Preity Gupta

HCL

Best Practices for Hybrid Application in cloud

Table of Contents

[The Original On-Premises Application Architecture 2](#_Toc498705042)

[Hybrid Application Architecture after Conversion 3](#_Toc498705043)

[Scenarios Covered 4](#_Toc498705044)

[Scenario First 4](#_Toc498705045)

[Scenario Second 4](#_Toc498705046)

[Scenario Third 4](#_Toc498705047)

[Deployment of Data 4](#_Toc498705048)

[Deploy All the Data in the Cloud 4](#_Toc498705049)

[Keep All Data On-premises 4](#_Toc498705050)

[Deploy Some of the Data in the Cloud 4](#_Toc498705051)

[Best Practices for deployment of data 5](#_Toc498705052)

[Choosing Data Storage in Cloud 5](#_Toc498705053)

[Available Data Storage 5](#_Toc498705054)

[Best Practices for Choosing Storage 6](#_Toc498705055)

[Synchronizing Data across Cloud and On-Premises Locations 6](#_Toc498705056)

[Available Data Sync Options 6](#_Toc498705057)

[Best Practices for Data Sync 6](#_Toc498705058)

[Choosing a Reporting Solution 7](#_Toc498705059)

[Available Reporting Solutions 7](#_Toc498705060)

[Best Practices for Reporting Services 8](#_Toc498705061)

[Choosing a Communications Mechanism 8](#_Toc498705062)

[Available Communication Mechanism 8](#_Toc498705063)

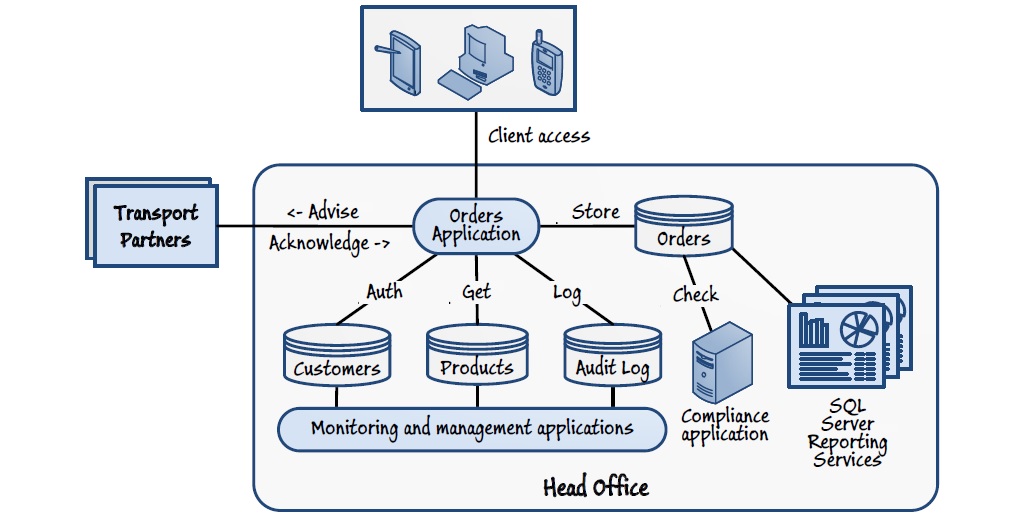
[Best Practices for Communication Mechanism 9](#_Toc498705064)

[Summary 9](#_Toc498705065)

[Appendixes 10](#_Toc498705066)

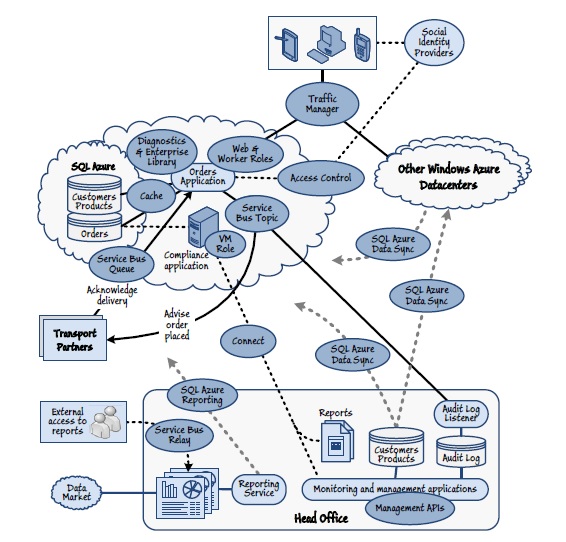
# The Original On-Premises Application Architecture

