



# MPS Partners

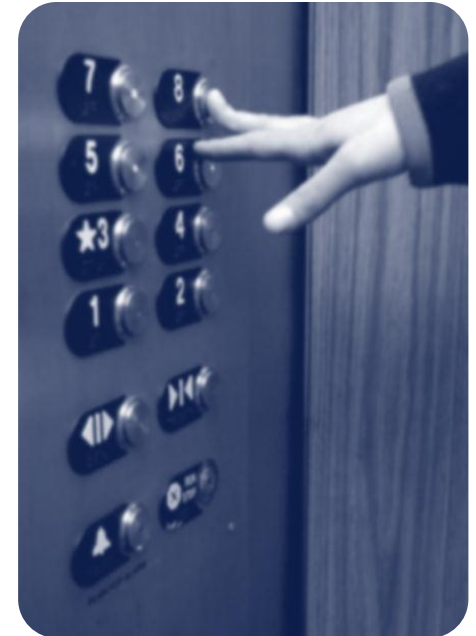
An SPR Company

Azure Learning Series:  
Creating Hybrid-cloud Applications Using the  
Windows Azure Service Bus  
December 15, 2011



[www.mpspartners.com](http://www.mpspartners.com)

- MPS Partners is a Microsoft Gold Certified Managed Partner with deep expertise in defining and deploying solutions based on Microsoft technology
- We focus in a few key areas:
  - ◆ Collaboration and enterprise content management
  - ◆ Business Intelligence
  - ◆ Integration of the Microsoft toolset with diverse technology landscapes and the cloud
  - ◆ We are especially known for having this expertise within accounts that run SAP
  - ◆ We can help develop your strategy around key Microsoft technologies



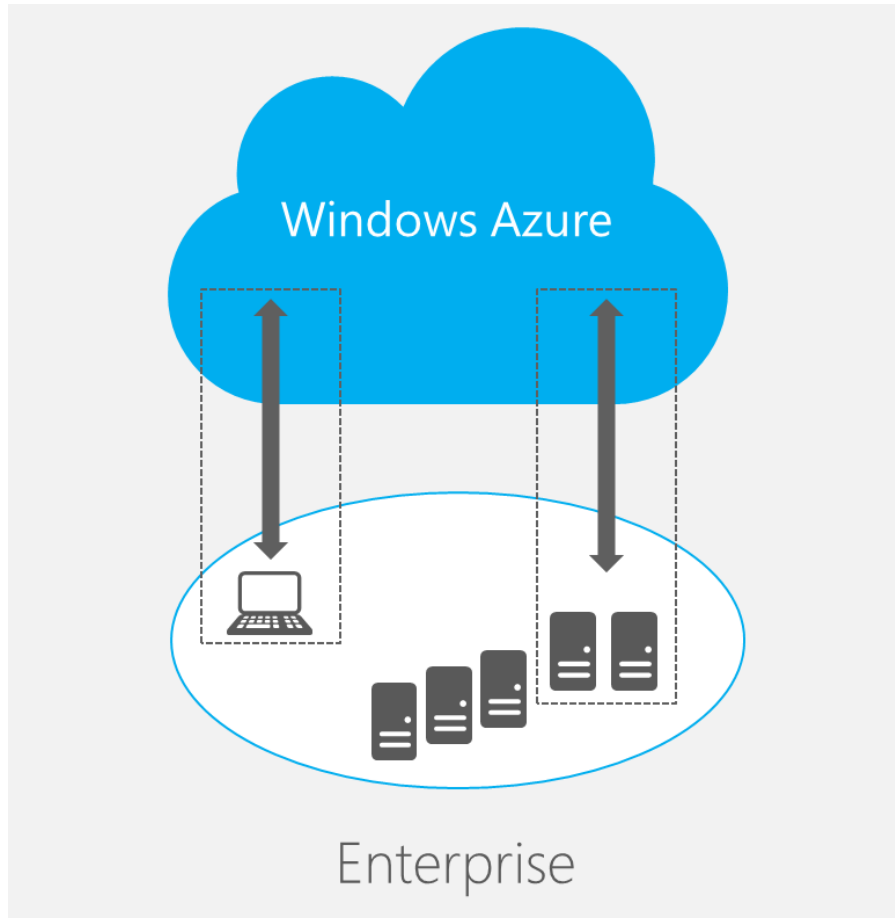
- Enterprises are seeing early benefits of cloud computing:
  - ▶ **Productivity**
    - Quicker time to market, Fast updates
    - Anywhere access
    - Elasticity
  - ▶ **Infrastructure Advantages**
    - Reliability and Fault Tolerance
    - Faster Deployment
    - No patching or maintenance
  - ▶ **Economics**
    - Low cost of entry
    - You pay for what you use



