**What is Microservices**

Microservices is an architectural style in which a large, complex software applications are

Monolithic VS microservices applications are composed of one or more smaller services. Each of the service constitute as a single business function and executes only that business function.

**Features of Microservices**

* Loosely coupled: Operated independently
* Small and Focused: It can be rewritten
* Language neutral: some services can write in node some of them in java or other languages
* Bounded context: Each microservice has its own execution boundary, and need no to understand the implementation of other microservices.

**Microservices VS SOA**

* Microservices
  + Single distributed system
  + Accelerated realization of benefits
  + Specialization of SOA
* Service Oriented Architecture
  + Large distributed system
  + Overall business

**What is Service Fabric and Its features**

**Types of Services ( from Service Fabric perspective)**

**Stateless microservice**

* Has either no state or it can be retrieved from an external store.
* There can be N instances
* Eg: Web frontends, protocol gateways, Azure cloud services

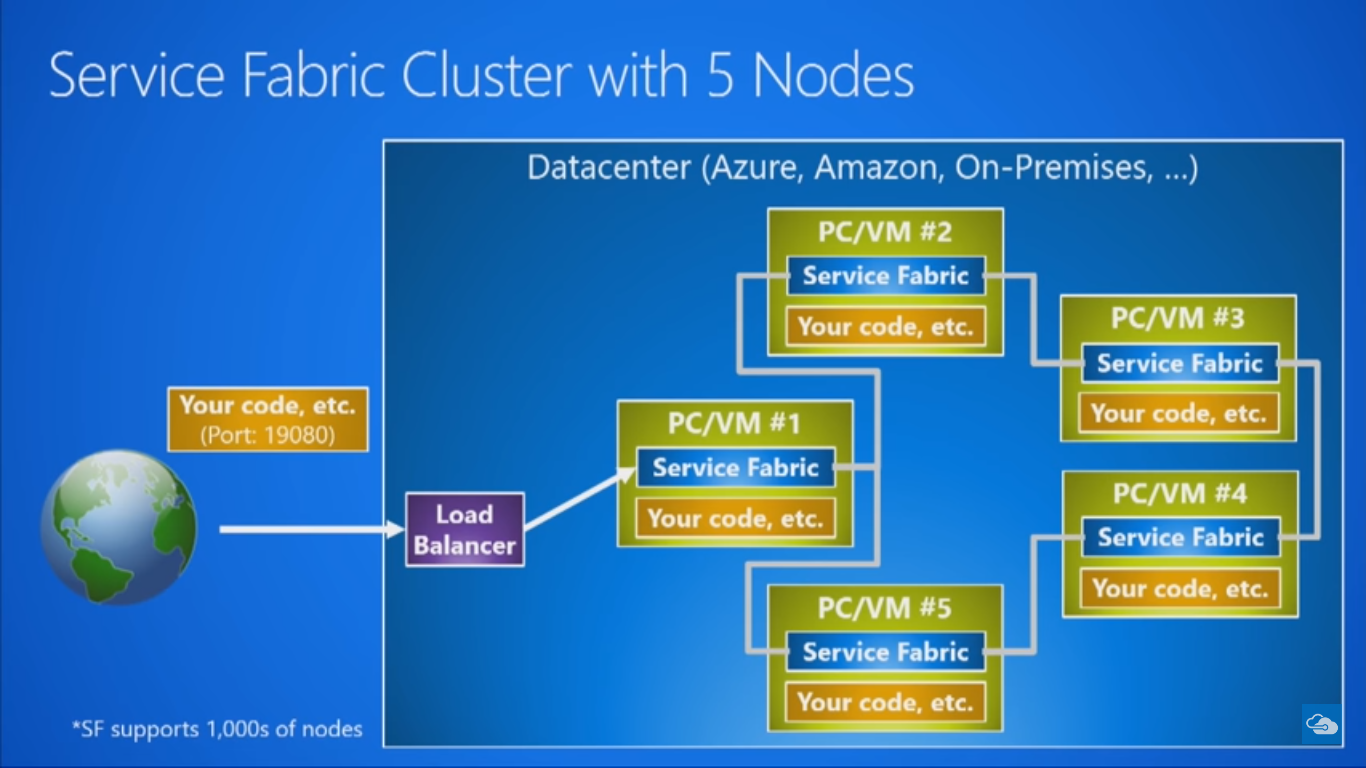
**Stateful microservice**

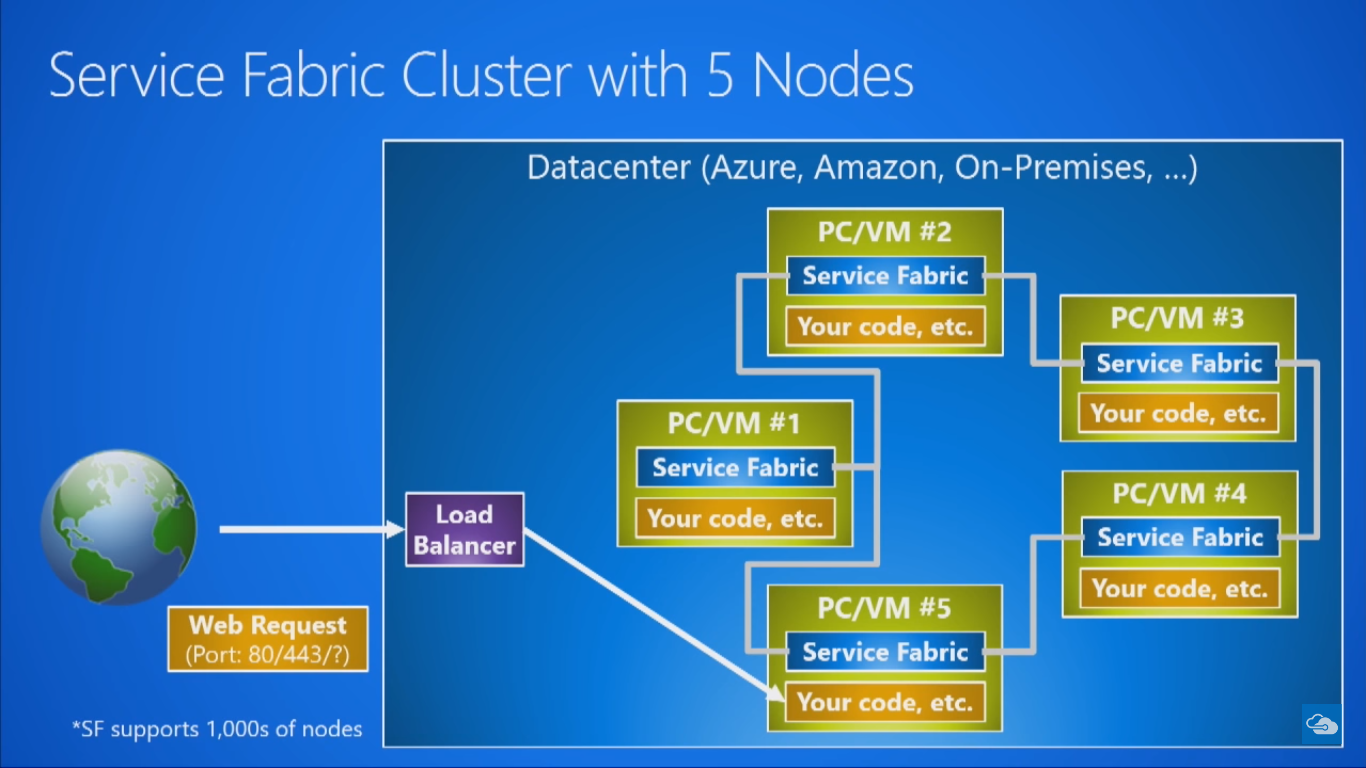
* Maintain hard, authoritative state
* N consistent copies achieved through replication of local persistence
* Eg: Database, documents, workflow, user profile, shopping card etc

