Oracle OpenWorld 2019

SAN FRANCISCO

Safe Harbor

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.

Statements in this presentation relating to Oracle's future plans, expectations, beliefs, intentions and prospects are "forward-looking statements" and are subject to material risks and uncertainties. A detailed discussion of these factors and other risks that affect our business is contained in Oracle's Securities and Exchange Commission (SEC) filings, including our most recent reports on Form 10-K and Form 10-Q under the heading "Risk Factors." These filings are available on the SEC's website or on Oracle's website at http://www.oracle.com/investor. All information in this presentation is current as of September 2019 and Oracle undertakes no duty to update any statement in light of new information or future events.

Exploring the Multicloud: Working with Azure and Oracle Autonomous Database

Santanu Datta, Vice President of Development, Oracle
Christian Shay, Senior Principal Product Manager, Oracle
Alex Keh, Senior Principal Product Manager, Oracle
Nirmala Sundarappa, Principal Product Manager, Oracle

September 2019

Agenda

- Setting up Azure to Autonomous Database connectivity

- Application Connectivity from Azure to Autonomous Database

- Visual Studio integration and .NET application deployment (Demo)

- Eclipse integration and Java application deployment (Demo)

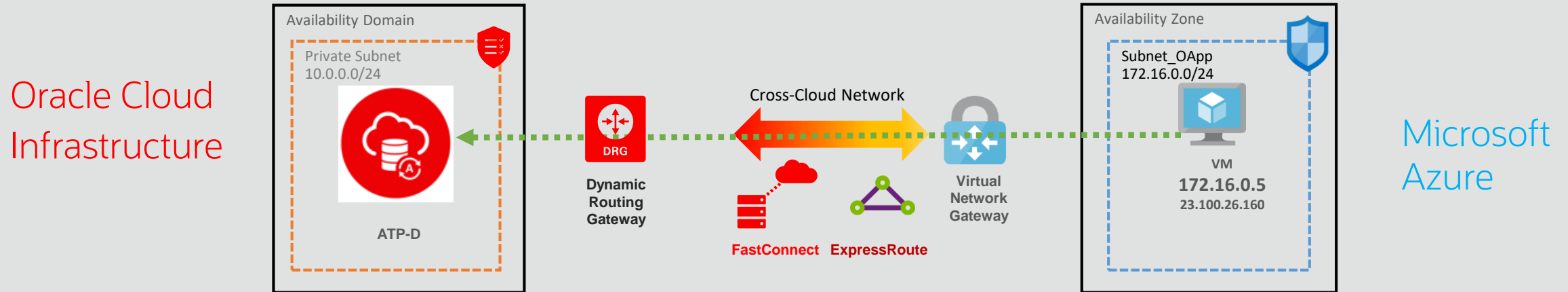- Database integration with Active Directory

