Deploy the Oracle Cloud and Microsoft Azure Interconnect Using Hub-and-Spoke Topology

Access Oracle Cloud Infrastructure Resources from Microsoft Azure over a Private Interconnect
PURPOSE STATEMENT

This document provides an overview of the capabilities enabled by the cross-cloud direct interconnection between Oracle Cloud Infrastructure and Microsoft Azure Interconnect. It’s solely intended to help you assess the business benefits of migrating workloads to Oracle Cloud Infrastructure and plan your IT projects.

DISCLAIMER

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. 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, and timing of any features or functionality described in this document remains at the sole discretion of Oracle.

Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

REVISION HISTORY

The following table shows the publication and revision history of this document:

<table>
<thead>
<tr>
<th>DATE</th>
<th>REVISION</th>
</tr>
</thead>
<tbody>
<tr>
<td>September 30, 2020</td>
<td>Initial publication</td>
</tr>
</tbody>
</table>
# Table of Contents

Introduction ................................................................................................................. 4  
Solution Summary ...................................................................................................... 4  
Prerequisites and Considerations .............................................................................. 5  
Deploying the Topology ............................................................................................. 5  
  Step 1: Set Up the Hub VCN .......................................................................................... 5  
  Step 2: Set Up the Spoke VCN ....................................................................................... 13  
  Step 3: Set Up the Azure Environment ......................................................................... 22  
  Step 4: Connect Oracle Cloud Infrastructure and Azure Using the Private Interconnect  27  
  Step 5: Access Autonomous Database from Azure Through the Private Interconnect ... 34  
Conclusion ................................................................................................................. 36  
References ................................................................................................................. 36
INTRODUCTION

The partnership between Oracle and Microsoft provides cloud interoperability that enables customers to distribute mission-critical enterprise workloads between the two cloud providers. This interoperability provides many opportunities for customers to deploy their applications across Oracle Cloud Infrastructure and Microsoft Azure simultaneously. A highly redundant, low-latency, high-bandwidth private connection provides a cross-cloud direct interconnection between Oracle Cloud Infrastructure and Azure in various regions across the globe. This interconnection is achieved by connecting Oracle Cloud Infrastructure FastConnect directly with Azure ExpressRoute without an intermediary network provider.

In a hub-and-spoke model, the hub is a virtual network that serves as a central location for managing external connectivity either to on-premises resources or other cloud vendors. The spokes are virtual networks that host applications and provide connectivity through a central hub by way of local peering gateways. Traffic passing through the hub can be routed, inspected, and centrally managed according to rules and processes.

Companies are increasingly using the concept of transit routing to isolate and compartmentalize their resources into dedicated spoke virtual networks. This configuration makes it easy to comply with governance and security guidelines because each spoke might have access to different levels of services and access controls.

A dynamic routing gateway (DRG) on Oracle Cloud Infrastructure has a one-to-one relationship with a virtual network gateway (VNet gateway) on Azure. If a customer is managing multiple virtual cloud networks (VCNs) in Oracle Cloud Infrastructure, then the only way to connect all the VCNs to Azure is to deploy multiple interconnects to establish the application connectivity between Azure and Oracle Cloud Infrastructure. This setup soon becomes expensive and hard to manage. With transit routing, however, instead of multiple VCNs being managed separately, they can be architected in a hub-and-spoke topology. This topology provides a single source of connection to Azure.

This document describes how to distribute a typical application workload in Oracle Cloud Infrastructure in a hub-and-spoke topology and how to access it from Azure through the private interconnect.

SOLUTION SUMMARY

The example in this document uses an Oracle Autonomous Database that runs on shared Oracle Exadata infrastructure. The database resides in the private subnet of a spoke VCN and is accessed from Azure over a private interconnect using the private IP endpoint of the database. The connectivity to the spoke VCN is established by enabling local peering between the hub and spoke VCN.