**Node**: Each VM that runs the microservice (with SF framework).

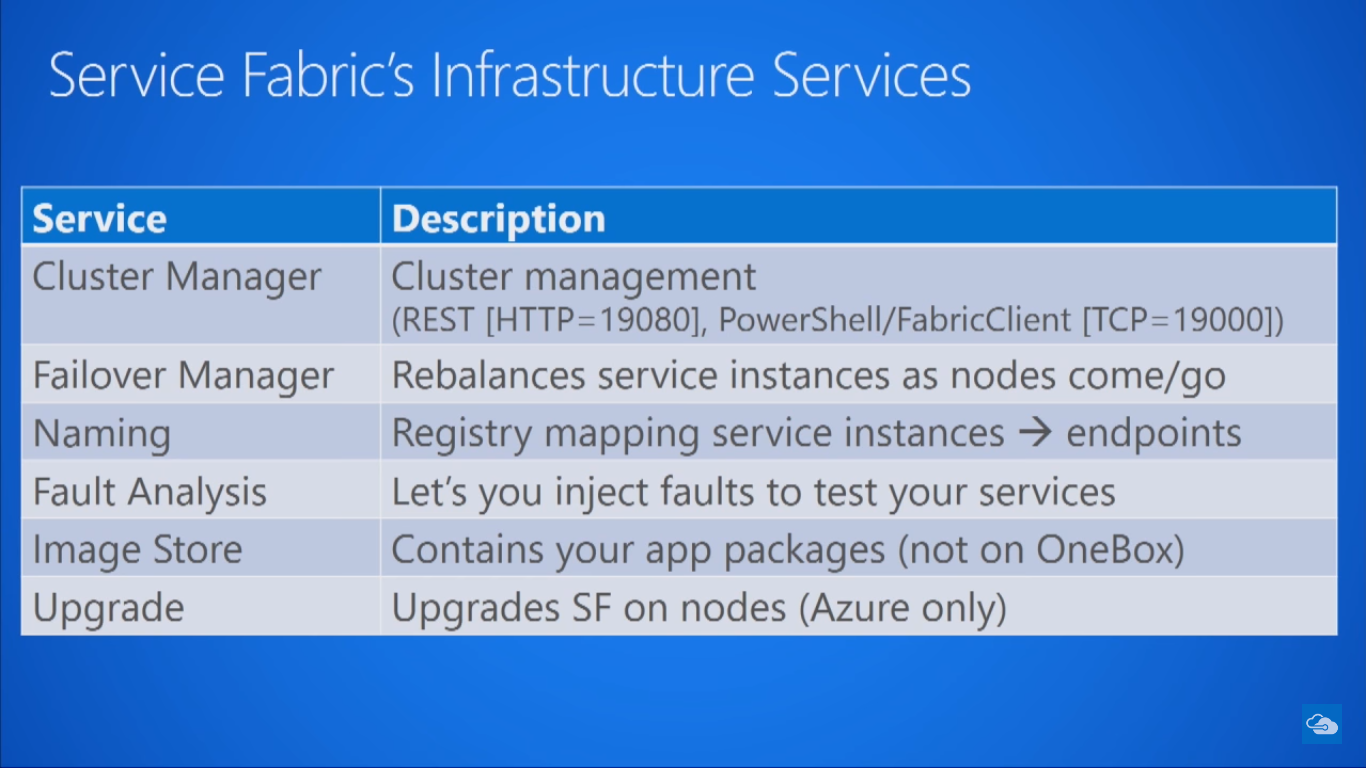
**Cluster**: Bunch of nodes that runs on Azure, Other cloud or on-prem.

A load balancer is required to distribute the work load between nodes in the cluster. In On-premise setup we need to explicitly add the LB, but on Azure, Azure will create and manage the LB for you. You can run the SF clusters on developer’s machines with local cluster installed during the SF SDK installation. That helps to debug the SF apps. We can connect to SF cluster using HTP on port 19080 and using TCP on port 19000. TCP is commonly used when connecting with PowerShell.