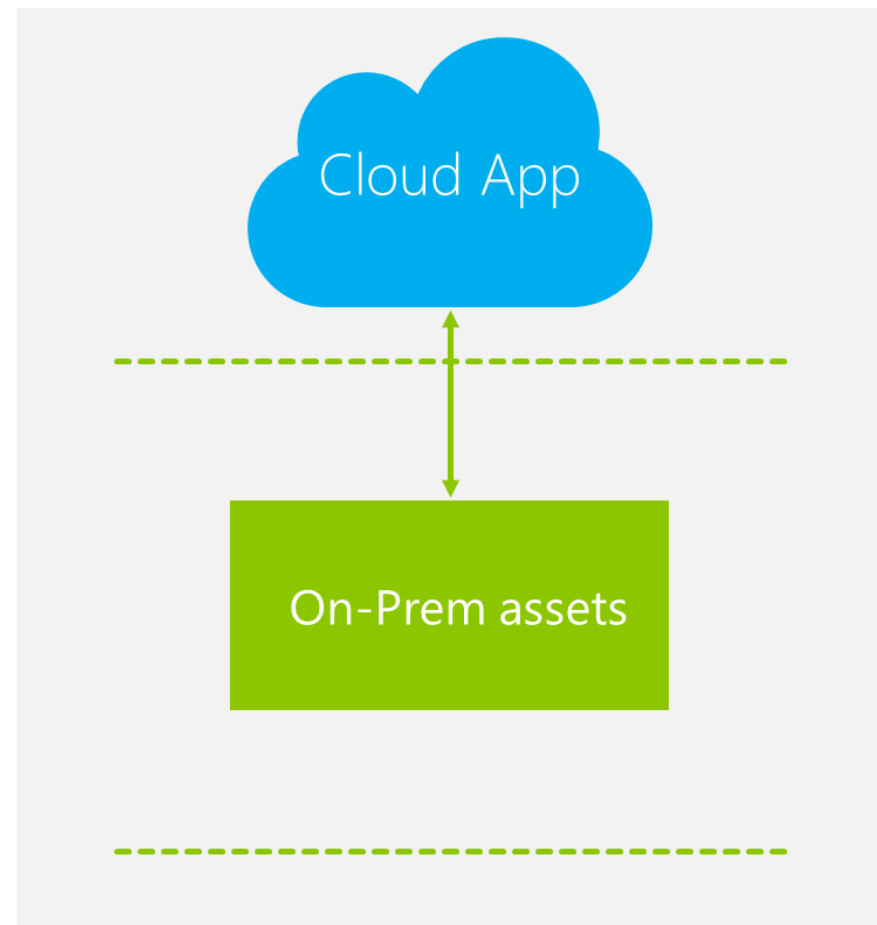
- LoB Systems will be slower to leverage cloud technologies
  - ▶ Cost of entry is much higher due to large install and code base
  - ▶ Security is a huge concern with mission critical data
- Business users require the agility and reach the cloud provides
  - ▶ Have more collaboration with external parties and business partners
  - ▶ Business IT continue to purchase cloud applications that require access to on premise data



A **hybrid cloud** is a **cloud computing infrastructure** composed of two or more clouds (private, community, or public) that remain unique entities but are bound together by standardized or proprietary technology that enables data and application portability.<sup>[1]</sup> By utilizing "hybrid cloud" architecture, companies and individuals are able to obtain degrees of fault tolerance combined with locally immediate usability without dependency on internet connectivity. Arguably Hybrid Cloud architecture is the ideal combination requiring on premises resources and off site (remote) server based cloud infrastructure.



Windows Azure Connect



Service Bus

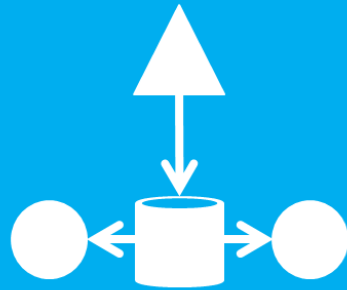
# Windows Azure Service Bus



## Connectivity

Service Relay  
Protocol Tunnel  
Eventing, Push

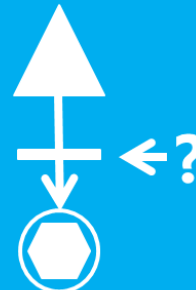
Rich options for interconnecting apps across network boundaries



## Messaging Queuing Pub/Sub

Reliable Transfer

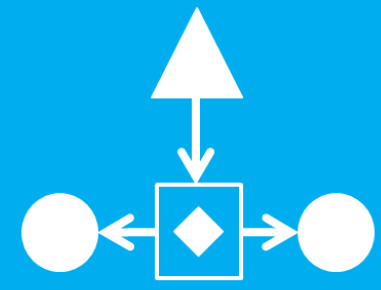
Reliable, transaction-aware cloud messaging infrastructure for business apps



## Service Management

Naming, Discovery  
Monitoring

Consistent management surface and service observation capabilities



## Integration

Routing  
Coordination  
Transformation

Content-based routing, document transformation, and process coordination

# "Expose Web Services from anywhere to anywhere"



## Key Capabilities

Relayed One-Way Unicast and Multicast

Relayed WCF NET.TCP with Direct Connect Option

Relayed WCF HTTP with support for REST and SOAP 1.1/1.2

Endpoint protection with Access Control

## Outbound TCP (Ports 9350-9353)

9350 Unsecured TCP One-way (client)

9351 Secured TCP One-way (all listeners, secured clients)

9352 Secured TCP Rendezvous (all listeners except one-way)

9353 Direct Connect Probing Protocol (TCP listeners with direct connect)

## Outbound HTTP (Port 80, Listeners)

TCP equivalent tunnel with overlaid TLS/SSL formed over pair of HTTP requests

Alternate connectivity path if outbound TCP is blocked

## Outbound HTTPS (Port 443, Senders)



## Connectivity Options

# Relay Programming Model

## Full WCF Programming Model

Bindings functionally symmetric with WCF

WebHttpRelayBinding (HTTP/REST)