Setting up Azure to Autonomous Database connectivity

Azure to ADB-Dedicated Connectivity
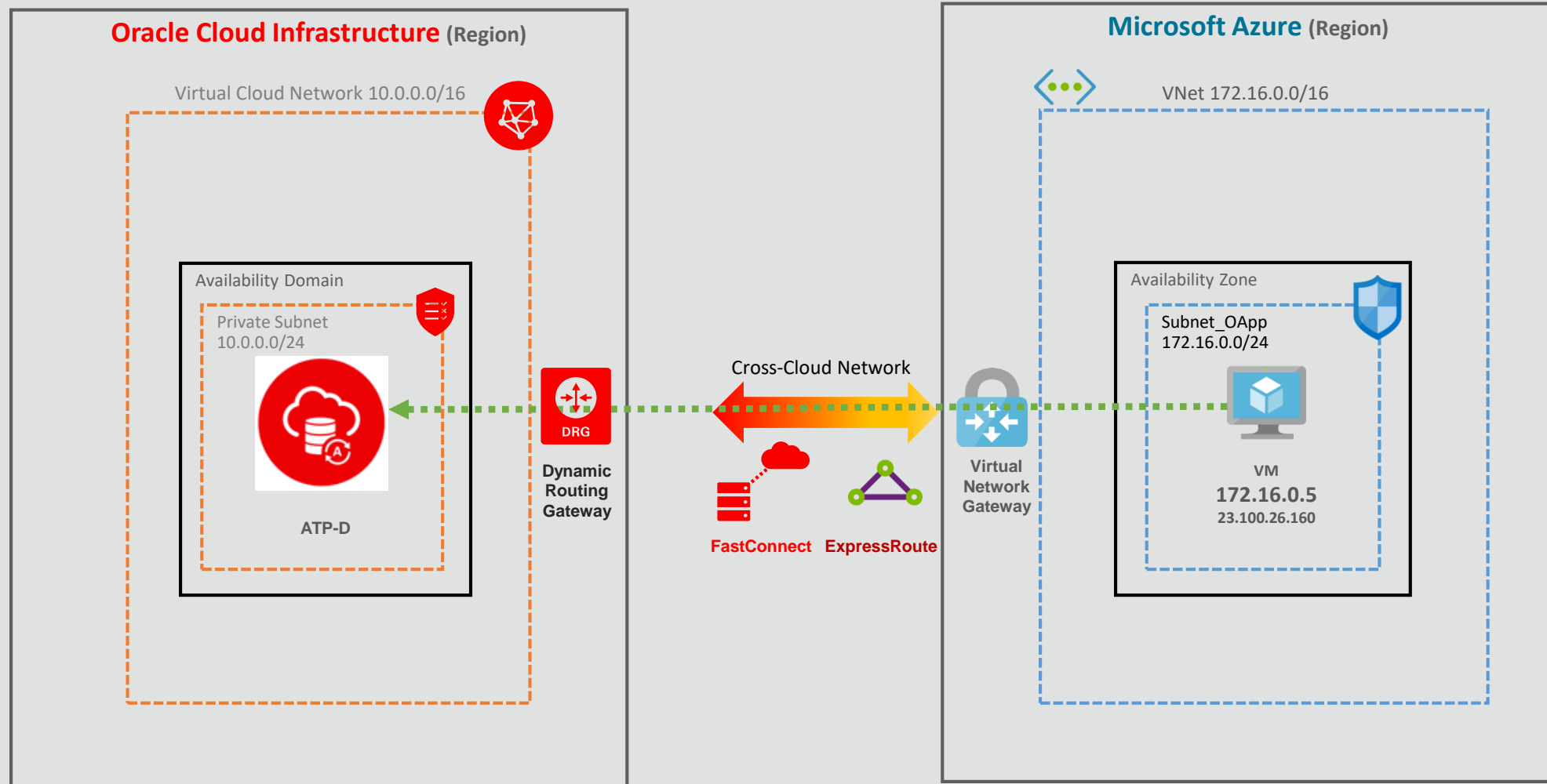
Azure to ADB-Dedicated Connectivity

Building Multicloud Applications on Microsoft Azure and Oracle Cloud Infrastructure (Sam Shah)

Configure Security List
- Step 1. From OCI Console, note down CIDR block of the VCN and CIDR block of the subnet for Autonomous Exadata Infrastructure; these will be needed for Azure configuration.
- Step 2. Go to your VCN and subnet and add security list to allow clients on Azure to access the database
  - Port numbers: 1521 (TCP), 2484(TCPS), 443 (HTTPS)
  - Source CIDR block is the same as the CIDR block of the VNet you are using on Microsoft Azure

Configure Route Table
- Step 3. Destination: CIDR address of the VNet on Microsoft Azure, Target type: Dynamic Routing Gateways

Configure Hybrid DNS in OCI (optional)