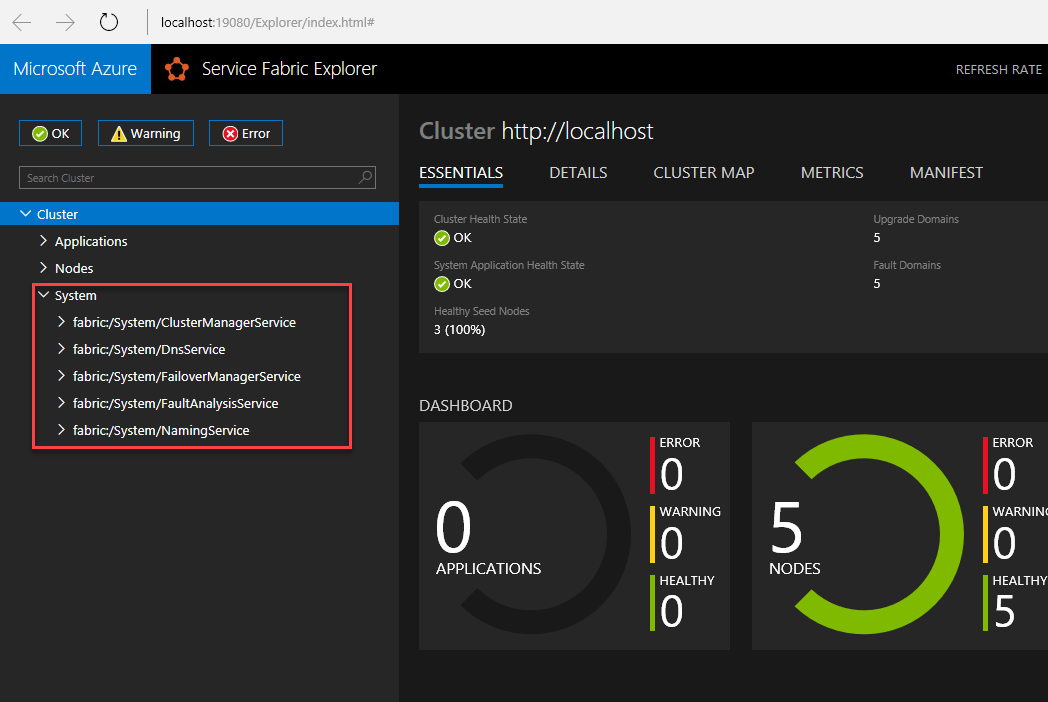




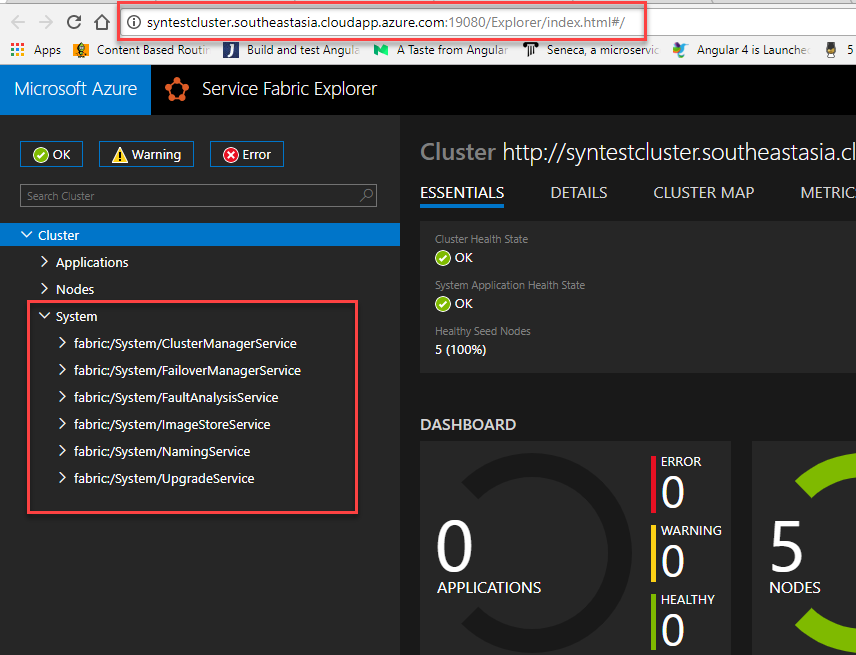
**Service fabric Infrastructure services**



**Local Service Fabric explorer that runs on Developer machine**



**Service Fabric Explorer that runs on Azure**



**Cluster Manager Service**: This runs on all the nodes of the cluster. Responsible for doing cluster management. This component is listening all the nodes for RESTful HTTP traffic and default to 19080 port. We can change the port in ClusterManifest. We have PowerShell commands and Fabric Client .NET library class that listens to port 19000.

**Failover management Service**: Responsible to detect when a new node enters the cluster or a node fails down. It rebalances the service instances as the nodes enters and fails (come and go). Eg: If a node goes down, it identifies which of the services were running on that node and creates a new node and start services on that node so you always get HA.