The Original On-Premises application has 3 Applications: The Orders application (is a website and contains associated business logic), the monitoring and management applications, Compliance Application, 4 Databases: Orders, Customers, Products and Audit Log and SQL Server Reporting Services.



The Orders application stores order details in Orders database, accessed by authenticated customers from Customers databases, get products details from Product databases and saved logs in Audit Log databases. The Orders application sends a message (date, weight and packaging number) to the appropriate transport partner when a customer places an order. There are 2 transport partners, one for local deliveries and another for deliveries outside of the area. The transport partner sends a message back to the Orders application after the delivery is completed.

# Hybrid Application Architecture after Conversion



* Windows Azure Traffic Manager, redirects customer request to the instance of the Orders application running in the closest datacentre.
* Windows Azure Access Control Service (ACS) manages authentication, and returns a token containing a unique user ID to the Orders application. The Orders application uses this token to look up the customer details in the Customers table of the database running in a local SQL Azure datacentre.
* Account details of Customer are synchronized between the Customers table held in the on-premises database and SQL Azure in all the datacentres.
* The Orders application displays a list of products stored in the Products table. The Products data is kept up to date by synchronizing it from the master database located On-Premises.
* The Orders application stores the order details in the Orders table of the database in the local SQL Azure datacentre. All orders are synchronized across all Windows Azure datacentres
* The Orders application sends an order message to the appropriate transport partner.
* The Orders application sends any required audit information, such as orders over a specific total value to the On-Premises management and monitoring application, which will store this information in the Audit Log table of the database located On-Premises.
* The compliance application running in the cloud validates the orders in the Orders table for conformance with legal restrictions and sets a flag in the database table on those that require attention by managers.
* Transport partners send a message to the Orders application on delivery of order.
* The On-Premises Reporting application uses the Business Intelligence features of the SQL Azure Reporting service running in the cloud to generate reports from the Orders table.

## Scenarios Covered

### Scenario First

UI layer i.e. Order Application is on Cloud. Business layer i.e. Monitoring and Management Application On-Premises & Database like Customer Database and Audit Log is On-Premise.

### Scenario Second

Application i.e. Order Application on Cloud and Integration/Interfaces like Service Bus Queues/topics for Interaction with Transport Partners.

### Scenario Third

UI layer i.e. Order Application is on Cloud. Business layer of Compliance Application is on Cloud, Database Like Customer Database and Audit Log is On-Premise

# Deployment of Data

## Deploy All the Data in the Cloud

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| All components of application are in close to each other, hence maximise performance and minimise response times | Failed connections due to occasional Internet networking and performance issues, and  additional costs would be incurred for access to the data from the on-premises applications. |
| Removes the requirement to synchronize data between cloud and on-premises locations | Requirement to synchronize the data between these deployments if the application is in more than one datacentre |
| Helps take advantage of the scalability | High storage costs for deploying large volumes of data or multiple databases |

## Keep All Data On-premises

|  |  |
| --- | --- |
| **Advantages** | **Disadvantages** |
| Data can be secured and managed easily, especially if most of the update operations are done by on-premises staff. | Security and reliability hampers when remote applications and services must perform updates across the Internet in multiple databases. |
| Ensures that it comply with legal or regulatory limitations on the location and security of sensitive information |  |
| No requirement to migrate or deploy data to a  remote location |  |