Configure Network Security Group
- Step 4. Add a rule to allow tcp traffic on ports 1521, 2484 and 443 between subnets on Azure and CIDR block of the subnet (noted in Step 1).
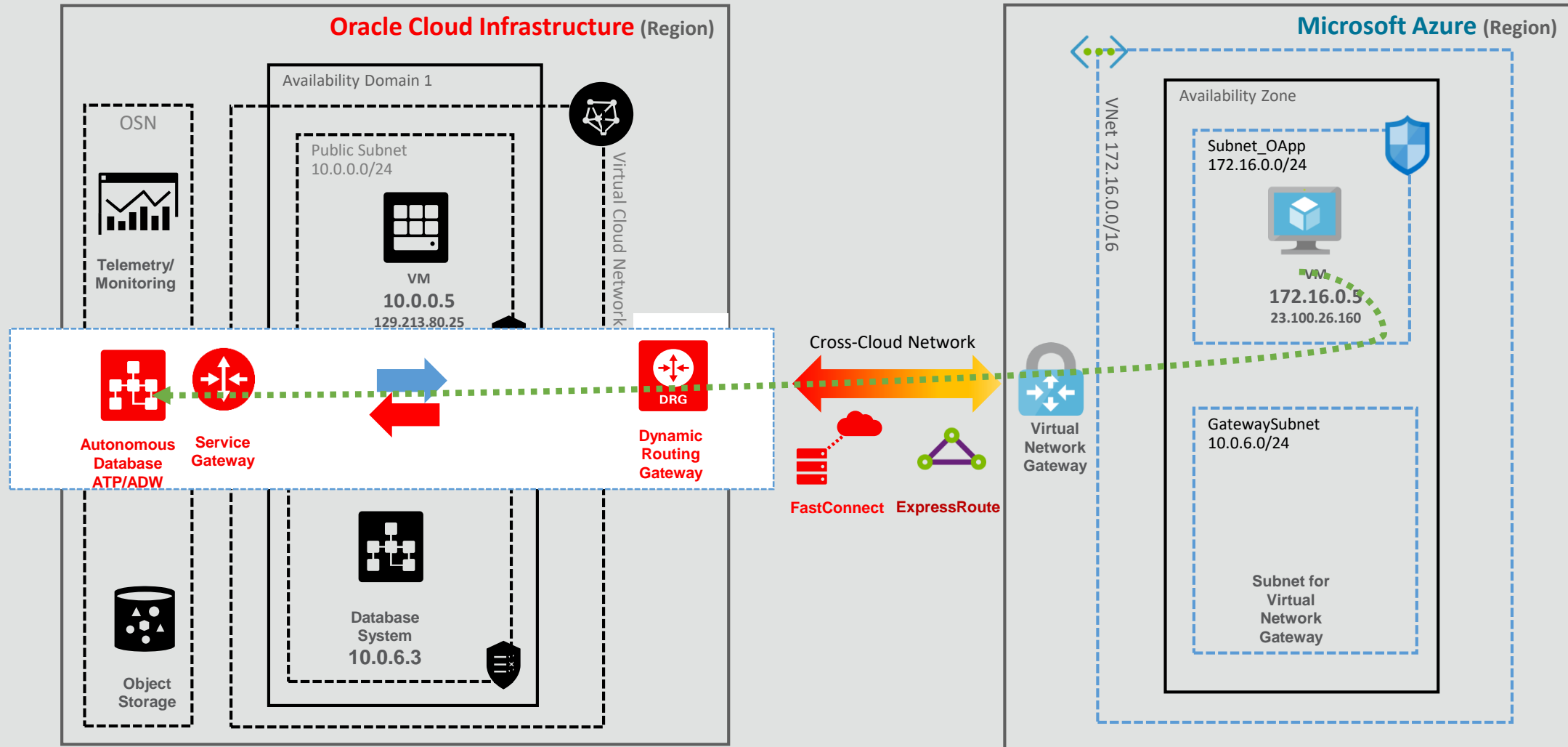
Configure Route Table
- Step 5. Go to the same subnet you opened up to allow traffic in Azure and add a new route that will send the traffic to the CIDR block of the subnet (noted in Step 1).

