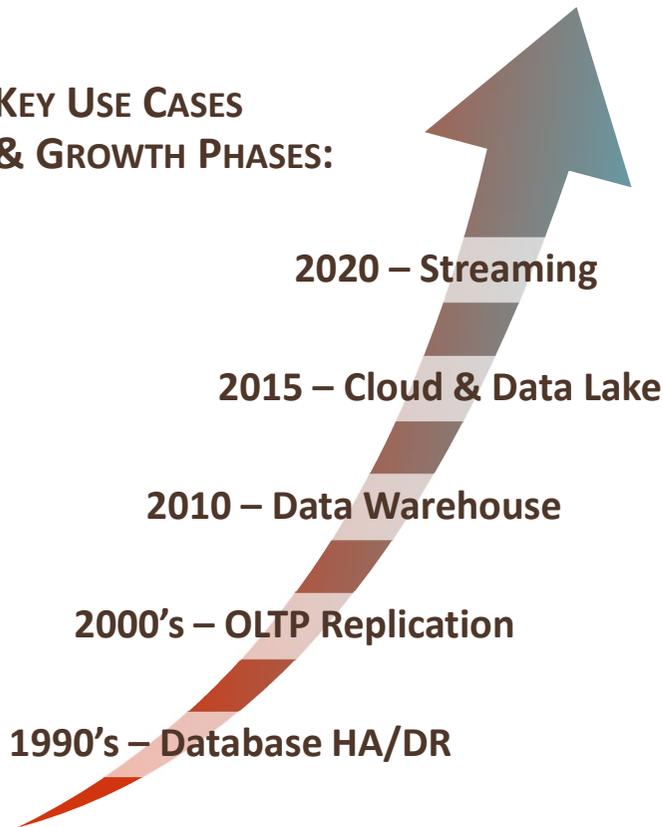
Senior Director, Oracle GoldenGate Product Management

# Today's Agenda

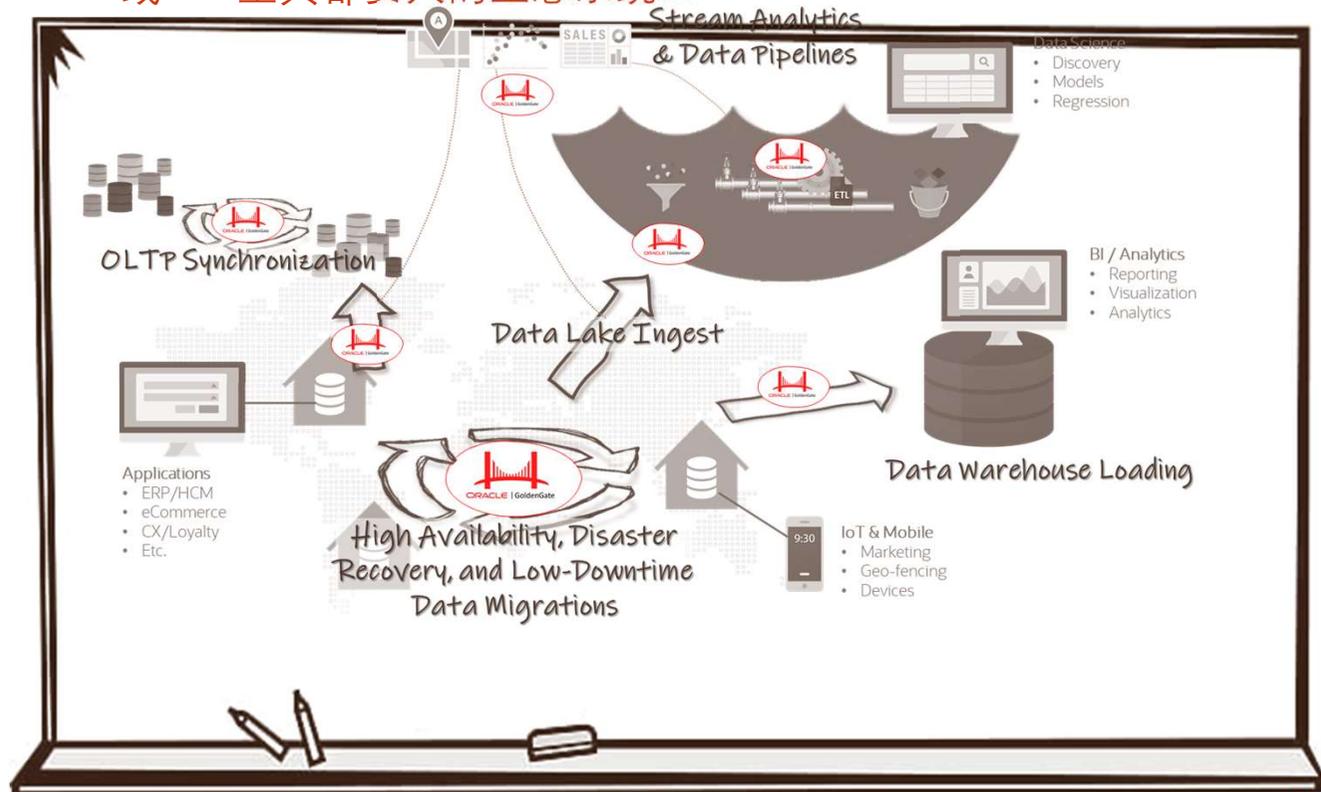
- **GoldenGate Platform**
- **Microservices Architecture - Why**
- **WebUI – For New Generation**
- **Distribution Path – Bridges between Deployments**
- **Adminclient and RestAPI – For Old Timer**
- **Docker Implementation – Easy for SYS Admins**

# GoldenGate Real-Time Data Ecosystem/ Goldengate实时数据生态系统

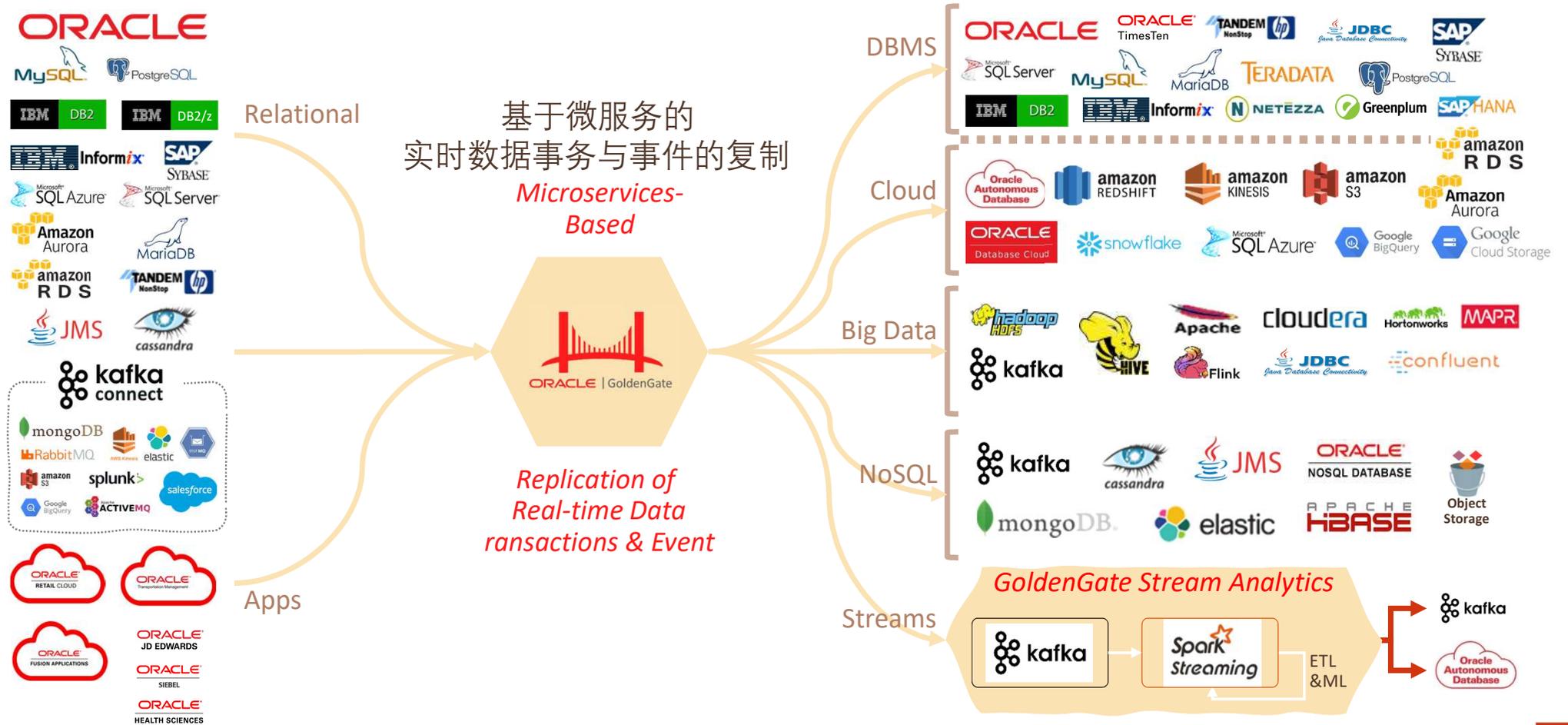
## KEY USE CASES & GROWTH PHASES:



Most successful Integration Tool in history: GoldenGate is a larger ecosystem than any EAI, SOA or ETL tool from any vendor.../历史上最成功的集成工具: GoldenGate是一个比任何供应商的任何EAI、SOA或ETL工具都要大的生态系统...



