**TỔNG QUAN**

**NỀN TẢNG WINDOWS AZURE**

**GVHD:** Phan Trung Hiếu

**Sinh viên thực hiện:**

Nguyễn Minh Tuấn 07520450

Nguyễn Anh Duy 07520058

**Mục lục**

[1. Tổng quan Windows Azure Platform 3](#_Toc289374438)

[1.1. Kiến trúc Windows Azure Platform 3](#_Toc289374439)

[1.2. Hệ điều hành Windows Azure 3](#_Toc289374440)

[1.3. SQL Azure 7](#_Toc289374441)

[1.4. .NET Services 8](#_Toc289374442)

[1.5. Live Services 13](#_Toc289374443)

[2. Nền tảng Windows Azure cho developer 15](#_Toc289374444)

[2.1. Vai trò của Developer 15](#_Toc289374445)

[2.2. Các kiến thức cần thiết cho Developer 16](#_Toc289374446)

[3. Khái quát về sự phát triển trên Windows Azure platform 16](#_Toc289374447)

[3.1. Kiến trúc dịch vụ 16](#_Toc289374448)

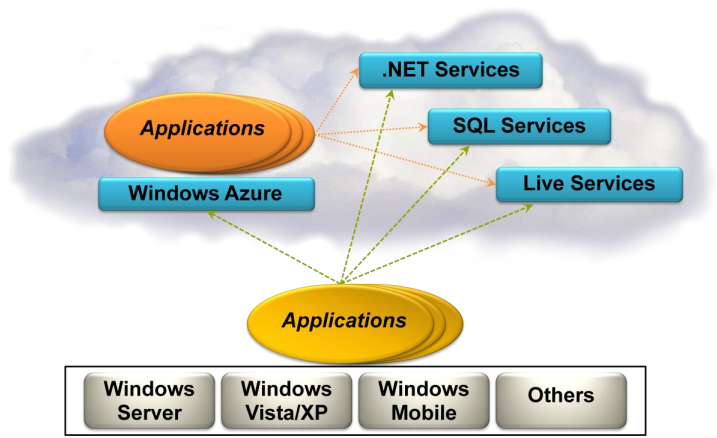
[3.2. Workflow cho developer 17](#_Toc289374449)

# 1. Tổng quan Windows Azure Platform

Windows Azure platform là nền tảng triển khai và phát triển cho việc phát triển dịch vụ “đám mây”. Mỗi thành phần của nền tảng Windows Azure được thiết kế để cung cấp những tính năng đặc trưng cho những dịch vụ “đám mây”.

## Kiến trúc Windows Azure Platform

Windows Azure Platform bao gồm bốn thành phần chính – Windows Azure, SQL Azure, AppFabric (.NET) và Live Services. Hình dưới đây minh hoạ cho bốn thành phần này.

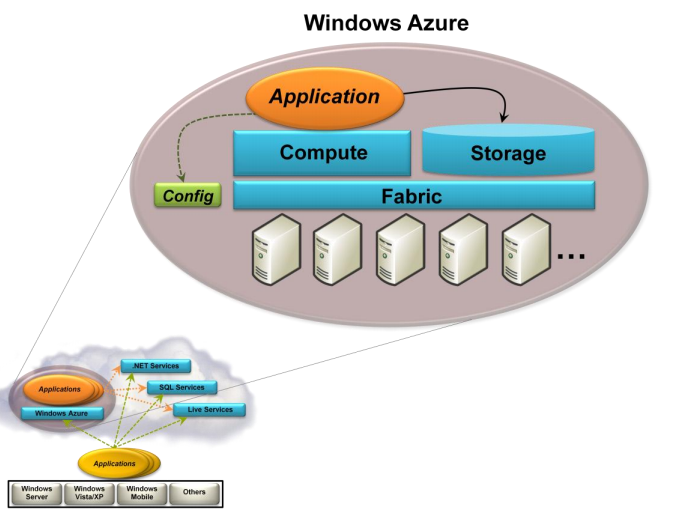


* *Windows Azure:* cung cấp môi trường phát triển dựa trên môi trường Windows cho việc chạy ứng dụng và lưu trữ dữ liệu trên các server trong trung tâm dữ liệu của Microsoft.
* *AppFabric (hay .NET):* cung cấp dịch vụ cơ sở hạ tầng phân tán đối cho những ứng dụng cục bộ và dựa trên “đám mây”.
* *SQL Azure*: cung cấp dịch vụ dữ liệu trong “đám mây” dựa trên SQL Server.
* *Live Services:* thông qua Live Framework, cung cấp sự truy suất dữ liệu từ các ứng dụng Live của Microsoft.