Pre-requisite: *How to setup the interconnect between Oracle Cloud Infrastructure and Microsoft Azure*
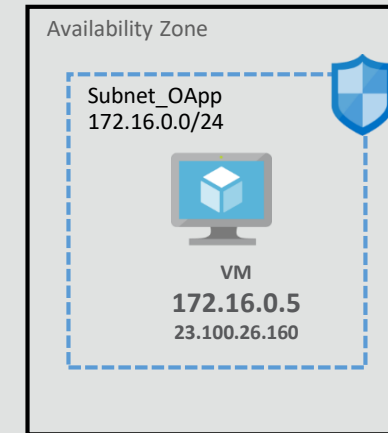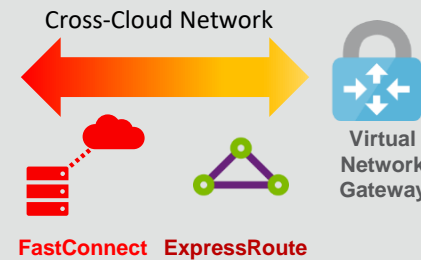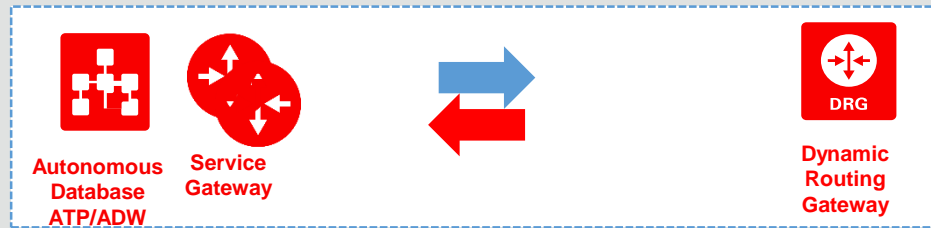
Azure to ADB-Serverless Connectivity

Azure to ADB-Serverless Connectivity

[Building Multicloud Apps on Microsoft Azure and Oracle Cloud Infrastructure (Vinay Rao)](#)

Oracle Cloud Infrastructure

Microsoft Azure

Create Service Gateway
- Choose your VCN and create a service gateway to reach All IAD Services in Oracle Services Network.

Create DRG Route Table
- Specify target type as Service Gateway and the destination service as All IAD Services in Oracle Services Network.
- Associate this route table with the DRG.

Create SGW Route Table
- Specify target type as Dynamic Routing Gateway and destination CIDR block as the address of the Azure Express route.
- Associate this route table with Service Gateway.

Configure network security groups
- Configure the network security groups to allow the required database port (1522 for TCPS) and other connections.

Pre-requisite: [How to setup the interconnect between Oracle Cloud Infrastructure and Microsoft Azure](#)
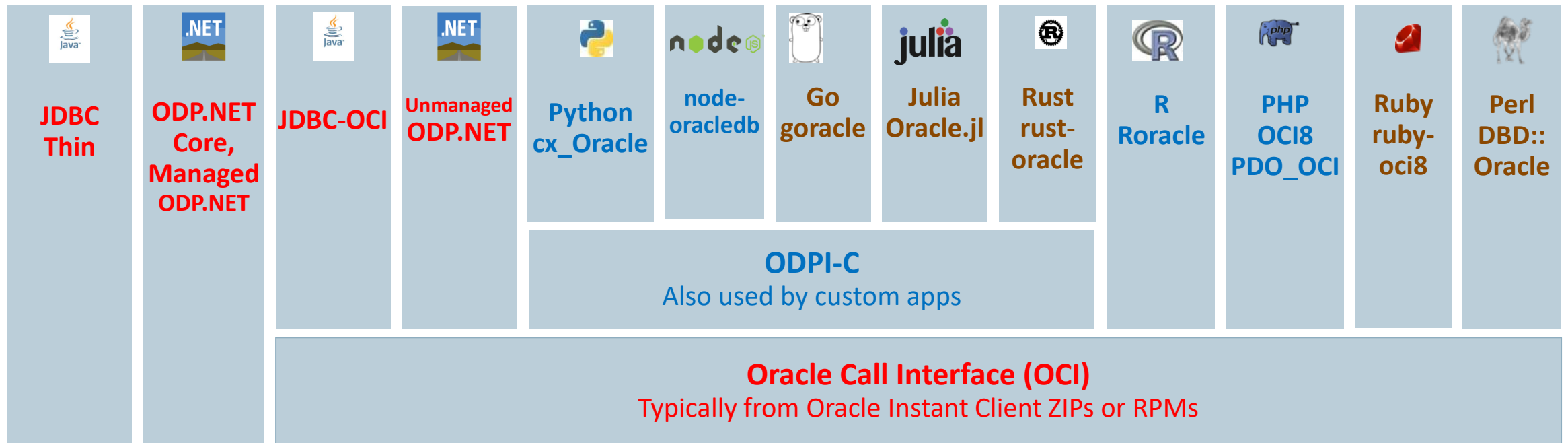
Application connectivity from Azure to Autonomous Database