BasicHttpRelayBinding (SOAP 1.1)

WS2007HttpRelayBinding (SOAP 1.2)

NetTcpRelayBinding (Binary transport)

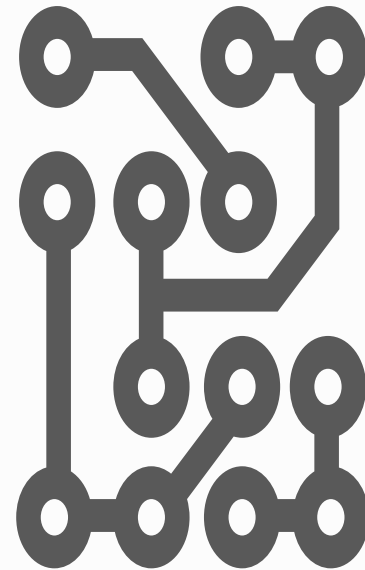
## Special Service Bus Bindings

NetOnewayRelayBinding (Multicast one-way)

NetEventRelayBinding (Multicast one-way)

Transport binding elements for custom binding stacks

WebHttpRelayBinding provides full interoperability with any HTTP/REST client, BasicHttpRelayBinding with any SOAP client



# Oneway

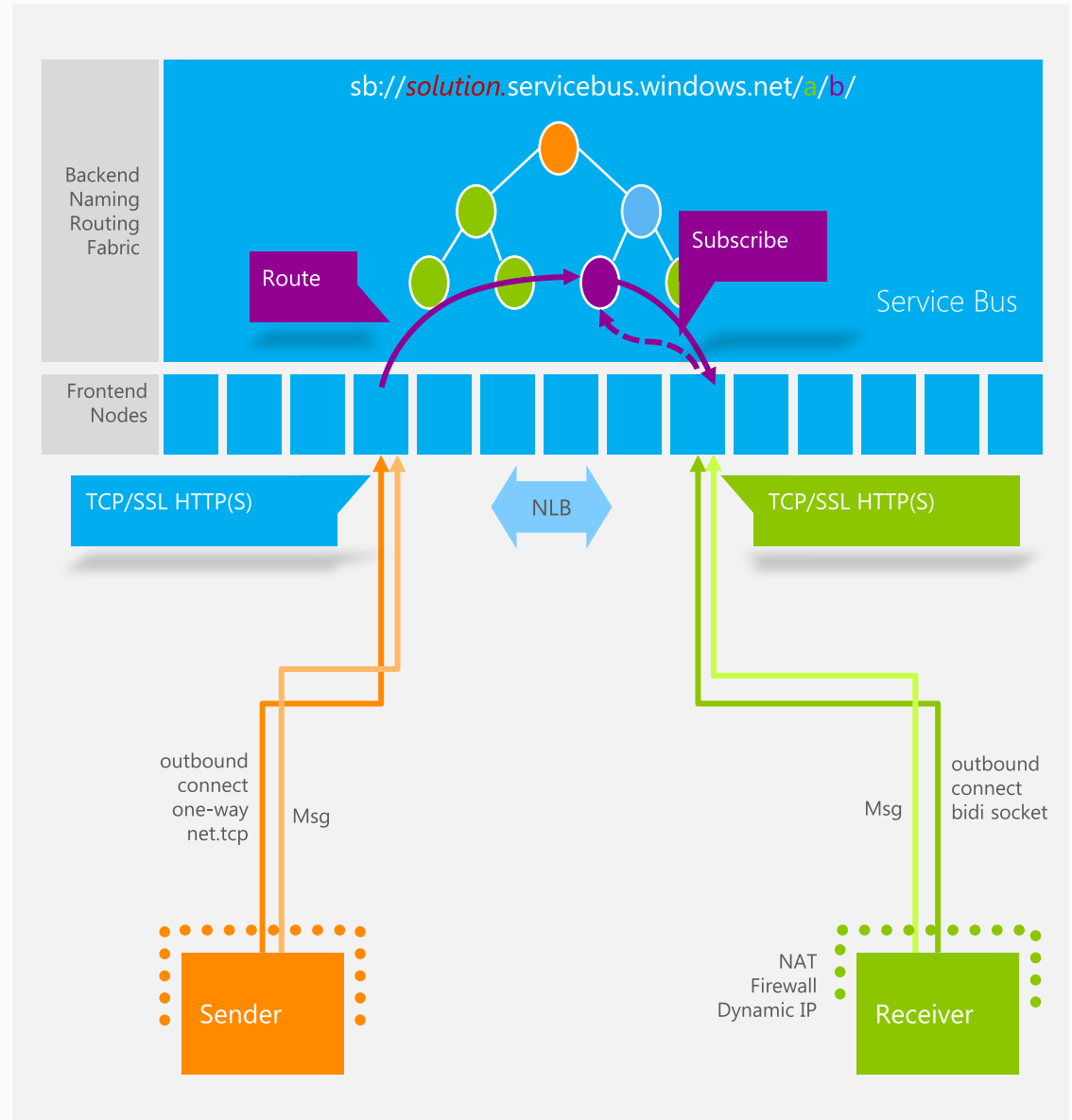
NetOnewayRelayBinding

All TCP and HTTP listeners use one-way as internal control channel

60KB message-size limit

One-way only

No rendezvous overhead



# Event

NetEventRelayBinding

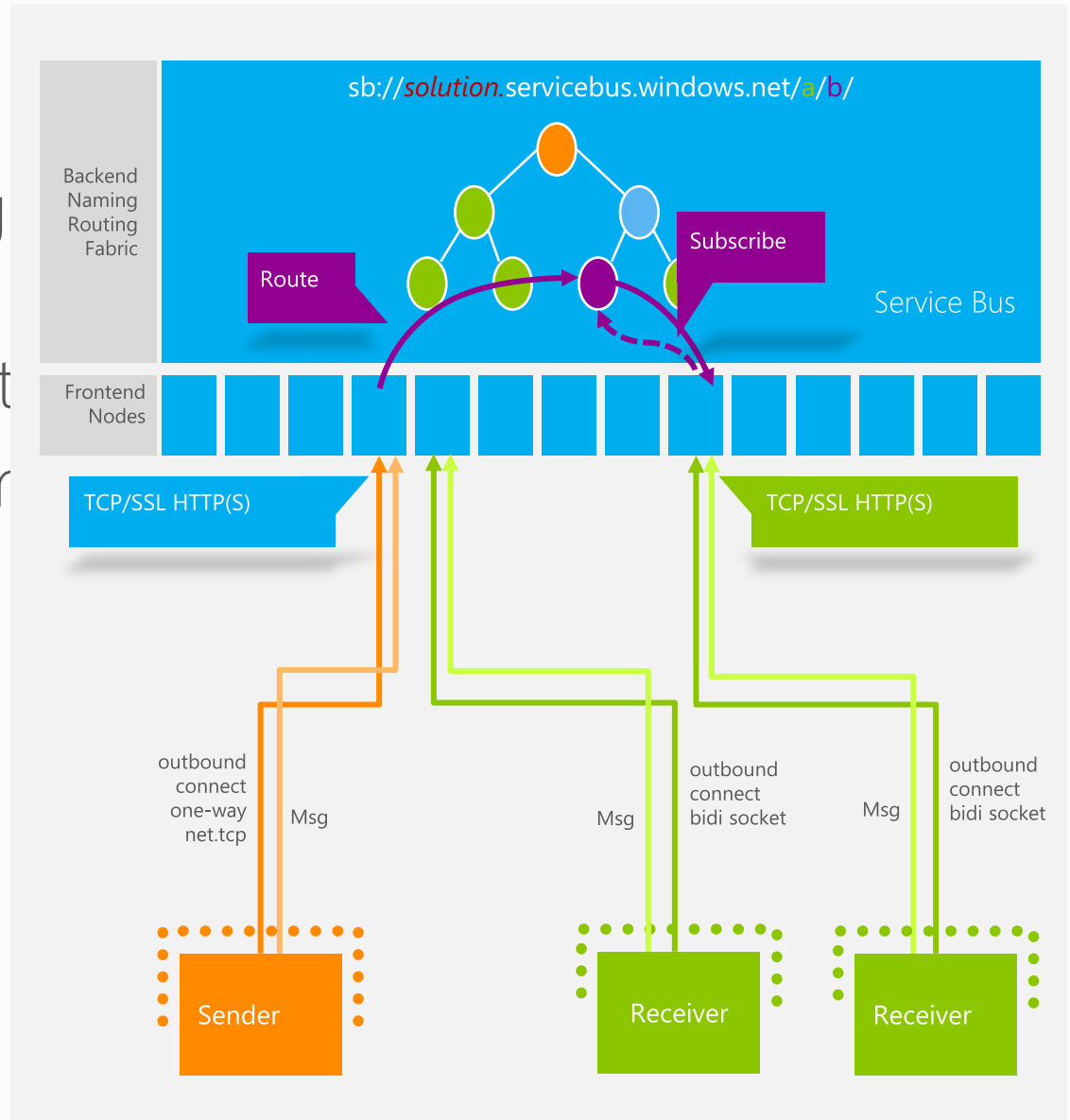