# Whole Enterprise Solution/全企业解决方案



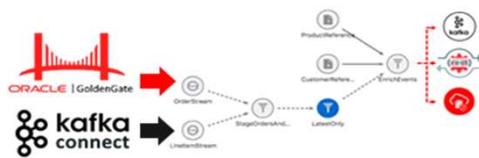
<https://www.oracle.com/middleware/technologies/goldengate.html>



# GoldenGate Platform Solution Capabilities/ GoldenGate平台解决方案功能

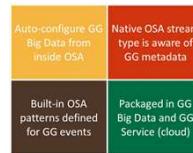
延申模式：流处理  
Expansionary Patterns:  
Stream Processing

Data Pipelines

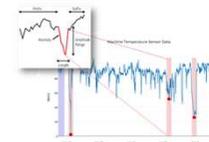


Data Transformation

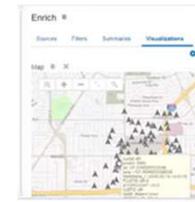
GoldenGate Integrations



Time Series Analysis



Geo-Fencing



Predictive Analytics



高增长模式：非关系数据的复制输入/输出  
High Growth Patterns:  
Replication in/out for Non-Relational

Data Lake Ingest



Streaming Ingest



Cloud Ingest



Messaging Replication



NoSQL Replication



SaaS Replication



基础模式：数据库复制  
Foundation Patterns:  
Database Replication

Unidirectional



Bi-Directional



Peer-to-Peer



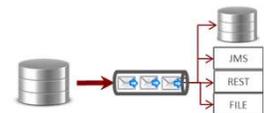
Broadcast



Consolidation



Distribution



# Differentiation: Scale, Trust and Performance/ 差异化: 可扩展、可靠和性能

## >8k Customers in >180 Countries

(Oracle Data Integration / Data Fabric Products and Services)

Communications	Finance & Banking	Media
Energy & Industrial	Health & Insurance	Retail
Services	High Tech	Other

## Global Payments Run on Oracle

(Oracle GoldenGate is backbone for banking High Availability fabric)

ACQUIRERS/ PROCESSORS	
CARD NETWORKS	
ISSUERS	
GATEWAYS	
ISOs / MSPs	

全球约90%的支付行业每天都要依靠 GoldenGate 转移超过30万亿美元的支付、外汇和证券。

~90% of global payments industry depends on GoldenGate to move more than \$30 Trillion in payments, forex and securities every day.

## Performance & Scale is Unmatched

(Core ETL/E-LT, Replication and Streaming is industrial strength)

### Performance Data Points:

#### Oracle DB:

- 4TB REDO/hr, raw performance seen in Oracle lab setting
- 1.5TB REDO/hr, single table, actual perf at large banking customer
- 1m Tx/second, actual perf at Fortune 200 manufacturing cust

#### Non-Oracle DB:

- 4b Tx/day on DB2 active-active at Fortune 200 retailer
- 55k Tx/second on Mainframe to data lake at APAC mega-bank
- 45k Tx/second from DB2/i to Amazon AWS MSK & Dynamo
- 1m DML/hr from MS SQL Server 2012 to AWS RDS (SQL Server)
- Petabytes per day GG to Kafka by most major web-scale companies



# Today's Agenda

- GoldenGate Platform
- Microservices Architecture - Why
- WebUI – For New Generation
- Distribution Path – Bridges between Deployments
- Adminclient and RestAPI – For Old Timer
- Docker Implementation – Easy for SYS Admins

## What's Included In Oracle GoldenGate Microservices?/Oracle GoldenGate微服务中包括哪些内容?

No extra charge, included as part of the base product./作为基本产品的一部分，不收取额外费用。

### Core Changes/核心变化

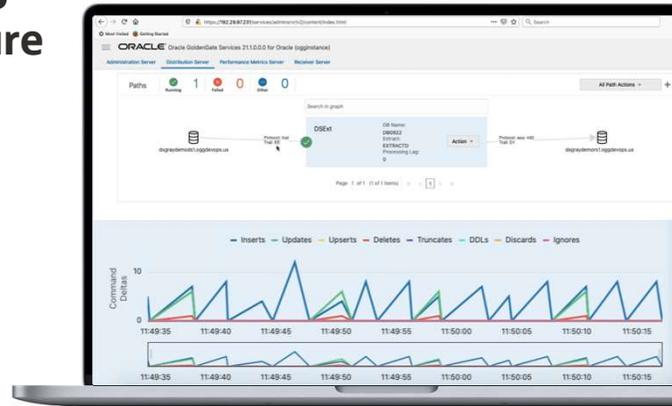
- Improved Command Line Interface – AdminClient/改进了CLI
  - Akin to SQL\*Plus. You can use the AdminClient to connect to any OGG Microservices Deployment, on the local server or any remote server./类似于SQL\*Plus。您可以使用AdminClient连接到本地服务器或任何远程服务器上的任何OGG Microservices部署。
  - More Secure, it now requires a username / password to make changes to OGG. /更安全的是，它现在需要用户名/密码来更改OGG。
  - Locally Authenticated – create a local credential store to securely store passwords for each of your favorite deployments/本地身份验证-创建本地凭据存储，以安全地存储每个常用部署的密码
- Beautiful Web User Interface/美化了WebUI
  - Intuitive graphical user interface that can be used on most browsers to manage your GoldenGate environments. /直观的图形用户界面，可在大多数浏览器上用于管理您的GoldenGate环境。
  - Anything you can do from the CLI can be done from the Web UI. /可以从CLI执行的任何操作都可以从Web UI执行。
  - Reduces the learning curve, and eliminates syntax issues. /减少学习曲线，并消除语法问题。
- REST API
  - A complete Rest API has been built to allow for streamlined development and automation of GoldenGate replication. /已经构建了一个完整的Rest API，以便优化GoldenGate复制的开发和自动化。
  - The Web User Interface is built using this API./ Web用户界面是使用此API构建的。



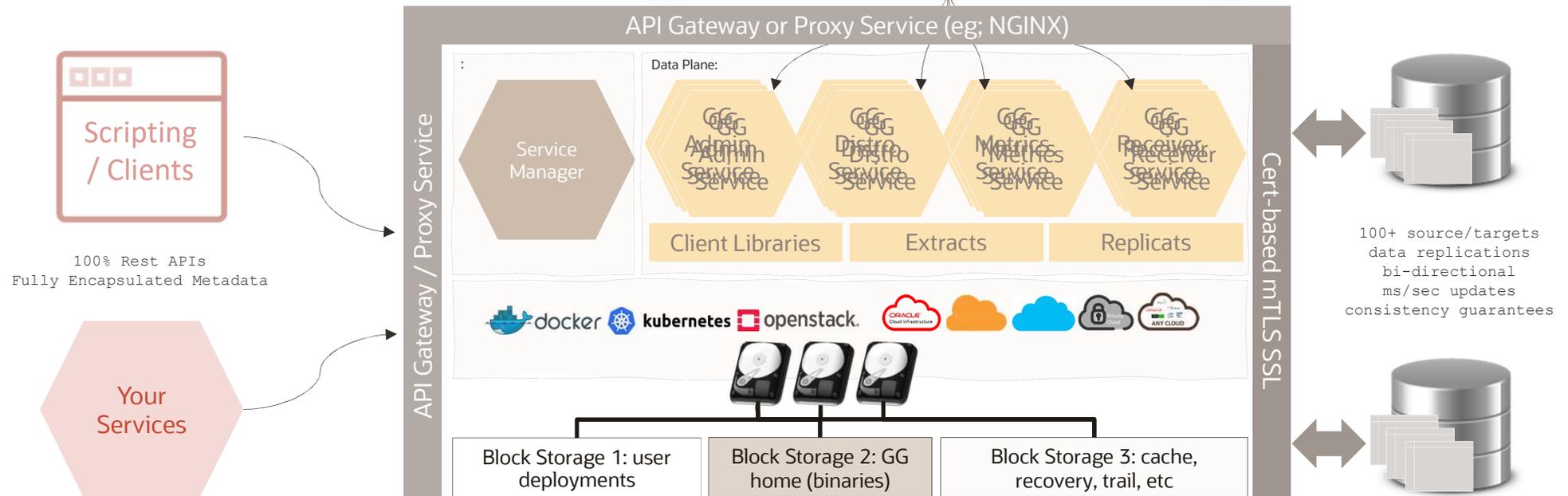
# Oracle GoldenGate微服务体系结构

## Oracle GoldenGate Microservices Architecture

GG微服务是完全封装的、基于C的服务，不需要额外的应用服务器或数据库  
 GG microservices are fully encapsulated, C-based services, no additional app server or databases required



Embedded JavaScript GUI per microservice

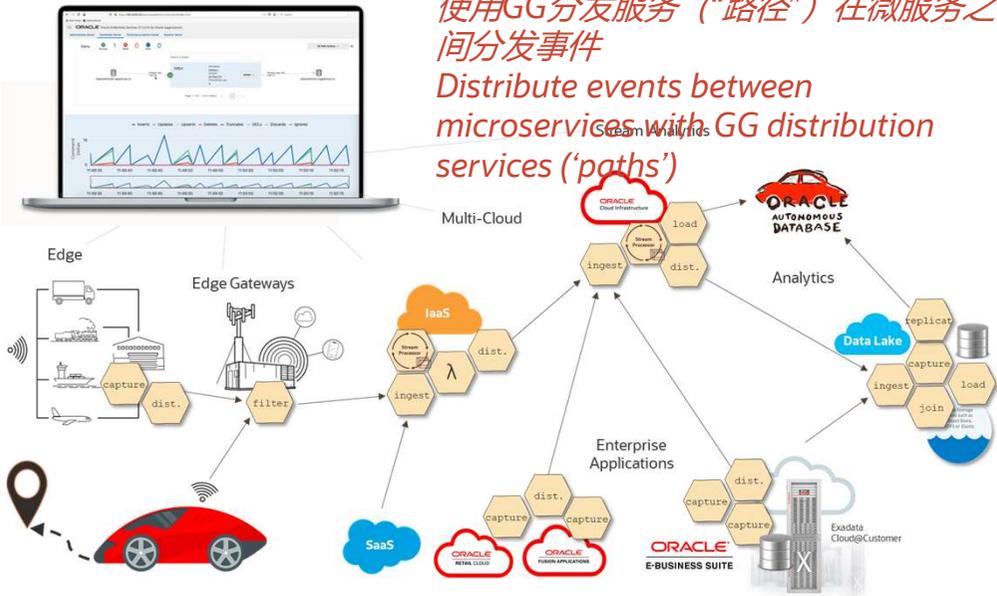


# 将GoldenGate作为网格或中心运行 (客户选择)

## Run GoldenGate as a Mesh or as a Hub (customer choice)

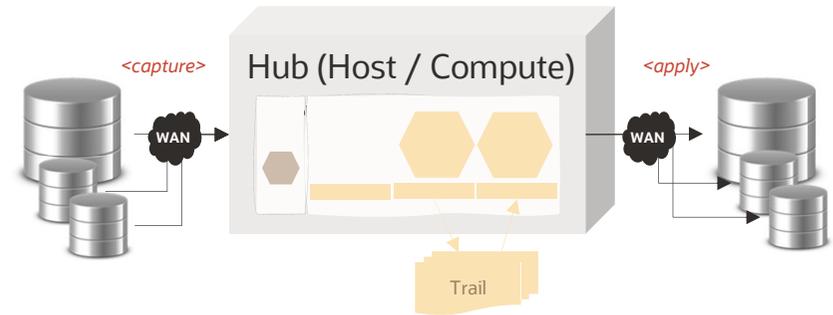
**Mesh:**适用于大型企业和多云  
suitable for large enterprises and multi-cloud