## Deploy Some of the Data in the Cloud

|  |  |
| --- | --- |
| Advantages | Disadvantages |
| Data for applications and services that require fast and reliable access can be in the cloud, close to the application or service that uses it, whereas data that is mostly accessed by head office applications can remain on-premises to provide fast and reliable access for these applications. | Suitable secure and reliable connectivity mechanism will be required |
| Data that does not need to scale can remain on-premises, saving hosting costs, whereas data  that must scale can be in Windows Azure to take advantage of the scalability these services offer. | Data replication and synchronization solution will be necessary to ensure that data in all locations is consistent |
| Data that is subject to legal or regulatory limitations regarding its storage location, or requires specific security mechanisms to be in place, can remain on-premises. |  |

## Best Practices for deployment of data

* Customer data is likely to be relatively static and not change much over time therefore its master database is managed On-Premises to maximise security. Registration of customer is done at On-Premises and only minor changes is allowed using application at Cloud. As Orders application (on Cloud) allows only authenticated customer to place orders thus replica of Customer data is maintained in cloud for better performance and reliability. Bidirectional Synchronisation helps updates in each data centres where ever Order application is hosted and replicating back the minor changes done by customers to their personal information.
* Product data is relatively static so keep the master Product data On-Premises. To maximize performance and reliability, replicate of some fields of the Product data (data required to list products, show product details, and accept orders) in the cloud, close to the Orders application. Unidirectional synchronization replicates to all datacentres that host the Orders application.
* Order data is highly dynamic so it will be kept at cloud, close to Order Application. When the Orders application is deployed to more than one datacentre, bi-directional synchronization of the order data between datacentres ensures that customers see their order information. The only issue is that SQL Server Reporting Services to create business intelligence reports cannot be done on the data directly.
* Audit Log database is a complete repository for all application management and monitoring facilities. It must be kept On-Premises to ensure security of sensitive data like exceptions handling, Orders above 1 lakh, government regulations on sale.

# Choosing Data Storage in Cloud

## Available Data Storage

|  |  |
| --- | --- |
| **Azure Storage** | |
| Blob Storage | Blob storage is ideal for storing unstructured information such as images, files, and other resources. |
| Table Storage | Table storage is best suited for structured information. Table storage is very flexible and supports geographical replication, so that access is fast. It is significantly cheaper than using a SQL Azure database. But it does not support reading and writing of data, database transactions and data cannot be directly imported from SQL Server to Table storage. That requires redesigning of data model which incurs cost and time during migration. |
| Queues | Queues are used for passing information between roles and services, and are not designed for use as a persistent storage mechanism. |
| SQL Azure | SQL Azure is a high-performance database service that fully supports SQL-based operations. It is also compatible with existing connectivity APIs, such as the Entity Framework (EF), ADO.NET, and Open Database Connectivity (ODBC). It can be easily managed through the Windows Azure Management Portal. In addition, data synchronization across cloud-hosted and on-premises databases is easily achieved  through Windows Azure Data Sync service or the Data Sync APIs. SQL Azure supports business intelligence reporting with the SQL Azure Reporting Service. |

## Best Practices for Choosing Storage

As the data formats and types, and the data access code, are all designed to work with SQL Server so best option is to choose SQL Azure as the data storage. This way, Schema redesign and code development costs is saved as compared to table storage. Database transactions and complex queries can easily be managed.

# Synchronizing Data across Cloud and On-Premises Locations

## Available Data Sync Options

|  |  |
| --- | --- |
| SQL Azure Data Sync | SQL Azure Data Sync offers a variety of options for unidirectional and bi-directional synchronization. It can synchronize data between on-premises SQL Server databases and one or more SQL Azure databases hosted in the cloud. It can be managed through the Windows Azure web portal. This helps to reduce the cost and time required to implement a solution. But SQL Azure Data Sync works with only SQL Server and SQL Azure databases. |
| Microsoft Sync Framework | SQL Azure Data Sync uses the components of the Microsoft Sync Framework to perform data synchronization. The Sync Framework is a comprehensive synchronization platform that supports any data type, any data store, any transfer protocol, and any network topology. It is not confined to use with just SQL Server and SQL Azure databases. As the developers have to write additional code to control the synchronization process, it incurs additional cost and time compared to using the SQL Azure Data Sync service. |