The hub VCN in Oracle Cloud Infrastructure is connected to Azure through a private interconnect that uses ExpressRoute and FastConnect. The traffic between Oracle Cloud Infrastructure and Azure is directed by a DRG on Oracle Cloud Infrastructure and a VNet gateway on Azure.

Optionally, a bastion server can be provisioned in the public subnet of the hub VCN to connect to other sources that might be deployed in the spoke VCN. Also, more application servers can be deployed in the public subnet of a spoke VCN that might need connectivity to the database.
The setup is highly scalable. More spoke VCNs can be created later and connected to Azure through the hub VCN, using the existing private interconnect. Figure 1 illustrates this hub-and-spoke setup.

**PREREQUISITES AND CONSIDERATIONS**

Before starting to create resources, it's important to carefully plan and consider the following prerequisites:

- Select the region for the setup carefully. Oracle Cloud Infrastructure and Azure provide the private interconnect facility for various regions across the globe, and the list is always growing. Be sure to select a region that offers the option of connecting Oracle Cloud Infrastructure and Azure through the private interconnect. For a complete list of regions that support this connection, see [Access to Microsoft Azure](#).

- Carefully plan your network and always use non-overlapping CIDR blocks across Oracle Cloud Infrastructure and Azure.

- Identify a pair of /30 CIDR blocks for the Border Gateway Protocol (BGP) IP addresses to use for the two redundant BGP sessions between Oracle and Azure.

- Because you can add spoke VCNs after the initial setup, when the demand grows to host more resources, the configuration can become complex to manage and troubleshoot. Be sure to plan for future growth.

**DEPLOYING THE TOPOLOGY**

This section provides the detailed steps for setting up a hub-and-spoke topology in Oracle Cloud Infrastructure and accessing it from Azure through the private interconnect.

**Step 1: Set Up the Hub VCN**

The first step is to create and configure a hub VCN. The hub VCN is connected to a spoke VCN in Oracle Cloud Infrastructure and to Azure through the private interconnect.
Be sure to select a region where the interconnect is offered. This example uses the Japan East region. For a complete list of connected regions, see Access to Microsoft Azure.

![Select a Region for the Interconnect](image)

**Figure 2: Select a Region for the Interconnect**

1. In the Oracle Cloud Infrastructure Console navigation menu, select **Networking** and click **Virtual Cloud Networks**.
2. Click **Create Virtual Cloud Network**.
3. Enter a name for the VCN (for example, Hub_VCN), and specify the compartment to create it in (for example, TransitRouting).
4. Specify a CIDR address block for the hub VCN, ensuring it doesn’t overlap with the spoke VCN or the Azure VNet.

![Create a Virtual Cloud Network](image)

**Figure 3: Create the Hub VCN**
5. Click **Create VCN**.

6. Create a local peering gateway (for example, HubLPG) for the hub VCN. This gateway connects the hub VCN with the spoke VCN. For instructions, see **To create a local peering gateway**.

   ![Create Local Peering Gateway](image)

   Figure 4: Create the Local Peering Gateway for the Hub VCN

7. Create an internet gateway (for example, HubIGW) for the hub VCN. An internet gateway is required only if you are planning to use a bastion server that would need accessibility from the internet. For instructions, see **To set up an internet gateway**.

   ![Create Internet Gateway](image)

   Figure 5: Create an Internet Gateway for the Hub VCN
8. Create a DRG (for example, AzureDRG) for the hub VCN. In a later step, you attach it to FastConnect to provide connectivity to Azure. For instructions, see To create a DRG.

![Create Dynamic Routing Gateway](image1)

**Figure 6: Create a DRG for the Hub VCN**

9. Attach the DRG to the hub VCN. In general, a VCN can be attached to only one DRG at a time, and a DRG can be attached to only one VCN at a time. For instructions, see To attach a DRG to a VCN.