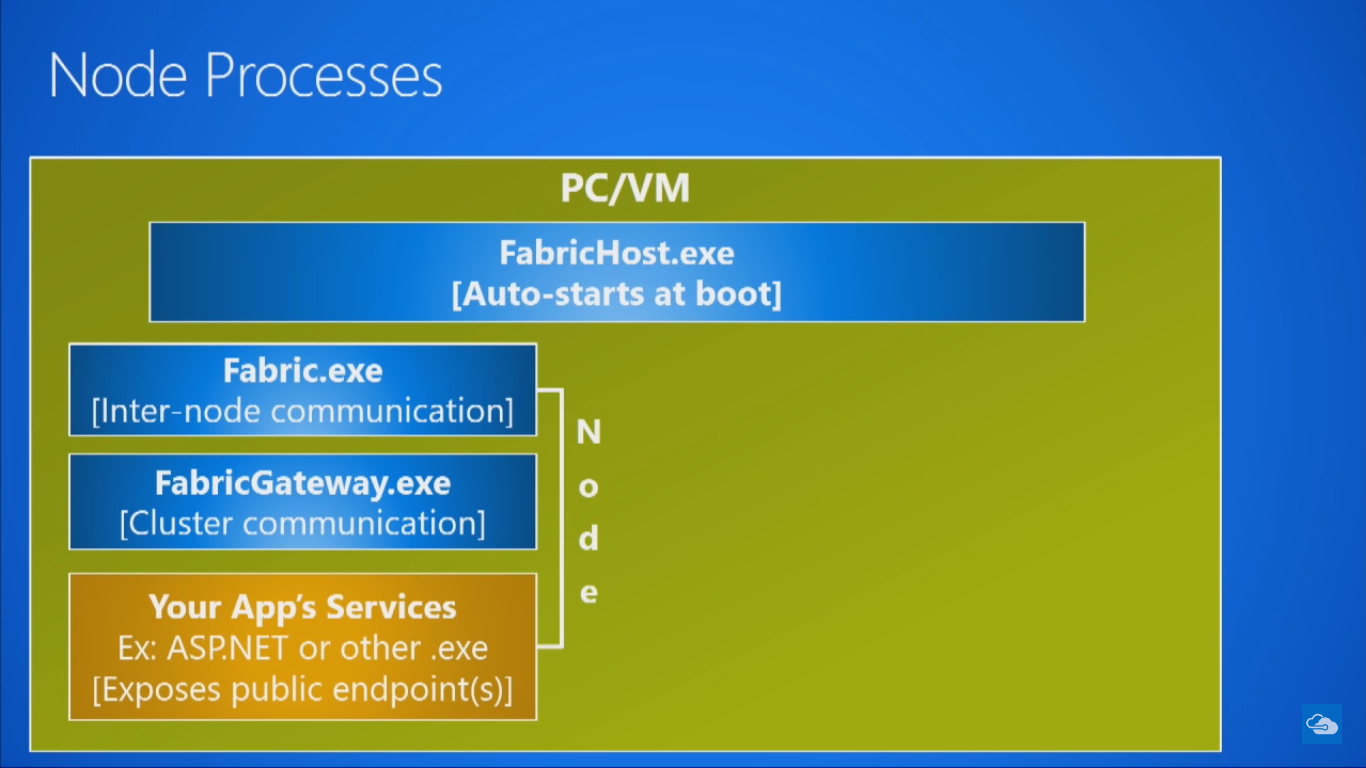
**Naming Service**: It maps the service instances to the endpoints. When a service instance comes to the node it maps it to an endpoint or set of endpoints that can listen different types of traffic(eg: UDP, HTTP, HTTPS etc). One service can look for another service that runs within the same cluster using the name with the help of naming service. If any of the service goes down, the client apps can query the naming service and get the latest endpoints lists and start communicating again.

**Fault analysis service**: The service that allows you to inject faults to test your service. SF ships with this service, it helps to kill nodes, bring them back and shift services around within the cluster to test how your service handle these kinds of failures.

**ImageStore Service**: Contains the application packages. When we deploy the services in to the cluster (eg: dlls, exe other dependencies), they are stored in the image store. When you want to start a new service, SF goes to imagestore gets the dll or exe of the application and copy to the nodes and start the service. ***Image store is not available for OneBox*** (SF installed on Developer machine). Image store exists only in the production environment (ex: Azure).