## Best Practices for Data Sync

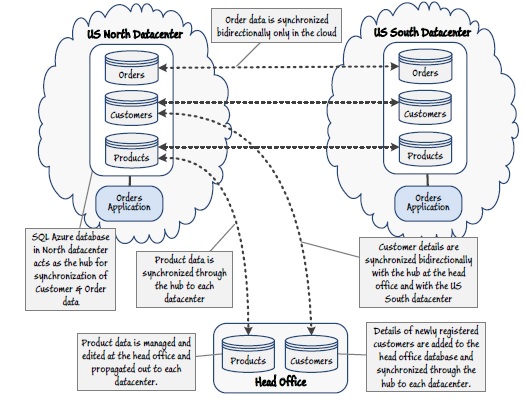
All the data should be stored in either SQL Server On-premises or SQL Azure in the cloud, so SQL Azure Data Sync will be able to access and synchronize all the data as required. The different types of information which can be synchronized are:

• Order data is maintained in the cloud using SQL Azure, and is synchronized between datacentres. This information is not propagated back to the On-Premises database.

• Product information is maintained exclusively On-Premises by using SQL Server, but the details required for placing orders are copied/replicated to each SQL Azure database at each datacentre on a periodic basis.

• New customers are registered On-Premises and their details are added to On-Premises SQL Server database. These details are replicated out to SQL Azure at each datacentre, enabling a customer to log in and access the Orders application.

Following diagram explains the Data Synchronization process:



# Choosing a Reporting Solution

## Available Reporting Solutions

|  |  |
| --- | --- |
| SQL Server Reporting Services | SQL Server Reporting Services creates attractive and comprehensive business intelligence reports from data stored in the database tables, or from a variety of other data sources like relational, multidimensional, XML, and custom data sources. It also supports publishing reports directly to a variety of targets, including a report server, SharePoint services, file shares, internal archive stores, and Office applications. |
| SQL Azure Reporting Service | The SQL Azure Reporting Service generates a range of customized business intelligence reports in common web and Microsoft Office-ready formats. The SQL Azure Reporting Service runs in the same datacentre as the SQL Azure database so network traffic between SQL Azure Reporting Service and the application is minimized and performance is enhanced. It is a chargeable service and so incurs subscription costs. |
| A Custom or Third Party Reporting Solution | Custom or third party reporting solutions may be closely tailored to the organization’s reporting requirements, thus focuses on specific areas of interest. Hence provides faster report generation and enhanced performance compared to more generic solutions. Using an existing reporting solution can reduce the cost of migrating an application to the cloud. To connect to the cloud-based data source, security of the data can be hampered. An existing or third party solution may not offer the required variety of formats, or equivalent functionality, compared to SQL Server Reporting Services or the SQL Azure Reporting Service. |

## Best Practices for Reporting Services

One solution is when the source data is located remotely i.e. on Cloud then download all the orders data to On-Premises database, and continue to use SQL Server Reporting Services to analyse it. However, unless data synchronization occurs on a scheduled basis, which will incur additional cost, the data transfer operation will result in longer waiting times while reports are generated.

Second Solution is as data is in azure SQL Azure, it made more sense to adopt the business intelligence capabilities of the SQL Azure Reporting Service. This approach minimizes network traffic over the Internet, ensures that the most recent data is included in the reports without incurring additional delays, and can still generate the reporting information in a variety of formats.