![AzureDRG](image2)

**Virtual Cloud Networks**

<table>
<thead>
<tr>
<th>Name</th>
<th>Lifecycle State</th>
<th>CIDR Block</th>
<th>Attachment State</th>
<th>Attachment Route Table</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hub_VCN</td>
<td>Available</td>
<td>10.1.15.1/24</td>
<td>Attached</td>
<td>10.1.0/24</td>
</tr>
</tbody>
</table>

![Attach to Virtual Cloud Network](image3)

**Figure 7: Attach the DRG to the Hub VCN**

10. Modify the default route table for the hub VCN by creating the following route rules. For instructions, see To update rules in an existing route table.

- Create a route rule of the target type **Dynamic Routing Gateways** that points the DRG that you created, which allows the route to the Azure CIDR.
- Create a route rule of the target type **Local Peering Gateway** that points to the local peering gateway that you created, which allows the traffic route to the spoke VCN.
- If you created an internet gateway, create a route rule of the target type **Internet Gateway** that points to it, which allows traffic to the public internet.
11. Create a route table (for example, HubDRG_RT) that you will attach to the DRG in a later step. The specified route allows the flow of traffic from Azure to the spoke VCN through the local peering gateway that you created. For instructions, see To create a route table.
12. Create another route table (for example, HubLPG_RT) that you will assign to the local peering gateway in a later step. This rule allows the flow of traffic from the local peering gateway to the Azure CIDR.

Figure 10: Create a Route Table for the Local Peering Gateway

13. Modify the default security list for the hub VCN to allow traffic to flow between Azure and the spoke VCN through the hub VCN. For this example, traffic is allowed on all protocols, but you can apply more restrictive port and traffic-specific rules. For instructions, see To update rules in an existing security list.

Figure 11: Create Security List Ingress Rules for the Hub VCN
14. Associate the route table that you created in step 11 (for example, HubDRG_RT) with the DRG. For instructions, see To associate a route table with an existing DRG.

![Figure 12: Associate a Route Table with the DRG](image)

15. On the details page for the hub VCN, verify that the route table is attached successfully to the DRG.

![Figure 13: Verify the Route Table Attachment to the DRG](image)
16. Associate the route table that you created in step 12 (for example, HubLPG_RT) with the local peering gateway of the hub VCN. For instructions, see To associate a route table with an existing local peering gateway.

**Figure 14: Associate a Route Table with the Local Peering Gateway**

17. On the details page for the hub VCN, verify that the route table is attached successfully to the local peering gateway.

**Figure 15: Verify the Route Table Attachment to the Local Peering Gateway**
18. Create a subnet (for example, HubPublic) for the hub VCN by using the default route table and default security list that you modified earlier. For instructions, see To create a subnet.

19. Optionally, create a bastion server, if required, in the hub VCN.

Step 2: Set Up the Spoke VCN

As a part of the spoke VCN setup, you create distinct private and public subnets.

- The private subnet hosts a shared Autonomous Database that is accessed from Azure through the hub VCN in Oracle Cloud Infrastructure over the private interconnect.

- The public subnet, optionally, can be used to host any other internet-facing applications. In this example, it is configured but not used.

For detailed information about how to create the Oracle Cloud Infrastructure resources, see the instructions referenced in the steps in the preceding section.
1. Create a spoke VCN (for example, SpokeVCN). Specify a CIDR address block that doesn’t overlap with the hub VCN or the Azure VNet.

![Figure 17: Create the Spoke VCN](image)

2. Create a local peering gateway (for example, SpokeLPG) for the spoke VCN.

![Figure 18: Create the Local Peering Gateway for the Spoke VCN](image)
3. Connect the hub VCN and the spoke VCN by using their local peering gateways. For instructions, see Setting Up a Local Peering.

![Figure 19: Create Local Peering from the Spoke VCN to the Hub VCN](image)

4. On the details page of the spoke VCN, verify that the peering is successful and connected.

![Figure 20: Verify Local Peering of Spoke and Hub VCNs](image)
5. Create a route table (for example, PrivateRT) that will be attached to create a private subnet in the spoke VCN. Specify rules that allow the flow of traffic to the hub VCN and to Azure through a local peering gateway.