使用GG分发服务 (“路径”) 在微服务之间分发事件  
Distribute events between microservices with GG distribution services (“paths”)



**Hub:**适用于单个项目、单个组织所有权  
suitable for single projects, single org ownership

将所有GG运行时整合到单个主机, 可用于大多数源/目标组合  
Consolidate all GG runtimes to single host, can work for most combinations of source/targets



# Oracle GoldenGate微服务的好处

## Benefits of Oracle GoldenGate Microservices

- More Secure, especially for cloud, and distributed architectures/更安全，特别是对于云和分布式体系结构
  - GoldenGate itself now has users and user authentication/GoldenGate本身现在拥有用户和用户身份验证
  - Improved network security - use WSS or HTTPS to send trail files/改进的网络安全性-使用WSS或HTTPS发送跟踪文件
  - Integration with Oracle Key Vault to store encryption keys on a separate server/与Oracle Key Vault集成以在单独的服务器上存储加密密钥
- Improved Architecture for Performance, Upgrades, Patching and CI/CD Lifecycle/改进了性能、升级、打补丁和CI/CD生命周期的体系结构
  - Single threaded Extract Pump has been replaced with a multithreaded Distribution Service/单线程抽取泵已被多线程分配服务所取代
  - GoldenGate Deployments offer their own private container for replication/GoldenGate部署为复制提供了自己的专用容器
  - Multiple Deployments can be attached to a single OGG Installation/可以将多个部署连接到单个OGG安装
  - Move a Deployment from one installation home to another to upgrade or patch them/将部署从一个安装位置移动到另一个安装位置以升级或修补它们
- Designed for use with Mesh, Hub, or Remote configurations/设计用于网状、集线器或远程配置
  - MAA design pattern for High Availability and Disaster Recovery of the OGG environment itself/OGG环境本身的高可用性和灾难恢复的MAA设计模式
    - <https://www.oracle.com/a/tech/docs/maa-goldengate-hub.pdf>
  - Deployments can be easily moved from one server to another/部署可以轻松地从一台服务器移动到另一台服务器
  - Run deployments in Docker, Kubernetes or directly on the DB servers/在Docker、Kubernetes或直接在DB服务器上运行部署



## Benefits of Oracle GoldenGate Microservices/Oracle GoldenGate微服务的好处

Continued from previous slide/继续上一张幻灯片

- Same mature core functionality that has been around for years/与多年前相同的成熟核心功能
  - Same Extract and Replicat processes as Classic Architecture/与经典体系结构相同的提取和复制过程
  - Same Trail files as Classic Architecture/与经典体系结构相同的跟踪文件
- Support for multiple platforms/支持多种平台
  - All Oracle database stores, on-premise or cloud/所有Oracle数据库存储，本地或云
  - Non-Oracle databases: SQL Server, DB2 (LUW & iSeries), PostgreSQL, MySQL
  - NoSQL: Oracle NoSQL, Cassandra, Mongo, etc.
  - Big Data: 18+ different technologies (Hadoop, Kafka, Elastic, etc)
  - Cloud: Oracle Cloud, Amazon, Azure, Google, etc.
- Mixed environment support/混合环境支持
  - Replicate from classic architecture Microservices and Vice-Versa/从经典体系结构微服务复制，反之亦然
- Most importantly, it is the strategic direction of the product/最重要的是，这是产品的战略方向



# Oracle GoldenGate Microservices REST API



- Standard GET / PUT calls
  - All fully documented: <https://docs.oracle.com/en/middleware/goldengate/core/21.1/oggra/>
- Make calls from anywhere/从任何地方调用
  - Web applications
  - PL/SQL (via UTL\_HTTPS package)
  - Scripts and Interpreters (Shell script, Python, Java and etc.)
- Integration with Oracle Clusterware / Grid Control for High Availability/与Oracle Clusterware/Grid Control集成以实现高可用性
- Integration with Oracle ACFS and DBFS for Disaster Recovery/与Oracle ACFS和DBFS集成以实现灾难恢复
- Support for Oracle, and non-Oracle builds, including Big Data/支持Oracle和非Oracle构建，包括大数据



# Customer Example/客户案例: Wells Fargo



## Streaming Replication Operational and Analytic Data Services/流式复制操作和分析数据服务

### Run the Business/业务运行

- High Availability (HA) of the online operational systems is of utmost importance, these run core banking applications/在线操作系统的高可用性 (HA) 至关重要, 这些系统运行核心银行应用程序
- GoldenGate is used for real-time replication and data synchronization between core operational apps/GoldenGate用于核心运营应用程序之间的实时复制和数据同步
  - >1,000 operational databases using GoldenGate
  - Migrated to newest GG Microservices architecture for efficiency

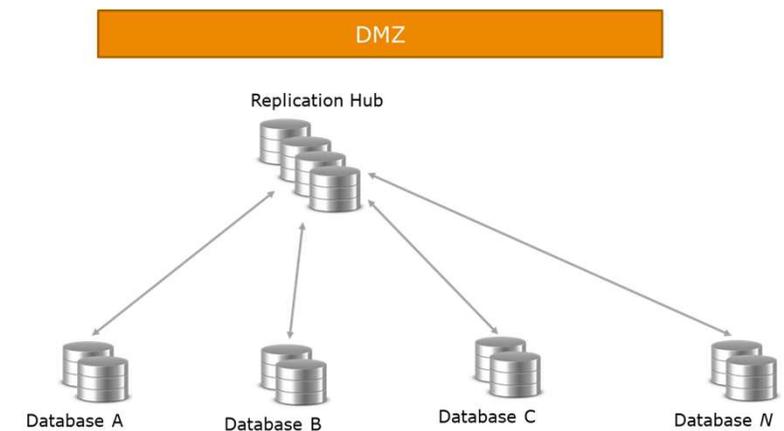
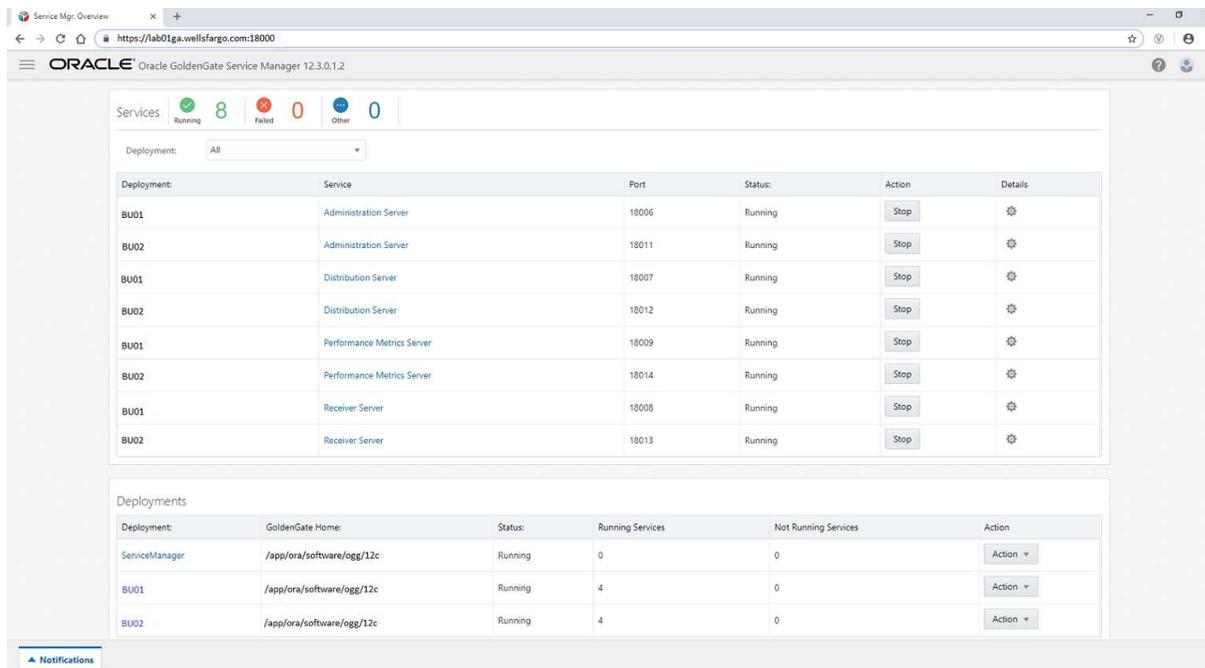
### Analyze the Business/业务分析

- GoldenGate provides the database events in real-time to a wide variety of analytic systems, including Hadoop, Kafka, Cassandra and Object Storage/GoldenGate为各种分析系统提供实时数据库事件, 包括Hadoop、Kafka、Cassandra和对象存储
  - >250 GoldenGate Hubs used for replication



# Data Fabric/Mesh for Operational and Analytic Data Flows/用于操作和分析数据流的数据结构/网格

Wells Fargo is a leader in adoption of GoldenGate Microservices architecture and has presented on the best practices, and approaches for simplifying >1500 GG Homes down to >250 using microservices hubs/富国银行 (Wells Fargo) 是采用GoldenGate微服务体系结构的领导者，并介绍了使用微服务中心将超过1500个GG简化为超过250个GG的最佳实践和方法