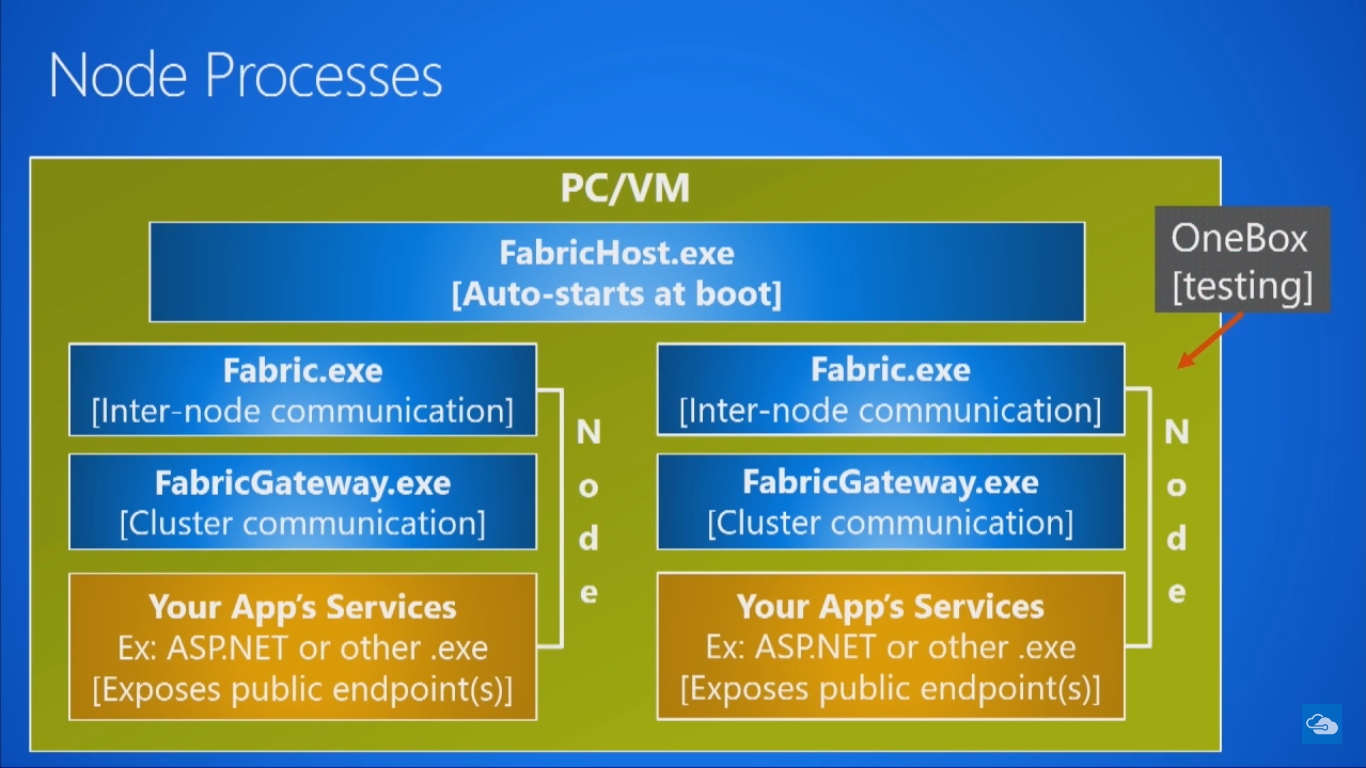
**Upgrade Service**: It is available only on hosted Azure environment. It is responsible to upgrade the SF in azure clusters.

**How SF works on VM/PC?**



FabricHost.exe is a windows NT service that starts whenever the VM/PC starts or the node restarts. This is called the Service Fabric component on that machine. The FabricHost.exe starts/spawns a process called Fabric.exe that is responsible for inter-node communication. FabricHost.exe also starts the FabricGateway.exe which is responsible for Intra node / Cluster communication. These two services make up the node itself. Then the final component in a node is your application, that could be a .NET application or Node application or an external application (eg:java). It exposes the endpoints defined in the application.

Usually, 99.9% of the cases there is one to one relation between the node and the PC. i.e.: every PC runs a single node. But in developer machines(OneBox) there could be multiple nodes that runs on single PC. In developer PC’s one PC can go and run max 5 nodes. When we run the application in developer machine, there will be only one FabricHost.exe running and there could be one or five fabric.exe can runs in a single dev machine.



**Azure Services built with Service Fabric**