![PrivateRT](image1)

**Figure 21: Create Route Table for Private Subnet**

6. If resources in the public subnet also require access from Azure, the hub VCN, or both, add the same rules to the default route table for the spoke VCN.

![Default Route Table for SpokeVCN](image2)

**Figure 22: Create Route Rules for the Spoke VCN Default Route Table**

7. Create a security list (for example, Private SL) for the private subnet, and specify ingress traffic rules from Azure, the hub VCN, and the public subnet of the spoke VCN. More specific rules can be specified to limit the flow of traffic on specified ports.
The rules specified in the security list of a private subnet don’t govern the access to the Autonomous Database. Access to the database is controlled and managed by a network security group, which you will create later.

8. Add an egress traffic rule to allow all the outgoing traffic. By default, user-created (non-default) security lists don’t have any default egress rules. More restrictive rules can be specified as well.
9. Modify the default security list for the spoke VCN and add ingress rules to allow the incoming traffic from the hub VCN and Azure VNet. The default egress rules are sufficient for this example.

![Security List](image)

**Default Security List for SpokeVCN**

<table>
<thead>
<tr>
<th>Ingress Rules</th>
<th>Egress Rules</th>
</tr>
</thead>
<tbody>
<tr>
<td>Source</td>
<td>Destination</td>
</tr>
<tr>
<td>IP Protocol</td>
<td>Port Range</td>
</tr>
<tr>
<td>Protocol</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>--------------</td>
</tr>
<tr>
<td>0.0.0.0/0</td>
<td>TCP All 22</td>
</tr>
<tr>
<td>ICMP</td>
<td>Destination Unreach 4</td>
</tr>
<tr>
<td>10.10.1.0/24</td>
<td>All Protocol All traffic for all ports</td>
</tr>
<tr>
<td>172.16.0.0/16</td>
<td>All Protocol All traffic for all ports</td>
</tr>
</tbody>
</table>

**Figure 25: Create Ingress Rules for the Default Security List for the Spoke VCN**
10. Create a public subnet (for example, SpokePublic) that uses the default route table and the default security list.

![Figure 26: Create a Public Subnet for the Spoke VCN](image-url)
11. Create a private subnet (for example, Spoke Private) using the route table and security list that you created (for example, PrivateRT and PrivateSL).

Figure 27: Create a Private Subnet for the Spoke VCN
12. Create a network security group (NSG) (for example, ADB_NSG) that is specifically used by the Autonomous Database and controls access to the database on the specified port. For instructions, see To create an NSG.

13. Create a shared Autonomous Database in the private subnet and use the network security group that you created to control the access to the database. For instructions, see Creating an Autonomous Database on Shared Exadata Infrastructure.

Note the private endpoint IP address, which is used to access the database from Azure.
Step 3: Set Up the Azure Environment

1. In the Azure Portal, create a virtual network (VNet).
   
   A. On the Basics page, enter a name for the VNet (for example, AzureVNET) and select the same region that you’re using for Oracle Cloud Infrastructure (for example, Japan East).

   ![Create virtual network](image1)

   **Figure 30: Create VNet in Azure, Specify Region**

   B. On the IP Addresses page, specify the CIDR address block and create subnets, as required. In this example, the subnet is called PublicSubnet.

   ![Create virtual network](image2)

   **Figure 31: Create VNet in Azure, Specify CIDR Address Block**
C. On the **Security** page, modify the security settings for the VNet, as required.

![Create virtual network](image)

Figure 32: Create VNet in Azure, Specify Security Settings

D. On the **Tags** page, specify any optional tags, as required.

![Create virtual network](image)

Figure 33: Create VNet in Azure, Specify Tags

E. On the **Review + create** page, review the information, and then click **Create**.

![Create virtual network](image)

Figure 34: Create VNet in Azure, Review Information
2. Create a VNet gateway (for example, AzureVNET_Gateway), and select ExpressRoute as the gateway type. The gateway requires a dedicated subnet (the minimum subnet requirement is /28) and a public IP address.