#OOW2018

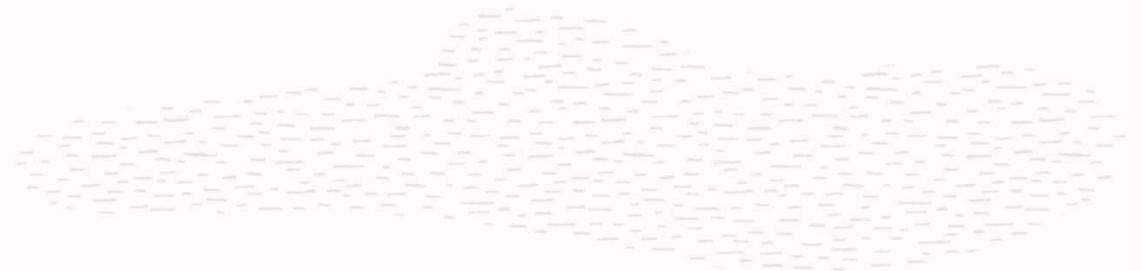


# Today's Agenda

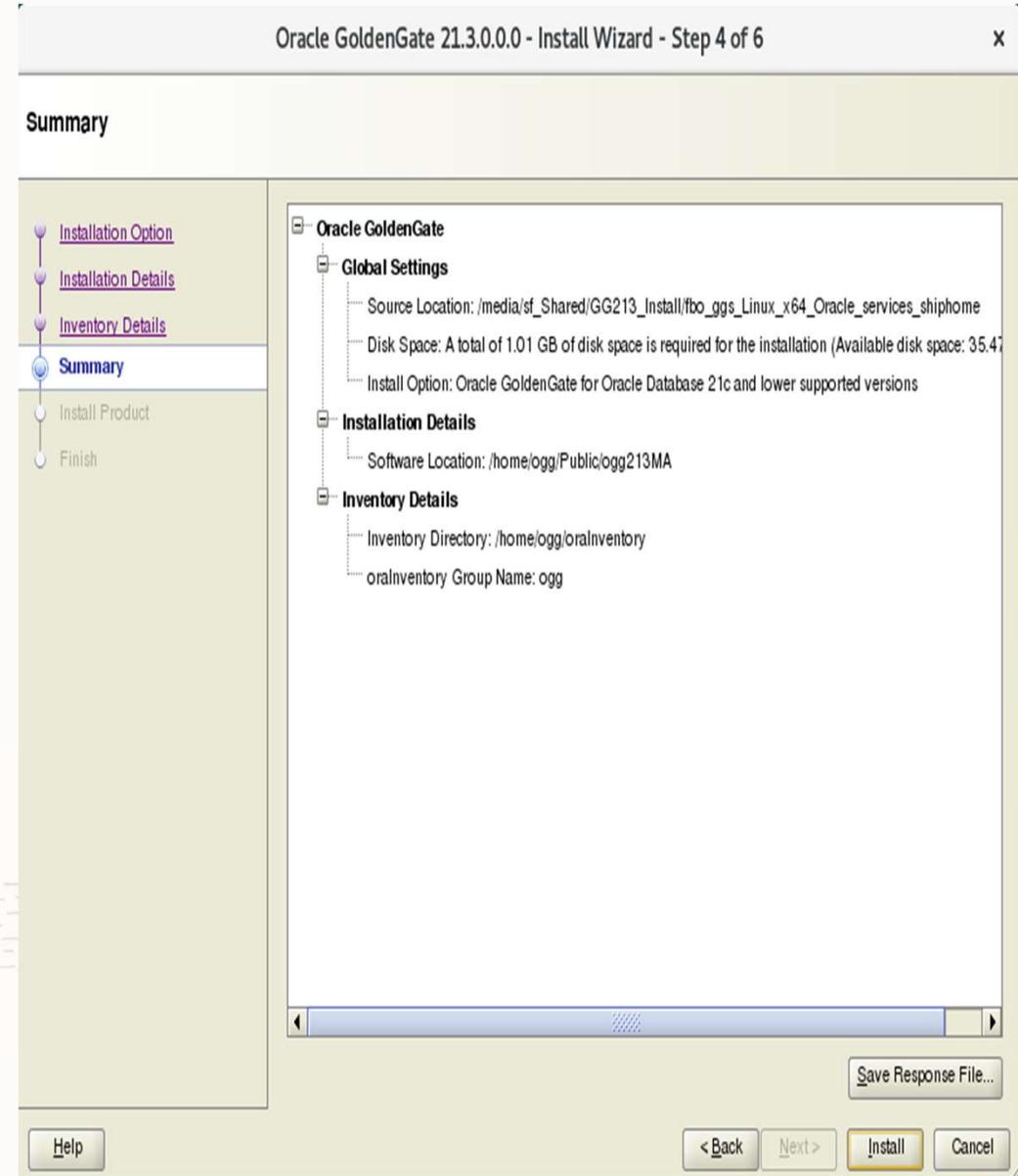
- GoldenGate Platform
- Microservices Architecture - Why
- WebUI – For New Generation
- Distribution Path – Bridges between Deployments
- Adminclient and RestAPI – For Old Timer
- Docker Implementation – Easy for SYS Admins

## GUI Installation and Configuration/GUI的安装与配置

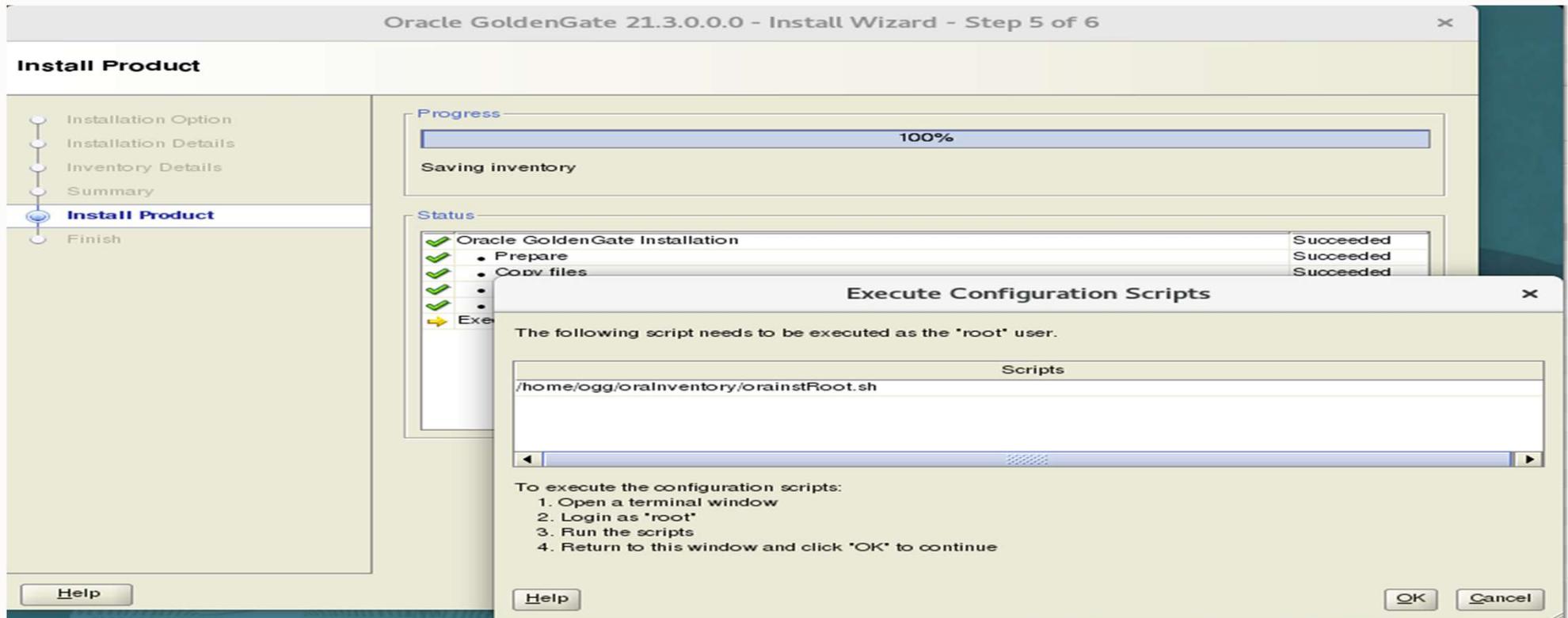
- Oracle Universal Installer (OUI) to install the binary programs/Oracle Universal Installer (OUI) 安装二进制程序
- oggca.sh to configure GoldenGate/oggca.sh配置GoldenGate
  - Create Service Manager/创建服务管理器
  - Create Deployment(s)/创建部署
  - Integrate with RAC/与RAC集成
  - Adding GoldenGate Service Manager as system service/将GoldenGate服务管理器添加为系统服务