Application Connectivity: Azure to Autonomous Database

- Oracle Database Drivers
- Connectivity options from Azure to Autonomous Database
- Database connectivity configurations
- Database Proxy

Oracle Database Drivers

| JDBC Thin | ODP.NET Core, Managed ODP.NET | JDBC-OCI | Unmanaged ODP.NET | Python cx_Oracle | node-oracledb | Go goracle | Julia Oracle.jl | Rust rust-oracle | R Roracle | PHP OCI8 PDO_OCI | Ruby ruby-oci8 | Perl DBD::Oracle |

ODPI-C
Also used by custom apps

Oracle Call Interface (OCI)
Typically from Oracle Instant Client ZIPs or RPMs

Oracle Open Source Drivers

Third Party Open Source Drivers

Oracle Proprietary Drivers
Oracle Call Interface, Oracle C++ Call Interface, ODBC, JDBC, ODP.NET, Pro*C, Pro*COBOL, SQLJ, OLE DB, OLE DB for OLAP

Distribution Channels

Instant Client →
RPMs on yum, ULN and Cloud OCI (yum install ..)

OTN no click through

JDBC →
Oracle Maven

Maven Central

.NET and VS Plugins →
NuGet binaries
Visual Studio Marketplace
Visual Studio Code Marketplace

Scripting →
GitHub (source)
RPMs on yum, ULN & Cloud OCI (yum install ..)
PyPI binaries, npm.org binaries

ODPI-C →
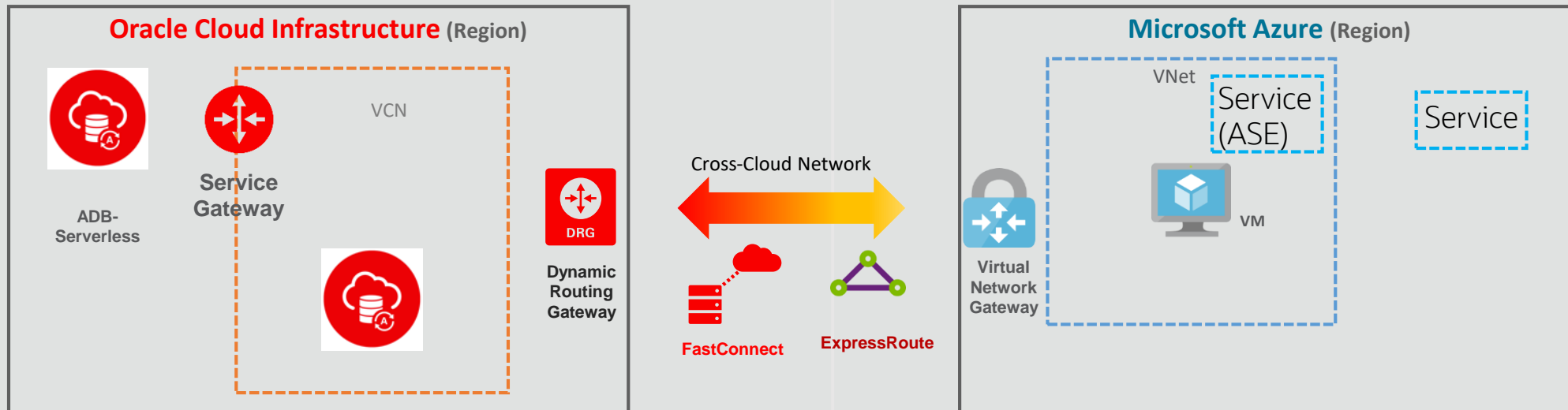GitHub (source)

Docker →
Docker Hub images for Server and Clients

GitHub (Dockerfiles)

Connectivity options: Azure to Autonomous Database

No ExpressRoute-FastConnect

- Azure VMs and Application Services connect to Autonomous Database over internet if the database allows internet connectivity.