Small-Scale

Synchronous Multicast

60KB message-size limit

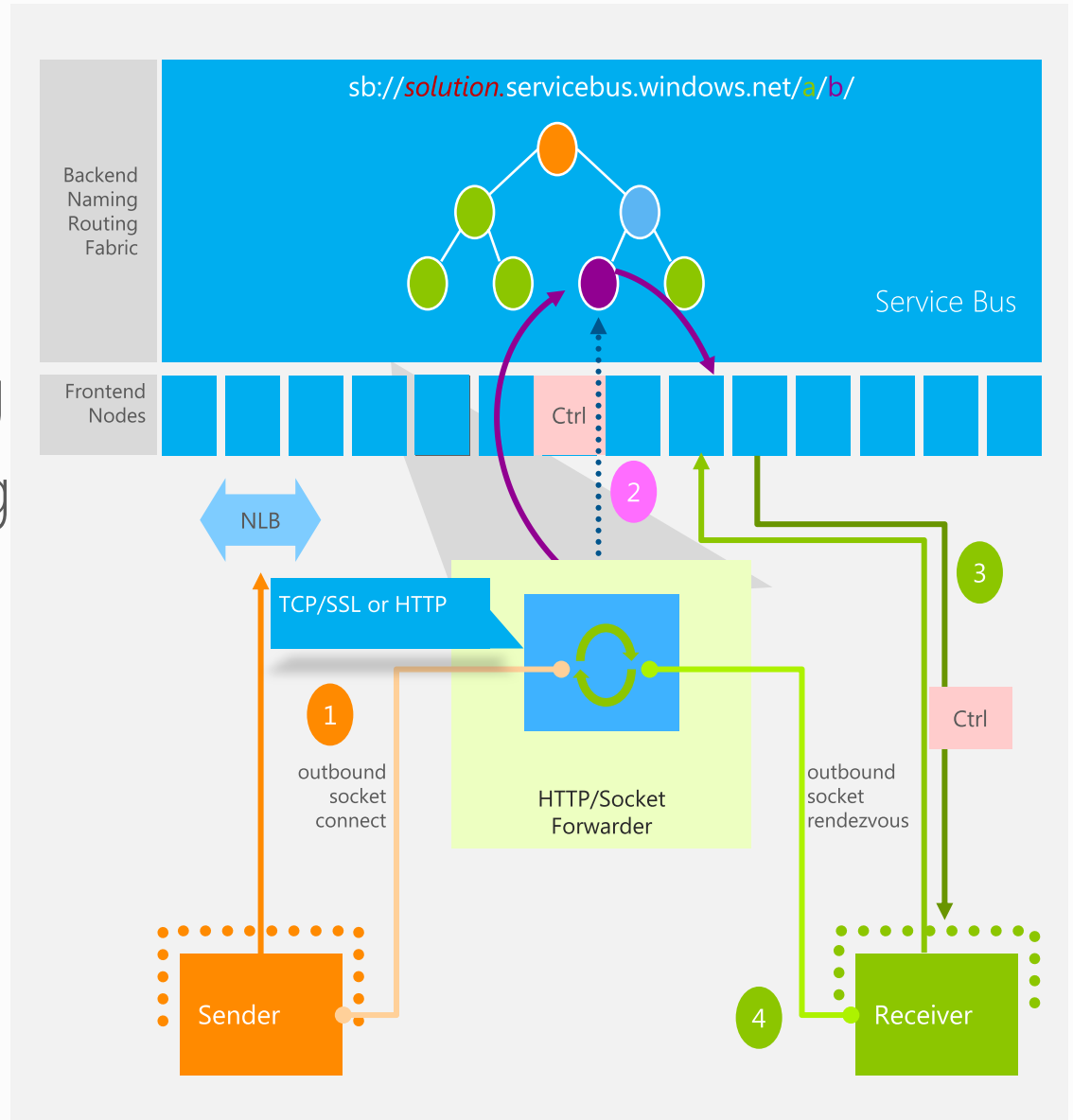
One-way only

No rendezvous overhead



# Rendezvous (TCP & HTTP)

NetTcpRelayBinding  
WebHttpRelayBinding  
BasicHttpRelayBinding  
WS2007RelayBinding  
Rendezvous  
Handshake  
Bi-Directional  
Net.Tcp Full Duplex  
No message size limit