# OUI Installation/OUI安装

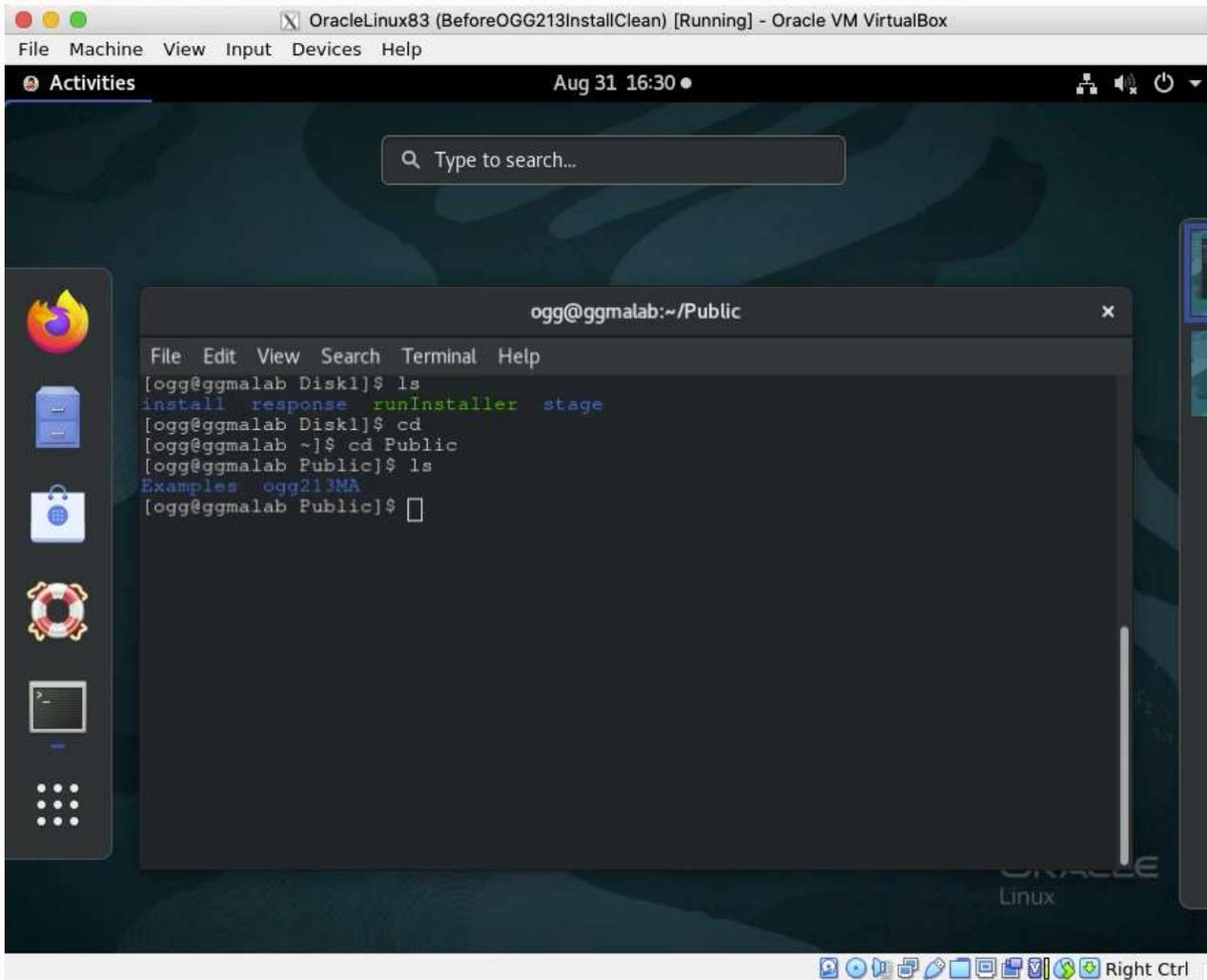


## OUI Installation – continued/OUI安装 (续)



- Support Response file
- Support Oracle install inventory

# oggca Create Service Manager and Deployment/oggca创建服务管理器和部署

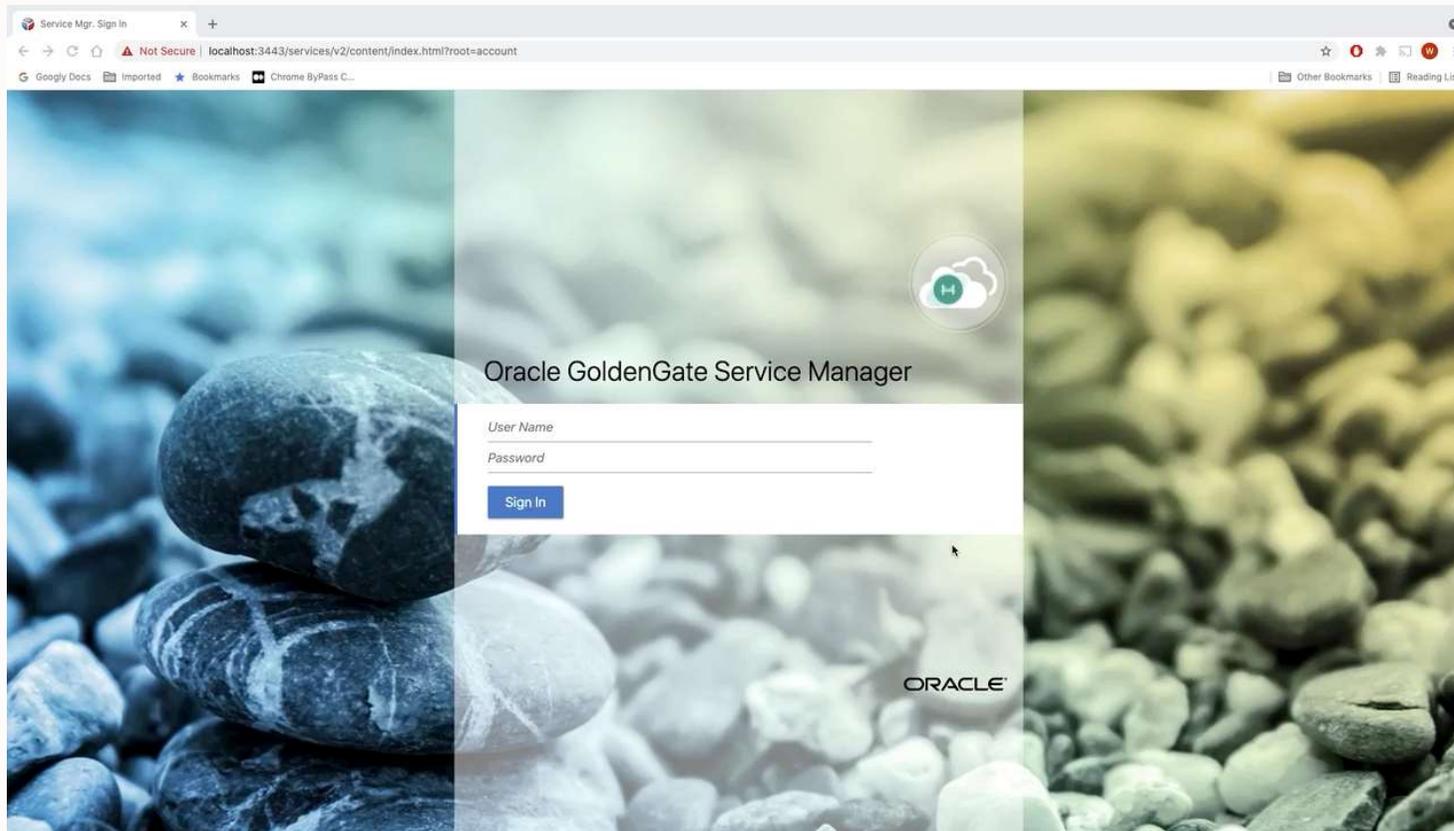


```
File Machine View Input Devices Help
Activities Aug 31 16:30
Type to search...
ogg@ggmalab:~/Public
File Edit View Search Terminal Help
[ogg@ggmalab Disk1]$ ls
install response runInstaller stage
[ogg@ggmalab Disk1]$ cd
[ogg@ggmalab ~]$ cd Public
[ogg@ggmalab Public]$ ls
Examples ogg213MA
[ogg@ggmalab Public]$
```

- Create Service Manager
- Create Deployment
- Choose ssl certification
- Determine the ports
- Response File is supported



# Easy to use WebUI Interface/易于使用的WebUI界面



# Unified Home Page for Deployment/用于部署的统一主页

## Manage adminsvr, distsvr, recvsrvr, pmsrvr in sam

Paths

Running 1 Failed 0 Other 0

All Path Actions +

Search in graph

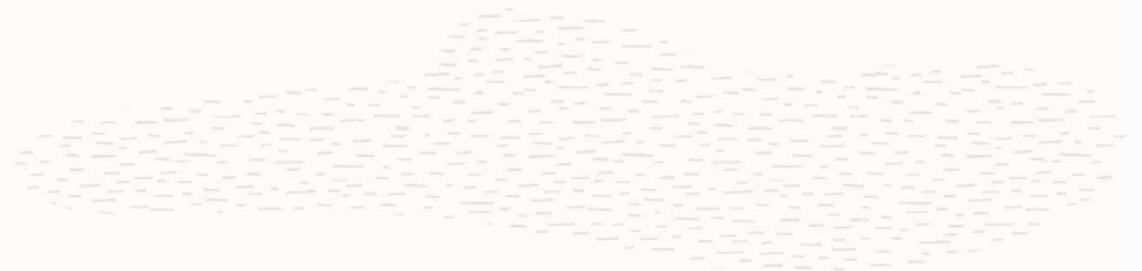
den03cdi Protocol: trail Trail: aa test1 DB Name: Extract: Action Protocol: wss: 9003 Trail: bb den03cdi

Page 1 of 1 (1 of 1 items) < 1 >



## Simplified Certificate Management/简化证书管理

- Web UI and REST APIs to
  - View certificates
  - Replace server certificates
  - Add multiple client certificates
  - Add CA certs (trust points)
- Important - Not supported - direct manipulation of wallets after installation /重要信息-不支持-安装后直接操作钱包
- Distribution paths can choose which client certificate to present to remote site if using certificate authentication method/如果使用证书身份验证方法，分发路径可以选择要向远程站点提供的客户端证书



# Simplified Certificate Management / 简化

ORACLE Oracle GoldenGate Service Manager 21.3.0.0.0 (ServiceManager)

Certificate Management

Deployment Name: oggma213dk

Client Certificates +

Status	Name	Issued To	Issued By	Validity Start	Validity Expiration	Action
✓	ClientCert	CN = wggclient	CN = wggclient	2021-08-18T04:01:41Z	2031-08-16T04:01:41Z	  