With ExpressRoute-FastConnect

Oracle Cloud Infrastructure (Region)

VCN

ADB-Serverless

Service Gateway

DRG

Dynamic Routing Gateway

Cross-Cloud Network

FastConnect   **ExpressRoute**

Microsoft Azure (Region)

VNet

Service (ASE)

Service

VM

Virtual Network Gateway

ADB-Dedicated

- VMs in the VNet connect using Private IP

- Application Services in App Service Environment - Isolated Mode (running in the same VNet) connect using Private IP (Testing to be done)

- Application Services in other modes which support VNet integration (To be investigated/tested)

ADB-Serverless

- VMs in the VNet connect using Public IP

- Application Services in App Service Environment - Isolated Mode (running in the same VNet) connect using Public IP (Testing to be done)

- Application Services in other modes which support VNet integration (To be investigated/tested)

Integrate your app with an Azure Virtual Network

Database connectivity: Azure to Autonomous Database
Security Considerations

- Secure Autonomous Database using firewall rules
  - Restrict access to specific IP addresses
  - Restrict access to specific VCN and IP addresses (Planned)
- Use TLS based Network Connections
  - For ADB-Serverless, only TLS based connection is supported
  - For ADB-Dedicated, choose an alias from TNSNAMES.ORA that uses TLS
- For IIS based Application Servers on Azure, enable the use of local wallets ✶
  - Azure VM
    - Configure your local IIS attribute Load User Profile to "true".
  - Azure App Service (Basic and higher)
    - Under Settings, click on the Configuration link, then click the Application settings tab.
    - Add a new application setting, WEBSITE_LOAD_USER_PROFILE, and set it to 1.

Connecting Azure Web Apps to Oracle Autonomous Database (Alex Keh)
Create, Deploy, and Run Java apps connecting to ATP on Azure (Nirmala Sundarappa)

Database connectivity: Azure to Autonomous Database
Recovery from network errors

- If you are NOT using Azure ExpressRoute-OCI Fast Connect, connections may be terminated after a few minutes of idle time
  - Enable Keep Alive to protect against such terminations
    - EXPIRE_TIME in minutes
- Use retries and timeouts
  - Generally a good practice
  - Very useful if you are not using Azure ExpressRoute-OCI Fast Connect
  - Connection establishment timeouts
    - Timeout for TCP connection establishment (TRANSPORT_CONNECT_TIMEOUT)
      - Enabled by default to 60 seconds
  - Connection retries and retry delay
    - Retry count for TCP connection establishment (RETRY_COUNT)
    - Retry delay, default 0 (RETRY_DELAY)

Database connectivity: Azure to Autonomous Database
Recovery from network errors

- Example:

oracletestdb_high =
(description=
    (address=(protocol=tcps)(port=1522)(host=adb.us-ashburn-1.oraclecloud.com))
    (expire_time=2)
    (transport_connect_timeout=10)
    (retry_count=3)(retry_delay=5)
    (connect_data=(service_name=ej8s7jqvzjvpq8j_oracletestdb_high.atp.oraclecloud.com))
    (security=(ssl_server_cert_dn="CN=adwc.uscom-east-1.oraclecloud.com,OU=Oracle
BMCS US,O=Oracle Corporation,L=Redwood City,ST=California,C=US"))

Note:

EXPIRE_TIME is only available in certain driver versions. You may need to set this at OS level or driver property level for some of the drivers.

Database connectivity: Azure to Autonomous Database
JDBC Examples

- Connect Timeout set through property oracle.net.CONNECT_TIMEOUT
- Read Timeout set through oracle.jdbc.ReadTimeout
  - Note: Do not use as a query-timeout.
- For Query Timeout, use Statement.cancel or Statement.setQueryTimeout
- How to set properties?
  1. ojdbc.properties file**(18c)**
  2. specify in code

```
Code Example:
Properties prop = new Properties();
prop.setProperty("user","scott");
prop.setProperty("password","tiger");
prop.setProperty("oracle.net.CONNECT_TIMEOUT","3000");
prop.setProperty("oracle.jdbc.ReadTimeout","3000");
Conn=(new oracle.jdbc.OracleDriver()).connect(url,prop);
```

Database connectivity: Azure to Autonomous Database
ODP.NET Examples

- Connection Timeout set through connection string (Default 15 Sec)

  "User Id=HR; Password=hr; Connection Timeout=20; Data Source=ORCLPDB1"

- For Command Timeout, use

  OracleConfiguration.CommandTimeout (sets it for the application level)

  `OracleCommand.` CommandTimeout (sets it for the individual command object)

- Enable KeepAlive on Oracle Connection

  OracleConnection.KeepAlive

  OracleConnection.KeepAliveTime

  OracleConnection.KeepAliveInterval

- How to set properties?