# Hybrid Connect

Special Mode of  
NetTcpRelayBinding

TcpRelayConnection-  
Mode.Hybrid

Starts as relayed connection

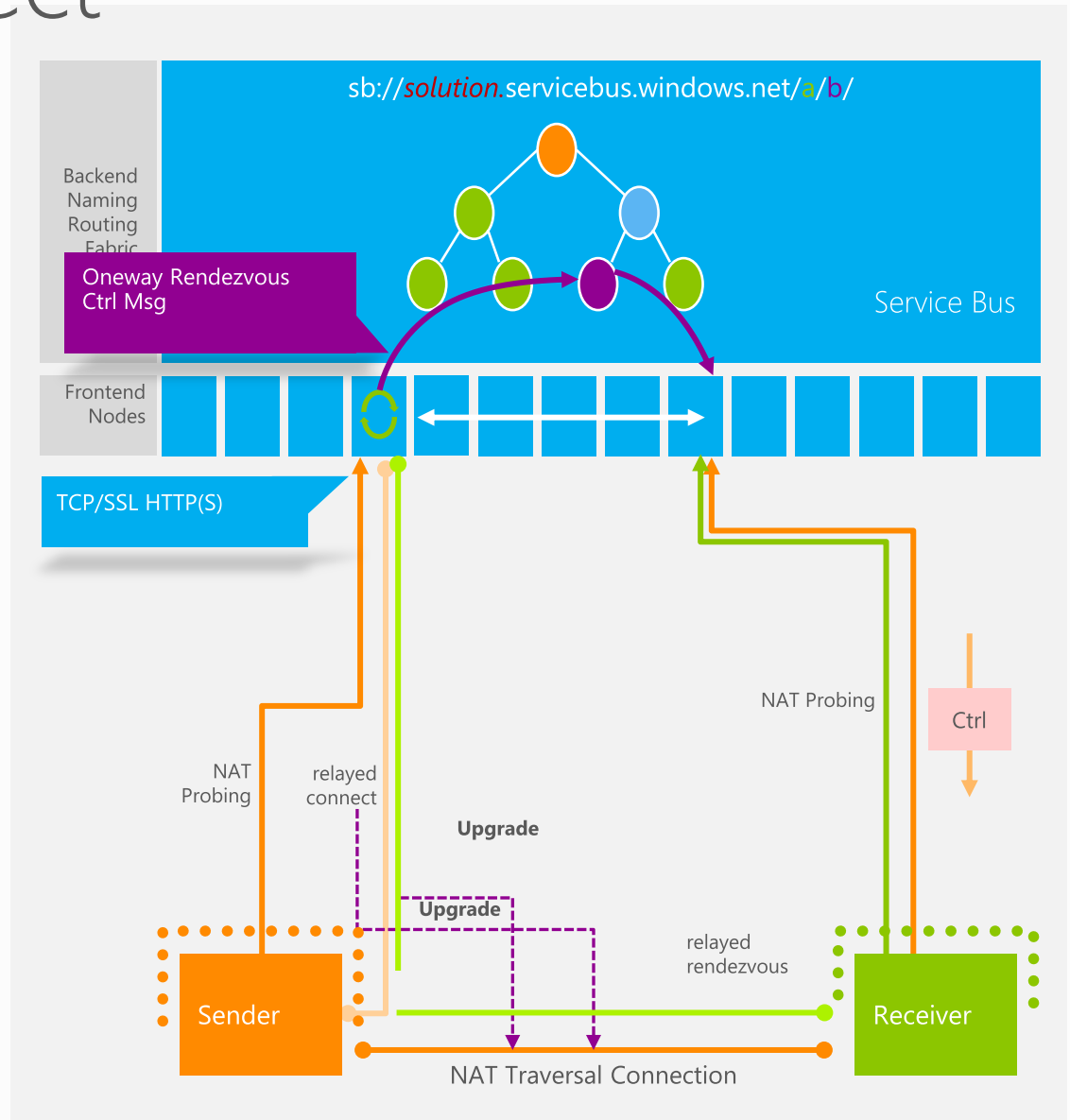
Performs NAT probing and  
behavior prediction

Establishes direct connection  
and upgrades if possible

Upgrade driven by traffic

Takes large transfers off the  
Relay

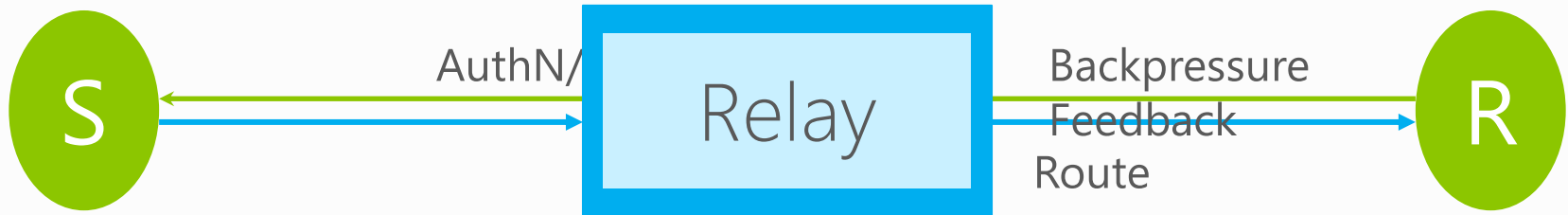
No transfer charges, lower  
latency



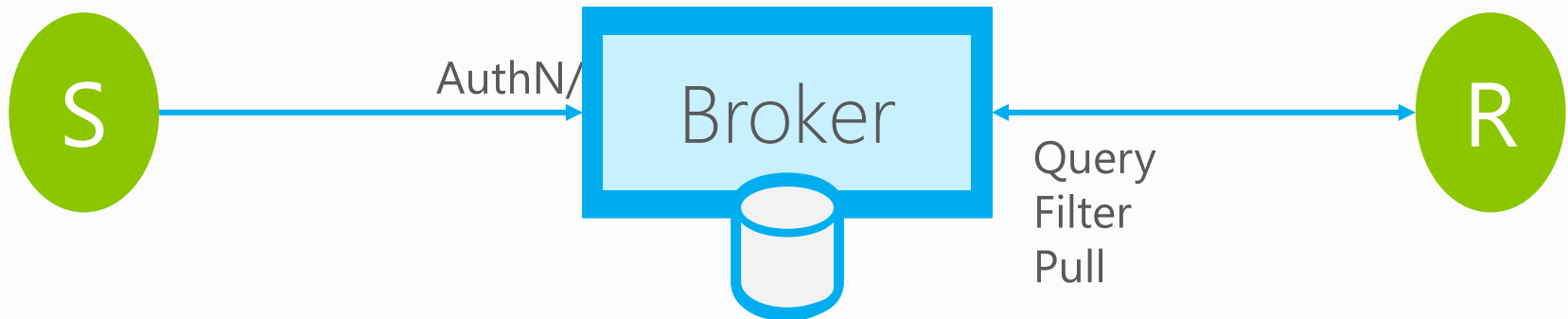
## Demos

- NetOneWayRelay Example
  - NetEventRelay Example
- Hybrid Cloud Application Example