CA Certificates +

Shared

Status	Name	Issued To	Issued By	Validity Start	Validity Expiration
✓	NGINXCert	CN = wggma213dk	CN = wggma213dk	2021-08-18T03:37:49Z	2031-08-16T03:37:49Z
✓	DigiRoot	CN = DigiCert Global Root CA	CN = DigiCert Global Root CA	2006-11-10T00:00:00Z	2031-11-10T00:00:00Z
✓	NGINX_GGBD_Marketplace	CN = ogg21cbigdata.sub10292314461.wernervcn.oraclevcn.com	CN = ogg21cbigdata.sub10292314461.wernervcn.oraclevcn.com	2021-08-03T19:23:22Z	2031-08-01T19:23:22Z
✓	wggma21windk	CN = wggma21windk	CN = wggma21windk	2021-07-01T04:33:09Z	2031-06-29T04:33:09Z
✓	windk19	CN = windkkg19ma	CN = windkkg19ma	2021-02-14T00:19:40Z	2031-02-12T00:19:40Z

Local



# References

- GG MA File Locations
  1. GLOBALS and ENCKEYS : \$OGGDEPLOY/etc/conf/ogg/
  2. Parameter files : \$OGGDEPLOY/etc/conf/ogg/
  3. Activity Logging : \$OGGDEPLOY/etc/conf/logging
  4. Report files : \$OGGDEPLOY/var/lib/report/ (IE/IR)
  5. SCA log files :\$OGGDEPLOY/var/log/
  6. Trails : \$OGGDEPLOY/var/lib/data/
  7. SQL scripts (ddl\_setup etc.): \$OGGHOME/lib/sql/legacy
  8. COM : \$OGGDEPLOY/var/temp
  9. BR/checkpoint files : \$OGGDEPLOY/var/run
  - 10.Wallet for Trail file encryption : \$OGGDEPLOY/var/lib/wallet
  11. Datastore : \$OGGDEPLOY/var/run/dirbdb or \$OGGDEPLOY/var/run/dirlmdb
  12. DB login Credential : \$OGGDEPLOY/var/lib/credential
  13. GG Service login wallet: \$OGGDEPLOY/var/lib/credential/secureStore



# References

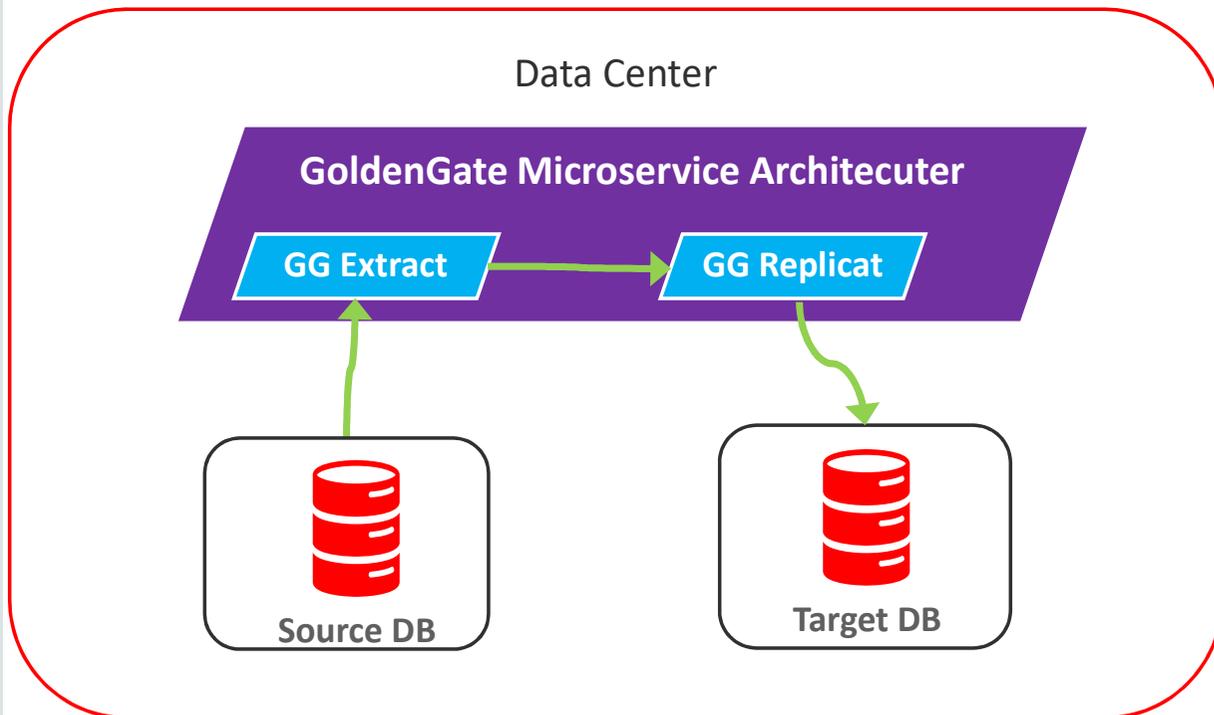
- GG MA Install  
[Installing Oracle GoldenGate Microservices Architecture](#)  
[Upgrading Oracle GoldenGate Microservices Architecture - GUI](#)  
[Using OGGCA to Create Your Secure Deployment](#)
- Using GGMA  
[How to Add Extracts](#)  
[How to Add Replicats](#)  
[How to Add a Distribution Path](#)  
[How to Add Target-Initiated Distribution Path](#)  
[How to Configure an Encryption Profile](#)



# Today's Agenda

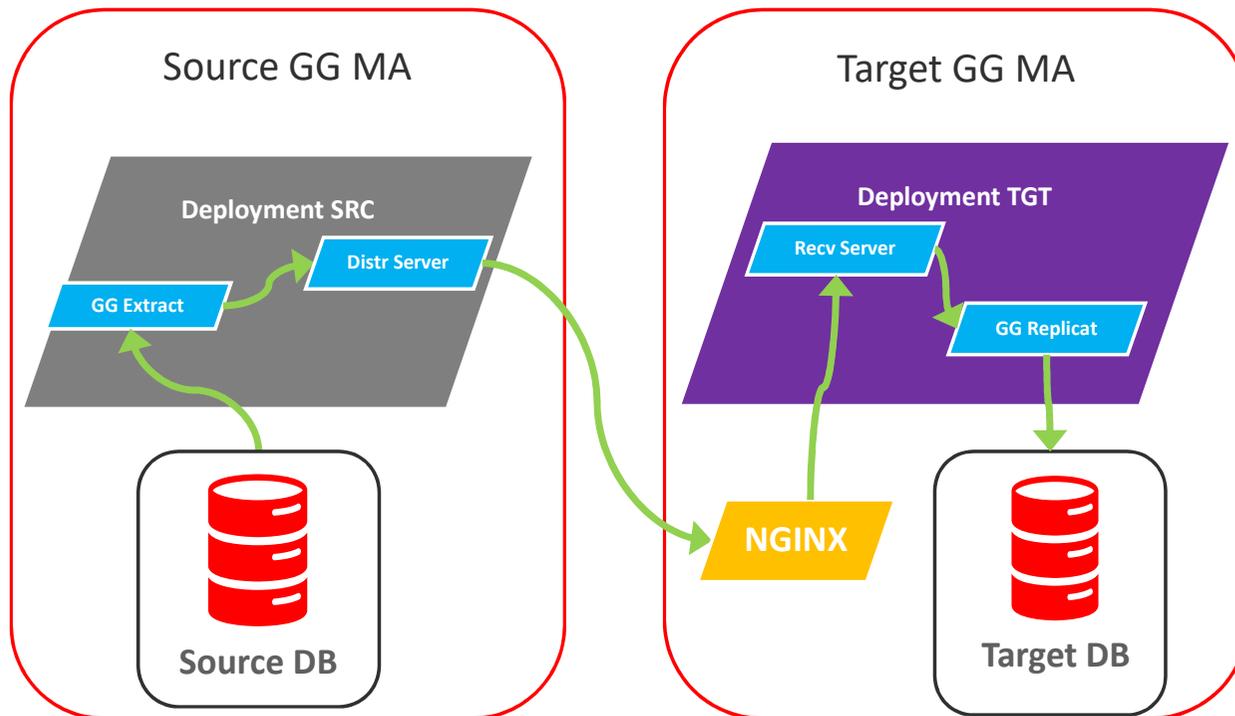
- GoldenGate Platform
- Microservices Architecture - Why
- WebUI – For New Generation
- **Distribution Path – Bridges between Deployments**
- Adminclient and RestAPI – For Old Timer
- Docker Implementation – Easy for SYS Admins

# Basic GG Hub Configuration/基本GG Hub配置



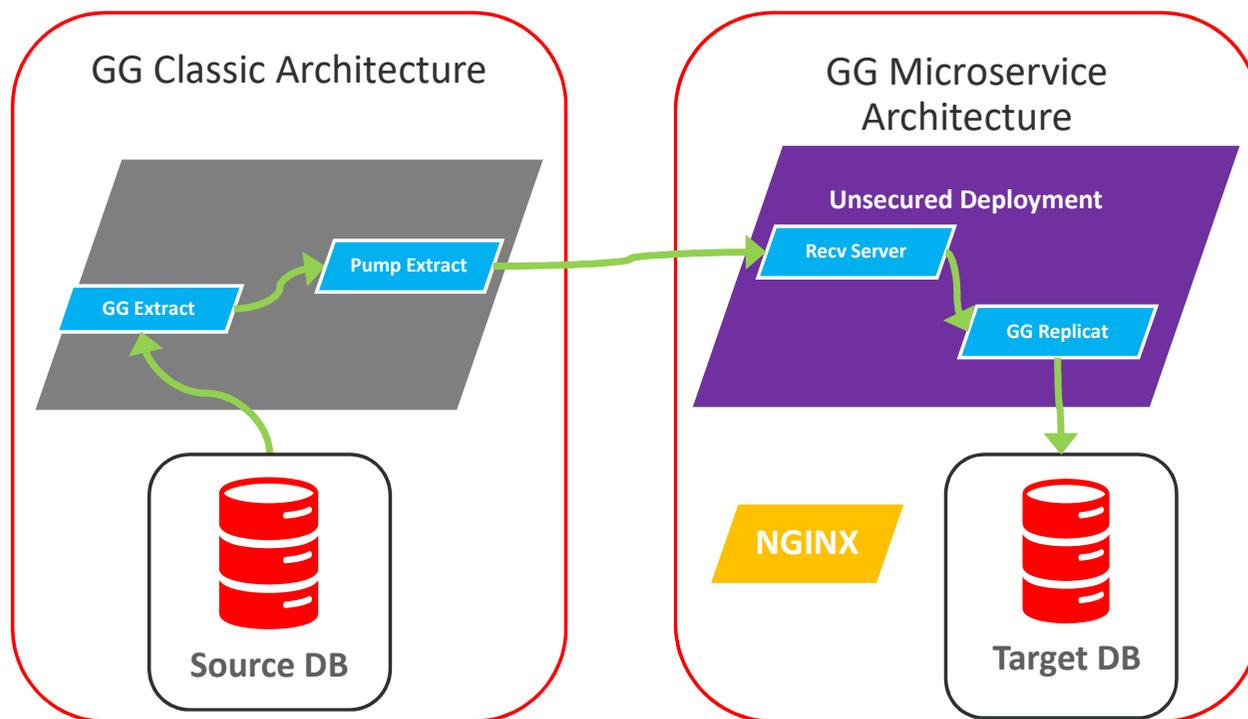
- Only one GoldenGate Instance
- Source DB and Target DB in same data center
- Remote Capture
- Remote Apply
- DB Requirement
  - Oracle DB 11g to 21c
  - SQLNet Access enabled for GG
- All data flow inside same Data Center
- No Distribution Path needed