  1. specify in code ⟶

```
Code Example:
String cs = "User Id = HR; Password = hr; Connection
Timeout = 20; Data Source = ORCLPDB1";
OracleConnection ocon = new OracleConnection(cs);
ocon.KeepAlive = true;
ocon.KeepAliveTime = 120;
ocon.KeepAliveInterval = 10;
ocon.Open();
OracleCommand oc = new OracleCommand();
oc.Connection = ocon;
oc.CommandTimeout = 5;
```

Database Proxy
Oracle Connection Manager (CMAN)

Clients

Application Servers
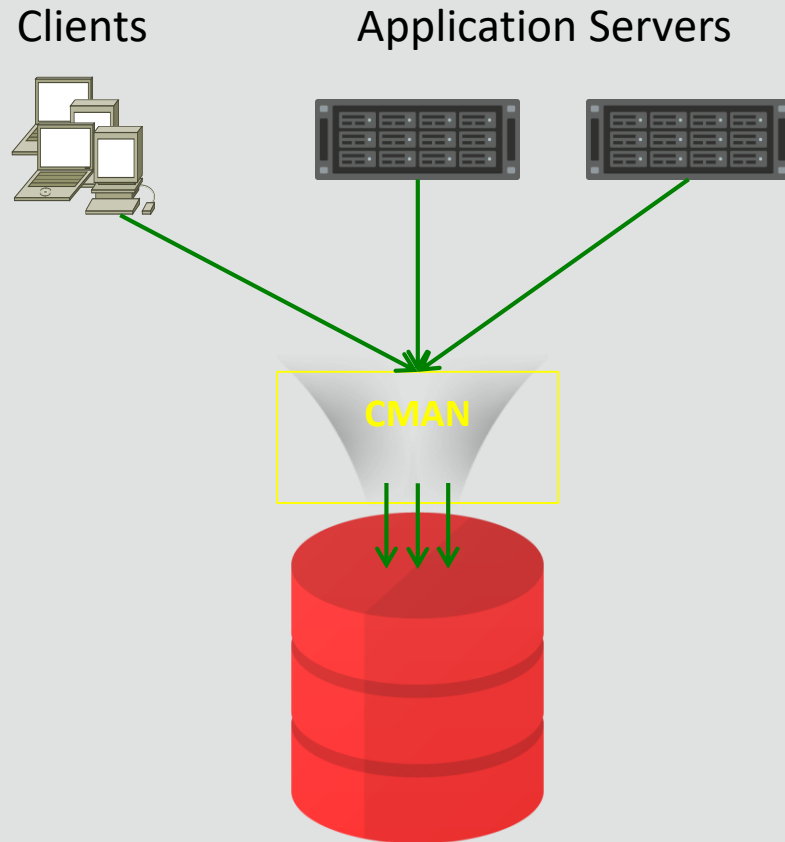
CMAN

- CMAN is an existing standard network proxy used in Oracle deployments
- Supports all SQL*Net protocols as well as protocol conversion
  - Can be used as a bridge
- Aware of Oracle database services
  - Fully Transparent: no application changes required
  - Leverages TNS Listener and cross-registration
  - Auto-updates when service topology changes
- CLI for management
  - Configurable using cman.ora
- Provides access control to services based on configurable rule lists
  - Used as a firewall proxy in ADW, EECS