![Create virtual network gateway](image1)

Figure 35: Create VNet Gateway

3. Create a route table (for example, Public_Tokyo_RT) for the public subnet that you created earlier. You add specific routes in a later step.

![Create route table](image2)

Figure 36: Create Route Table
4. Create a network security group (NSG) (for example, Public_Tokyo_NS) to control the incoming and outgoing traffic. You add individual rules in a later step.

![Create network security group](image)

**Figure 37: Create Network Security Group**

5. Associate the route table and the network security group that you created with the subnet, which will access the Autonomous Database that you created in Oracle Cloud Infrastructure (in the preceding section).

![PublicSubnet](image)

**Figure 38: Associate Route Table and Network Security Group with Subnet**
6. Modify the route table, and specify the CIDR address block of the hub VCN in Oracle Cloud Infrastructure.

![Figure 39: Modify Route Table with CIDR Address Block of the Hub VCN](image)

7. Modify the network security group, and specify the incoming and outgoing traffic rules to the hub VCN in Oracle Cloud Infrastructure.

![Figure 40: Modify Network Security Group with Traffic Rules to the Hub VCN](image)
Step 4: Connect Oracle Cloud Infrastructure and Azure Using the Private Interconnect

1. In Azure Portal, create an ExpressRoute connection.
   
   A. On the Basics page, enter a name (for example, Azure_OCI_Circuit) and select the appropriate region and resource group (that you specified in the preceding section).

   ![Figure 41: Create an ExpressRoute Connection in Azure](image)

   B. On the Configuration page, select Oracle Cloud FastConnect as the provider. The peering location shows only the regions that are currently supported for the private interconnect with Oracle Cloud Infrastructure. Select the SKU option carefully; the price can vary drastically among the options. The Local SKU option has no separate ingress or egress charges.

   ![Figure 42: Configure the ExpressRoute Circuit](image)
C. Verify the options that you selected, and click **Create**.

![Create ExpressRoute](image)

*Figure 43: Verify the ExpressRoute Connection and Create*

2. After the ExpressRoute circuit is created, copy the service key that is shown on the details page. You use this key in the next step to provision an Oracle Cloud Infrastructure FastConnect connection and establish dynamic routing between Azure and Oracle Cloud Infrastructure.

![ExpressRoute Details Page Showing the Service Key](image)

*Figure 44: ExpressRoute Details Page Showing the Service Key*
3. In the Oracle Cloud Infrastructure Console, create a FastConnect connection.
   
   A. On the Connection Type page, select **Microsoft Azure: ExpressRoute** from the list of FastConnect partners. For instructions, see *FastConnect: With an Oracle Partner*

   ![Figure 45: Create FastConnect Connection to Azure, Specify Connection Type](image)

   B. On the **Configuration** page, specify the dynamic routing gateway (DRG) that you created when you set up the hub VCN. Also, specify the service key that was provided when you set up the ExpressRoute circuit in Azure.

   Provide the nonoverlapping BGP IP addresses to use for the two redundant BGP sessions between Oracle and Azure. For each pair, you must provide a separate /30 block of addresses (each /30 has four IP addresses).
C. Click Create.

After a few minutes, the FastConnect connection is provisioned and dynamic routing is established between Azure and Oracle Cloud Infrastructure.
4. In the Azure Portal, verify that the provider status of the ExpressRoute circuit has changed to **Provisioned**.

![Image of Azure Portal with ExpressRoute circuit settings](image)

Figure 48: Verify Status of Provider on ExpressRoute Details Page

5. In the navigation pane, under **Settings**, click **Connections**.

6. Create a connection to attach the ExpressRoute circuit with the VNet gateway.

   A. On the **Basics** page, enter a name for the connection (for example, Azure_OCI_Connection).

   ![Image of Create Connection page](image)

   Figure 49: Create Connection to Attach ExpressRoute to VNet Gateway
B. On the **Settings** page, select the virtual network gateway (for example, AzureVNET_Gateway) from the list.