# Relay vs. Message Broker

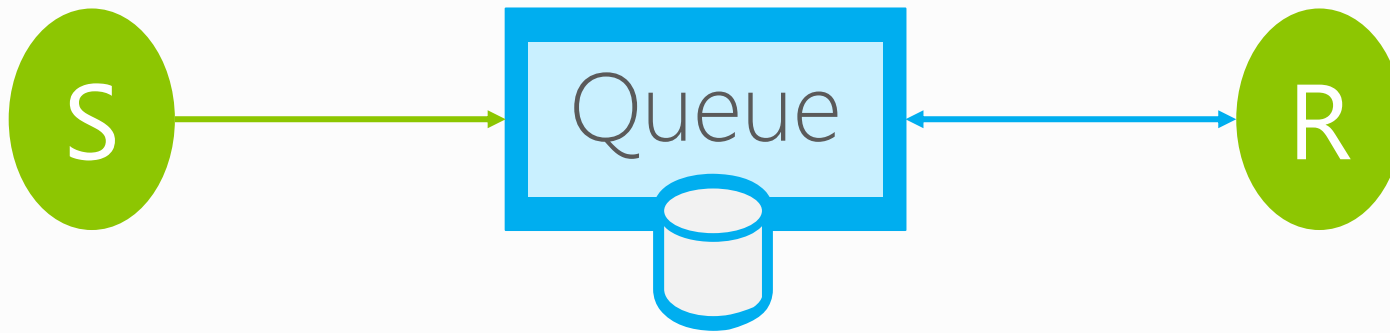


The Relay routes messages 'straight through' with feedback path and network backpressure into sender



Brokers hold messages for retrieval and querying

# Queues



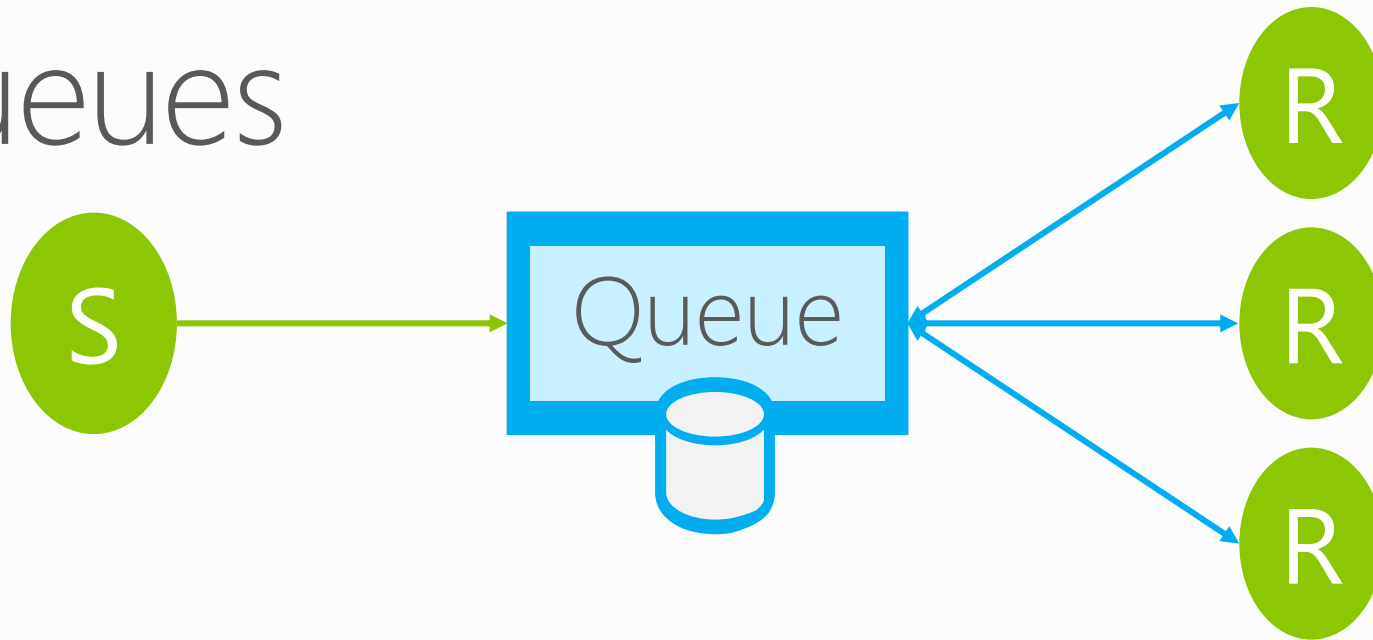
## Load Leveling

Receiver receives and processes at its own pace. Can never be overloaded. Can add receivers as queue length grows, reduce receiver if queue length is low or zero. Gracefully handles traffic spikes by never stressing out the backend.