Demo

Visual Studio integration and .NET
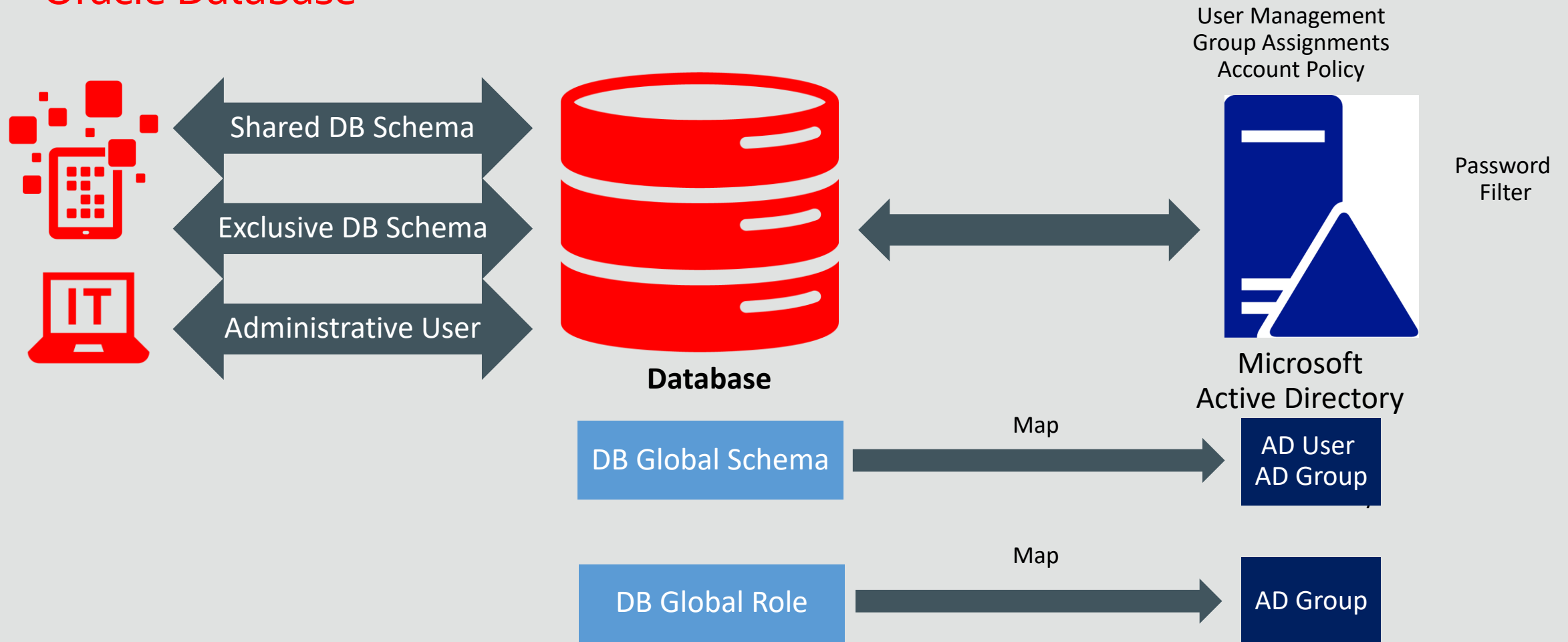Application Deployment

Demo

Eclipse integration and Java
Application Deployment

Active Directory Integration

Active Directory integration

Oracle Database

User Management
Group Assignments
Account Policy

Shared DB Schema

Exclusive DB Schema

Administrative User

Database

Password
Filter

Microsoft
Active Directory

DB Global Schema — Map → AD User / AD Group

DB Global Role — Map → AD Group

Active Directory integration

Oracle Database

Database

Global user:
HR_RUNTIME

Global Role:
HR_MGR

Map:
Global user HR_RUNTIME to AD Group hr-rep
Global role HR_MGR to AD Group hr-mgr

CREATE USER HR_RUNTIME IDENTIFIED GLOBALLY AS
'cn=

CREATE ROLE HR_MGR IDENTIFIED GLOBALLY AS
'cn=hr-mgr,ou=hr,dc=examplecorp,dc=com';
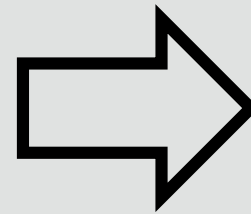
Directory

Domain (dc=examplecorp, dc=com)
cn = Users

Users:
    Susan, Diana, Jennifer

Groups:
– hr–rep {*Susan, Diana, Jennifer* }
– hr–mgr {*Susan* }

Active Directory integration

Autonomous Database

- Under investigation

- Would like to understand the interest in Active Directory integration for multicloud deployments.