![Create connection](image1)

**Figure 50: Create Connection to Attach ExpressRoute to VNet Gateway, Select VNet Gateway**

C. On the **Review + create** page, confirm the options, and click **Create**.

![Create connection](image2)

**Figure 51: Create Connection to Attach ExpressRoute to VNet Gateway, Verify Options**
7. Verify that the connection is successfully added to the ExpressRoute circuit.

![Azure_OCI_Circuit](image)

**Verify Connection Is Added**

The effective routes should now be visible in the route table that is associated with the subnet (for example, Public_Tokyo_RT). The routes to the hub and spoke VCNs in Oracle Cloud Infrastructure should be visible through the virtual network gateway.

![Public_Tokyo_RT](image)

**Effective Routes Between Azure and Oracle Cloud**
Step 5: Access Autonomous Database from Azure Through the Private Interconnect

1. Create a virtual machine in the Azure public subnet and install Oracle Client to be able to access an Oracle database in Oracle Cloud Infrastructure.

```
[oracle@AzAppServer ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
       inet 172.169.1.1 netmask 255.255.255.0 broadcast 172.169.1.255
       inet6 fe80::20d:8aff:fe40:49d2 prefixlen 64 scopeid 0x20<link>
       ether 00:0d:3a:cd:4e:92 txqueuelen 1000 (Ethernet)
       RX packets 673307 bytes 1270932208 (1.1 GiB)
       RX errors 0 dropped 0 overruns 0 frame 0
       TX packets 353034 bytes 36754225 (33.1 MiB)
       IX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    loop txqueuelen 1000 (Local Loopback)
    RX packets 10437 bytes 13305824 (12.6 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 10437 bytes 13305824 (12.6 MiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[oracle@AzAppServer ~]$ 
```

*Figure 54: IP Address of an Azure VM*

2. In the Oracle Cloud Infrastructure Console, go to the details page of the Autonomous Database, click **DB Connection**, and download the instance wallet. The wallet contains the connection string and the wallet files for database traffic encryption between the database and the clients.

*Figure 55: Select the Oracle Wallet Type for the Autonomous Database*
3. Specify the password for the wallet and download it to your machine.

![Download Wallet]

Database connections to your Autonomous Database use a secure connection. The wallet file will be required to configure your database clients and tools to access Autonomous Database.

Please create a password for this wallet. Some database clients will require that you provide both the wallet and password to connect to your database (other clients will auto-login using the wallet without a password).

Password

Password: 

Confirm password

Password: 

Download

Figure 56: Download Oracle Wallet for the Autonomous Database

4. Transfer the wallet file to the Azure client server and unzip the contents in the $ORACLE_HOME/network/admin folder, overwriting any existing content that might exist.

5. On the database client machine, locate the tnsnames.ora file in $ORACLE_HOME/network/admin and replace the hostname with the private endpoint IP address of the Autonomous Database.

6. Perform a tnsping ping test to ensure that the listener of the Autonomous Database is reachable.

![TNSPing]

Figure 57: Result of a Ping Test

The connection to the database is now successfully established.

![SQLPlus]

Figure 58: Successful Database Connection
CONCLUSION

The implementation of transit routing (hub-and-spoke configuration) offered in Oracle Cloud Infrastructure can be extended to applications that span between Microsoft Azure and Oracle Cloud Infrastructure. A Database service deployed in a spoke VCN in Oracle Cloud Infrastructure can be securely accessed over the private interconnect with Azure without the use of service gateway. Traffic originating from Azure can be centrally controlled and directed from a hub VCN in Oracle Cloud Infrastructure to the appropriate spoke VCN.

REFERENCES

- Transit Routing: Access to Multiple VCNs in the Same Region
- Set up a hub-and-spoke network topology
- Set up a direct interconnection between Azure and Oracle Cloud Infrastructure
- Create and modify an ExpressRoute circuit using PowerShell
- Access to Microsoft Azure
Deploy the Oracle Cloud and Microsoft Azure Interconnect Using Hub-and-Spoke Topology
September 2020