# Choosing a Communications Mechanism

## Available Communication Mechanism

|  |  |
| --- | --- |
| Electronic Data Interchange (EDI) | The worker role could connect to the transport partner over an interface, protocol, and format that the worker role and the transport partner both understand, such as EDIFACT, RosettaNet, cXML, and BASDA. These are commonly accepted and well understood standards. But worker role requires additional software and infrastructure to connect to an EDI interface. Microsoft BizTalk Server provides adapters for many well-known protocols and formats, but this solution requires passing all orders through BizTalk Server running On-Premises. |
| Web Services (Push Model) | Web services are a common, well-understood, and maturing technology. From a security perspective, requests can be easily encrypted. Issues with Push approach is, if the transport partner does not provide a web service interface, then this approach cannot be used. Another issue is that the web service may not provide an appropriate level of reliability i.e. the worker role may be unaware if the transport partner’s system raises an error that causes the details of the order to be lost. Connectivity is also an issue: if the web service at the transport partner is temporarily unavailable then a connection cannot be established. |
| Web Services (Pull Services) | Web services can provide a secure, scalable, and reliable communication mechanism. Pull approach enables transport partners full control of functionality and provides protection, security, scalability and reliability. New transport partners can be also easily integrated without modifying the worker role. Furthermore, the worker role can take advantage of Service Bus Relay for authentication through the Windows Azure Access Control Service (ACS), thus enables transparently route messages to the web service endpoints published by the worker role. But as the volume of orders increases, transport partners may not query the web service sufficiently, causing a backlog of orders to build up. |
| Windows Azure Storage Queue | This mechanism is relatively simple, reliable, scalable, and secure. Windows Azure storage queues are managed and maintained within a datacentre. Transport partners need access rights to connect to the queue to retrieve and post messages. |
| Windows Azure Service Bus Queues | This approach is highly scalable as the worker role can post messages to the Service Bus queue as quickly as orders are placed. It also offers improved reliability as after the worker role has posted a message to a Service Bus queue, it will not be lost. It will either remain on the queue until it expires or delivered to transport partner. Security is highly configurable as It is managed through the Windows Azure Access Control Service (ACS). |
| Service Bus topic and Subscription | Service Bus queues provide an attractive and scalable as no need to create and manage a separate queue for each partner. Therefore, the final option is to post the details of orders as messages to a Windows Azure Service Bus topic. Transport partners can subscribe to topic to retrieve the orders to be shipped. Messages acknowledging receipt of the order details and messages indicating that delivery was completed are posted back to the worker role through a Service Bus queue. Like Service Bus queues, Service Bus topics and subscriptions are highly scalable and reliable, with configurable security. Each transport partner can connect to the Service Bus topic through its own subscription, which can filter the messages based on this metadata so that each transport partner receives only the orders that it should process. Topics also enable messages to be routed to multiple destinations, so orders with a value over 1 lakh can be directed to the Audit Log Listener. The only drawback to, is that each transport partner must be prepared to connect to the appropriate Service Bus topic to retrieve messages. Another limitation is a topic can currently have a maximum of 2000 subscriptions and can support up to 100 concurrent connections. |

## Best Practices for Communication Mechanism

Send orders from the worker role to the transport partners by using a Service Bus topic. Each transport partner receives messages by using a subscription that filters the orders. In this way, each transport partner receives only the orders that it should ship. To bridge the potential technology gap between the systems implemented by the transport partners and the Service Bus, construct a set of connectivity components to translate messages retrieved from the Service Bus and convert them into format expected by the transport partner. Provides credentials necessary to enable the connector to listen to the appropriate Service Bus subscription. The transport partner’s own system uses this connector to retrieves the details of orders from the subscription. Additionally, the connector exposes an interface that the transport partner’s system uses to post acknowledgment messages back to the Service Bus queue that the Orders application listens on.

# Summary

When planning to move parts of an existing application from On-Premises to the cloud, there are many concerns related to scalability, reliability, security and communication. For example, how will cloud-based applications call On-Premises services, or send messages to On-Premises applications? How will cloud-based applications access data in On-Premises data stores? How can you ensure that all instances of the application running in cloud datacentres have data that is up-to-date? This helps you to identify them more accurately, and discover the solutions to resolve them.

# Appendixes

SQL Azure Data Sync Overview > <http://social.technet.microsoft.com/wiki/contents/articles/sql-azure-data-sync-overview.aspx>

Windows Azure Queues and Windows Azure Service Bus Queues - Comparison >

<http://msdn.microsoft.com/en-us/library/windowsazure/hh767287(v=vs.103).aspx>