# GG Microservice -> GG Microservice



- Use wss protocol with reverse proxy and UserIDAlias authentication
- If NGINX uses self-signed cert, need import NGINX certificate into Source CA certs, otherwise, just need add root cert.
- The CN name or SAN name in NGINX certificate should be routable to reach the Target GG MA
  - Can update local /etc/hosts on the source
- Alias authentication needed
  - Create an operator user on target GG MA
  - Add the login credential to the credential store on the source

# GG Classic → GG Microservice

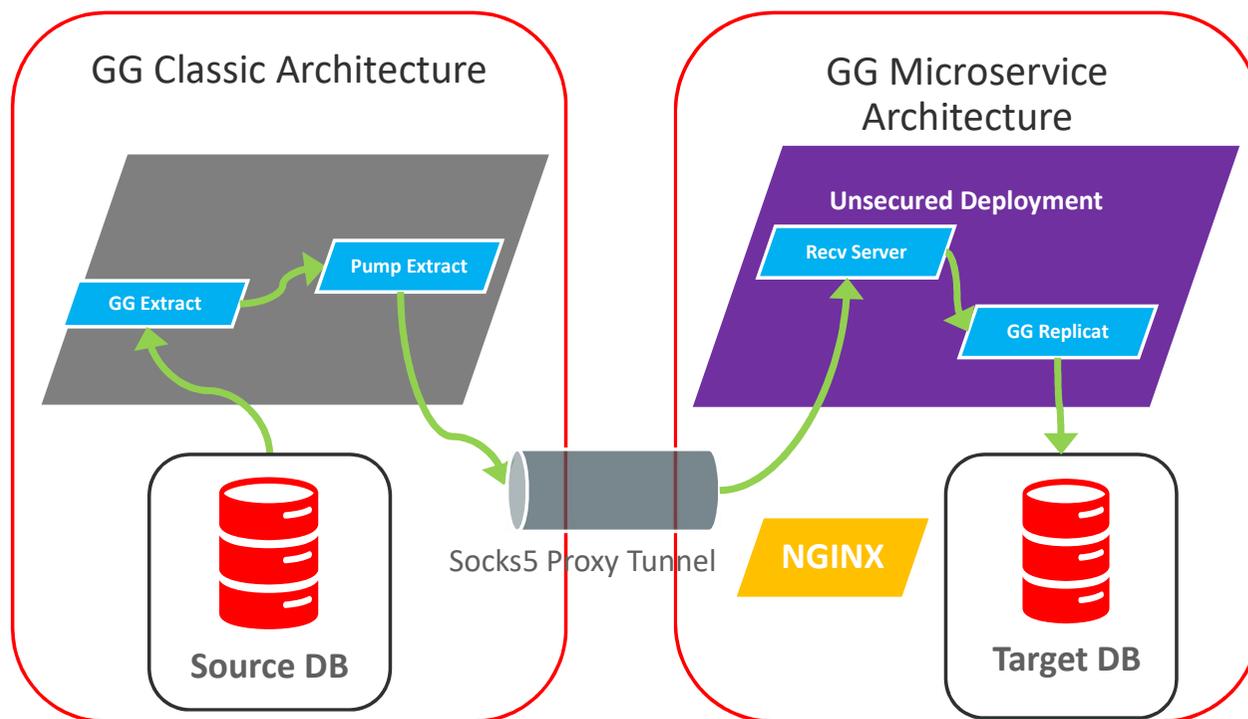


- GG MA needs use unsecured deployment
- Needs modify firewall to allow ingress traffic to receiver server port on GG MA
- GG Classic pump Extract uses “port” parameter

```
rmthost <gg_ma_host>, port  
<receiver_server_port>
```

- Highly recommend trail file encryption on the source
- FastConnect/VPN connection recommended if traffic goes through public internet

# GG Classic → GG Microservice (socks5 proxy tunnel)



- GG MA needs use unsecured deployment
- GG Classic pump Extract uses “port” and socksproxy parameter

```
rmthost <gg_ma_host>, port  
<receiver_server_port>, socksproxy  
<socks_proxy_ip>:<port>
```

- No need to open any GG MA ports for ingress policy
- SSH connection to the GG MA server needs to be open to create the Socks5 Proxy Tunnel

# Today's Agenda

- GoldenGate Platform
- Microservices Architecture - Why
- WebUI – For New Generation
- Distribution Path – Bridges between Deployments
- Adminclient and RestAPI – For Old Timer
- Docker Implementation – Easy for SYS Admins

## Adminclient vs GGSCI

- No GGSCI in Microservice architecture/微服务架构中没有GGSCI
- What's similar/有什么相似之处
  - Both are command interface that can manage GoldenGate instance/两者都是可以管理GoldenGate实例的命令界面
  - Almost all GGSCI commands can run without any change in Adminclient/几乎所有GGSCI命令都可以运行，而无需对Adminclient进行任何更改
- What's adminclient's advantage/adminclient的优势是什么
  - GG User Authentication, different users, different roles, more secure/GG用户身份验证，不同用户，不同角色，更安全
  - Running remotely connecting to a
  - Supports Linux style command hi  
Linux风格的命令历史记录和调用，



## Adminclient Example

```
[ogg@oggma213dk bin]$ ./adminclient
Oracle GoldenGate Administration Client for Oracle
Version 21.3.0.0.0 OGGCORE_21.3.0.0.0_PLATFORMS_210728.1047
Copyright (C) 1995, 2021, Oracle and/or its affiliates. All rights reserved.
Oracle Linux 7, x64, 64bit (optimized) on Jul 28 2021 12:32:46
Operating system character set identified as US-ASCII.
OGG (not connected) 2> connect https://localhost:443 as oggadmin !
Password for 'oggadmin' at 'https://localhost':
Using default deployment 'oggma213dk'
OGG (https://localhost oggma213dk) 3> info all
```

Program	Status	Group	Type	Lag at Chkpt	Time Since Chkpt
ADMINSRVR	RUNNING				
DISTRVR	RUNNING				
PMSRVR	RUNNING				
RECVSRVR	RUNNING				
EXTRACT	RUNNING	EDB19MAC	INTEGRATED	00:00:02	00:00:05



## Adminclient Example - continued

OGG (https://localhost oggma213dk) 4> info extract EDB19MAC, detail

Extract EDB19MAC Last Started 2021-09-02 03:26 Status RUNNING

Description Database 19c running on Mac Docker

Checkpoint Lag 00:00:00 (updated 00:00:00 ago)

Process ID 591

Log Read Checkpoint Oracle Integrated Redo Logs

2021-09-02 05:09:22

SCN 0.3656202 (3656202)

Settings Profile Default

Encryption Profile LocalWallet

Target Extract Trails:

Trail Name	Seqno	RBA	Max MB	Trail Type
et	4	1345	50	EXTTRAIL

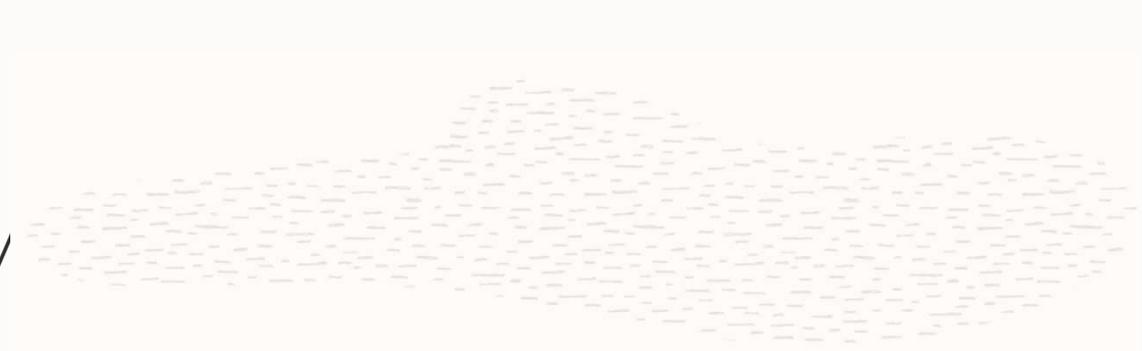
Integrated Extract outbound server first scn: 0.3016529 (3016529)

Integrated Extract outbound server filtering start scn: 0.3016840 (3016840)

Extract Source	Begin	End
Not Available	2021-08-19 03:58	
2021-09-02 05:09		
Not Available	* Initialized *	2021-08-19 03:58
Current directory /		
Report file		
/u02/Deployment/var/lib/report/EDB19MAC.rpt		
Parameter file		
/u02/Deployment/etc/conf/ogg/EDB19MAC.prm		
Checkpoint file		
/u02/Deployment/var/lib/checkpt/EDB19MAC.cpe		
Process file		
/u02/Deployment/var/run/EDB19MAC.pce		
Error log	/u02/Deployment/var/log/ggserr.log	