## Offline/Batch

Allows taking the receiver offline for servicing or other reasons. Requests are buffered up until the receiver is available again.

# Queues

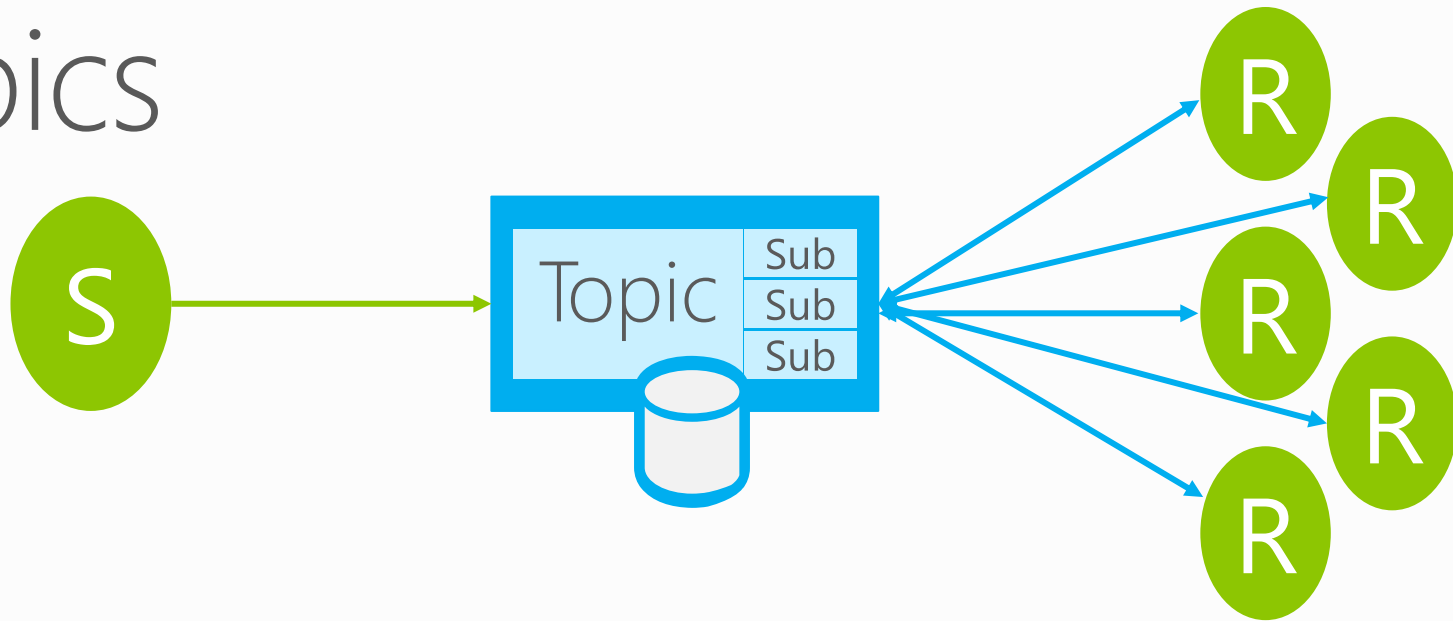


## Load Balancing

Multiple receivers compete for messages on the same queue (or subscription). Provides automatic load balancing of work to receivers volunteering for jobs.

Observing the queue length allows to determine whether more receivers are required.

# Topics



## Message Distribution

Each receiver gets its own copy of each message. Subscriptions are independent. Allows for many independent 'taps' into a message stream. Subscriber can filter down by interest.

## Constrained Message Distribution (Partitioning)

Receiver get mutually exclusive slices of the message stream by creating appropriate filter expressions.

# Subscription Filters

Filter conditions operate on message properties and are expressed in SQL'92 syntax

```
InvoiceTotal > 10000.00 OR ClientRating <3  
ShipDestCtry = 'USA' AND ShipDestState='WA'  
LastName LIKE 'V%'
```

Filters actions may modify/add/remove properties as message is selected

```
SET AuditRequired = 1
```

## Demos

- Queues Example
- Topics Example

- Service Bus
  - ▶ \$0.10 per 100 relay hours
  - ▶ \$0.01 per 10,000 messages
- No charge for billing months beginning before April 1, 2012.
- Quotas (Monthly):
  - ▶ 5 billion Service Bus messages
  - ▶ 2 million Service Bus relay hours

- **Windows Azure Platform Training Kit**
- **Common Prerequisites**
  - ▶ [Microsoft .NET Framework 4.0](#)
  - ▶ [Visual Studio 2010](#)
  - ▶ [Windows Powershell](#)
  - ▶ [Microsoft Internet Information Server 7](#)
  - ▶ [ASP.NET MVC 3](#)
- **Windows Azure SDKs & Tools**
  - ▶ [Windows Azure Tools for Microsoft Visual Studio 1.6](#)  
[Windows Azure Libraries for .NET 1.6](#)

- <http://windowsazurecat.com/2011/08/hybrid-reference-implementation-biztalk-server-windows-azure-sql-azure/>