# RestAPI

- Talk directly to the microservices endpoints
- Reference Guide, <https://docs.oracle.com/en/middleware/goldengate/core/21.3/oggra/>
- Support any RestAPI client
  - Curl
  - Postman
- Can embed in all scripts/languages that supports curl
  - Sql script
  - Java
  - Python
  - Shell scripts
- Supported Methods
  - GET/HEAD/POST/PUT/PATCH/OPTIONS/



## Example REST Service Call to Create Extract

Single Call to create, update parameter file, register, and create integrated Extract

POST <https://mytest.com/services/Source/adminsrvr/v2/extracts/UTEST>

JSON Payload:



```
{
  "credentials": {
    "domain": "OracleGoldenGate",
    "alias": "GGSRC"
  },
  "intent": "Unidirectional",
  "status": "stopped",
  "begin": "now",
  "managedProcessSettings": "Default",
  "encryptionProfile": "LocalWallet",
  "source": {
    "tranlogs": "integrated"
  },
  "registration": {
    "share": false,
    "containers": [
      "\"WDBOCI2_PDB1\""
    ]
  },
  "targets": [
    {
      "name": "ww",
      "remote": false,
      "sequence": 0,
      "sizeMB": 500,
      "offset": 0
    }
  ],
  "config": [
    "extract UTEST",
    "useridalias GGSRC domain OracleGoldenGate",
    "exttrail ww",
    "table WDBOCI2_PDB1.U1.*;"
  ]
}
```

## Example Rest Service Call to get all services status from pmsrvr

- Curl Command

```
curl --cacert wggma213dk.pem -K access.cfg  
https://localhost:3443/services/oggma213dk/pmsrvr/v2/mpoints/processes -XGET|python -m  
json.tool
```

- Options explained

--cacert, CA root cert or self-signed cert when GG/NGINX uses self-signed cert

-K, the config file that has the userid and password, the content will be  
user = "<user>:<password>"

https:// ....., The GG Service endpoint for this call

-XGET, method to be applied is GET

|python -m json.tool, beautify the output for human to read

# Example Rest Service Call to get all services status from pmsrvr -- Response

```
"response": {
  "$schema": "mpoints:processes",
  "processes": [
    {
      "firstMessage": 0,
      "lastHeartbeat": "2021-09-02T03:50:17.579Z",
      "lastMessage": 0,
      "portNumber": 9012,
      "processId": 68,
      "processMode": "Not Applicable",
      "processName": "ADMINSRVR",
      "processState": "Running",
      "processType": "AdminServer",
      "startTime": "2021-09-02T03:04:54.600Z",
      "stateTime": "2021-09-02T03:04:54.614Z"
    },
    {
      "firstMessage": 0,
      "lastHeartbeat": "2021-09-02T03:50:16.943Z",
      "lastMessage": 0,
      "portNumber": 9013,
      "processId": 66,
      "processMode": "Not Applicable",
      "processName": "DISTSVR",
      "processState": "Running",
      "processType": "DistServer",
      "startTime": "2021-09-02T03:04:55.238Z",
      "stateTime": "2021-09-02T03:04:55.264Z"
    }
  ],
}
```

```
{
  "firstMessage": 681,
  "lastHeartbeat": "2021-09-02T03:50:16.156Z",
  "lastMessage": 683,
  "portNumber": 9015,
  "processId": 70,
  "processMode": "Not Applicable",
  "processName": "PMSRVR",
  "processState": "Running",
  "processType": "PMServer",
  "startTime": "2021-09-02T03:04:54.713Z",
  "stateTime": "2021-09-02T03:04:54.732Z"
},
{
  "firstMessage": 0,
  "lastHeartbeat": "2021-09-02T03:50:17.588Z",
  "lastMessage": 0,
  "portNumber": 9014,
  "processId": 72
}
]
```



# Today's Agenda

- GoldenGate Platform
- Microservices Architecture - Why
- WebUI – For New Generation
- Distribution Path – Bridges between Deployments
- Adminclient and RestAPI – For Old Timer
- Docker Implementation – Easy for SYS Admins

## Why Docker/为什么使用Docker

- Easy to implement/易于实施
  - No need to do the installation from scratch/无需从头开始安装
  - Isolate from the host OS/与主机操作系统隔离
  - Smaller footprint compared to VM solutions as many resource shared at OS abstract level/与虚拟机解决方案相比，占用空间更小，因为在操作系统抽象级别共享了大量资源
  - Portable deployment/便携式部署
- Mapping volume directly to the host OS file system makes is easier to manage GG patching and upgrading/将卷直接映射到主机操作系统文件系统使GG修补和升级更容易管理
- With GG V21.3, one container can deal with DB from 11.2.04 to 21c/使用GG V21.3，一个容器可以处理11.



# Docker Image Build

- Docker Build files available on GIT, under Oracle Docker images  
<https://github.com/oracle/docker-images/tree/main/OracleGoldenGate/21c>
- Download GoldenGate installation files from Oracle OTN download site  
<https://www.oracle.com/middleware/technologies/goldengate-downloads.html>
- Run Docker Build command, giving the GG installation file to INSTALLER argument  
\$ docker build --tag=oracle/goldengate\_microservices:21.3.0.0.0 \  
--build-arg INSTALLER=213000\_fbo\_ggs\_Linux\_x64\_Oracle\_services\_shiphome.zip .  
Sending build context to Docker daemon  
...  
Successfully tagged oracle/goldengat
- Check the image is successfully built  
\$ docker images oracle/goldengate-n  
REPOSITORY TAG  
oracle/goldengate-microservices 21.3.0.0.0 119100007107 2 weeks ago 2.12GB



## Run Docker Image

- Use docker run command to create the container

```
$ docker run \  
  --name <container name> \  
  -p <host port>:443 \  
  -e OGG_ADMIN=<admin user name> \  
  -e OGG_ADMIN_PWD=<admin password> \  
  -e OGG_DEPLOYMENT=<deployment name> \  
  -v [<host mount point>:]/u02 \  
  -v [<host mount point>:]/u03 \  
  -v [<host mount point>:]/etc/nginx/cert \  
  oracle/goldengate_microservices:21.3.0.0.0
```

- Parameters

<container name> - A name for the new container (default: auto generated)

-p <host-port> - The host port to map to the Oracle GoldenGate HTTPS server (default: no mapping)

-e OGG\_ADMIN - The name of the administrative account to create (default: oggadmin)

-e OGG\_ADMIN\_PWD - The password for the administrative account (default: auto generated)

-e OGG\_DEPLOYMENT - The name of the deployment (default: Local)

-v /u02 - The volume used for persistent GoldenGate data (default: use container storage)

-v /u03 - The volume used for temporary GoldenGate data (default: use container storage)

-v /etc/nginx/cert - The volume used for storing the SSL certificate for the HTTPS server (default: create a self-signed certificate)



## Docker Run Example

```
$docker run --name oggma213dk -p 3443:443 -h oggma213dk \  
-e OGG_ADMIN=oggadmin -e OGG_ADMIN_PWD=Uguess! -e OGG_DEPLOYMENT=oggma213dk \  
-v /Users/werhe/Documents/Software/OracleDocker/OGG213Docker/u02:/u02 \  
-v /Users/werhe/Documents/Software/OracleDocker/OGG213Docker/u03:/u03 \  
-v /Users/werhe/Documents/Software/OracleDocker/OGG213Docker/cert:/etc/nginx/cert \  
-d oracle/goldengate-microservices:21.3.0.0
```

```
$ docker ps -f name=oggma213dk
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS	NAMES
abaa91df5aba (healthy)	oracle/goldengate-microservices:21.3.0.0	"/usr/local/bin/depl..."	14 hours ago	Up	About an hour	oggma213dk



## Attach to the container and run adminclient

```
$ docker ps
```

CONTAINER ID	IMAGE	COMMAND	CREATED	STATUS	PORTS
abaa91df5aba	oracle/goldengate-microservices:21.3.0.0.0	"/usr/local/bin/depl..."	24 hours ago	Up 2 hours (healthy)	0.0.0.0:3443->443/tcp
78f6aee41b10	oracle/database:19.3.0-ee	"/bin/sh -c 'exec \$O..."	6 months ago	Up 2 hours (healthy)	0.0.0.0:9521->1521/tcp, 0.0.0.0:9500->5500/tcp

```
$ docker exec -it oggma213dk /bin/bash
```

```
[root@oggma213dk /]# su - ogg
```

```
Last login: Wed Sep 1 19:14:28 UTC 2021 on pts/0
```

```
[ogg@oggma213dk ~]$ cd /u01/ogg/bin
```

```
[ogg@oggma213dk bin]$ ./adminclient
```

```
Oracle GoldenGate Administration Client for Oracle
```

```
Version 21.3.0.0.0 OGGCORE_21.3.0.0.0_PLATFORMS_2
```

```
Copyright (C) 1995, 2021, Oracle and/or its affiliates. All rights reserved.
```

```
Oracle Linux 7, x64, 64bit (optimized) on Jul 28 2021 12:00:00
```

```
Operating system character set identified as US-ASCII.
```



## Attach to the container and run adminclient - continued

OGG (not connected) 2> connect https://localhost:443 as oggadmin !

Password for 'oggadmin' at 'https://localhost':

Using default deployment 'oggma213dk'

OGG (https://localhost oggma213dk) 3> info all

Program	Status	Group	Type	Lag at Chkpt	Time Since Chkpt
---------	--------	-------	------	--------------	------------------

ADMINSRVR	RUNNING				
-----------	---------	--	--	--	--

DISTRVR	RUNNING				
---------	---------	--	--	--	--

PMSRVR	RUNNING				
--------	---------	--	--	--	--

RECVSRVR	RUNNING				
----------	---------	--	--	--	--

EXTRACT	RUNNING	EDB19MAC	INTEGRATED	00:00:02	00:00:05
---------	---------	----------	------------	----------	----------

OGG (https://localhost oggma213dk) 4>