## Hệ điều hành Windows Azure

**1.2.1. Kiến trúc**

Windows Azure là hệ điều hành nằm dưới hỗ trợ chạy các các dịch vụ “đám mây” trên nền Windows Azure Platform. Nó cung cấp môi trường trong lúc thực thi bao gồm web server, dịch vụ tính toán, lưu trữ cơ bản, hàng đợi, dịch vụ quản lý, và load-balancer. Windows Azure cũng cung cấp cho các nhà phát triển khung phát triển cục bộ (local development fabric) hỗ trợ kiểm chứng và xây dựng các dịch vụ trước khi chúng được triển khai trên “đám mây”. Hình sau minh họa ba dịch vụ chính của Windows Azure:



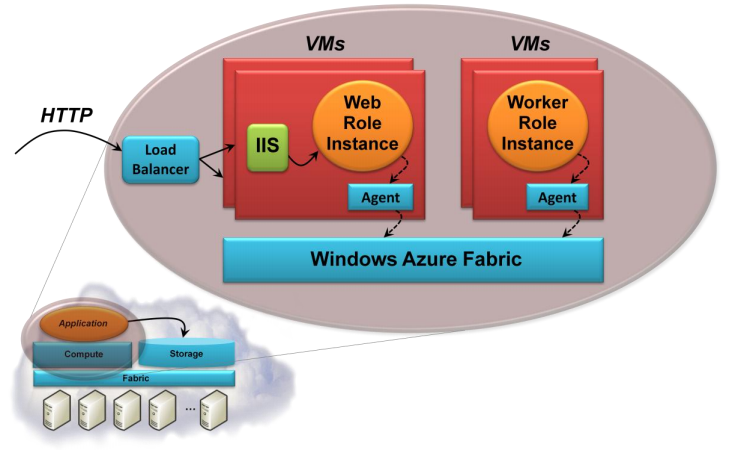
Ba dịch vụ cốt lõi của Windows Azure như sau:

* *Compute*: dịch vụ compute cung cấp hosting cho các dịch vụ trên nền tảng Windows Server 2008 64-bit với Hyper-V. Nền tảng được ảo hóa và thiết kế để có thể co giãn một cách linh động dựa trên như cầu. Nền tảng này chạy trên IIS7 hỗ trợ các ứng dụng ASP.NET.
* *Storage*: có ba kiểu lưu trữ mà Windows Azure hỗ trợ: table, blob và queue. Kiểu dữ liệu bảng trong Windows Azure không phải là bảng dữ liệu quan hệ truyền thống giống như bảng dữ liệu trong SQL Server, thay vào đó nó cung cấp khả năng lưu trữ dữ liệu cấu trúc và mô hình dữ liệu độc lập được gọi là mô hình thực thể. Windows Azure blob được thiết kế để lưu trữ khối lượng lớn các kiểu dữ liệu video, hình ảnh và âm thanh. Windows Azure queue là những kênh giao tiếp không đồng bộ cho việc kết nối giữa các dịch vụ và ứng dụng không chỉ trong Windows Azure mà còn những ứng dụng on-premise.
* *Fabric (Management):* dịch vụ quản lý cung cấp cơ sở hạ tầng tự động và khả năng quản lý dịch vụ đối với các dịch vụ “đám mây” Windows Azure. Khả năng này bao gồm sự ủy nhiệm tự động của các máy ảo và triển khai các dịch vụ trên chúng, cũng như cấu hình các switch, truy cập router cho việc duy trì người dùng được địch nghĩa bởi các dịch vụ.

**1.2.2. Chạy ứng dụng**

Trên Windows Azure, một ứng dụng tượng trưng cho nhiều instance (thực thể). Mỗi instance chạy trên máy ảo của riêng mình (VM). Các máy ảo chạy hệ điều hành Windows Server 2008 64-bit.

Nhưng ứng dụng Windows Azure không thể thấy được máy ảo đang chạy trong chính nó. Một nhà phát triển (developer) không được phép cung cấp máy ảo riêng của chính mình để có thể chạy Windows Azure. Thay vào đó phiên bản CTP cho phép nhà phát triển tạo ra những ứng dụng .NET 3.5 sử dụng Web role và/hoặc Worker role. Hình sau là mô hình chạy ứng dụng



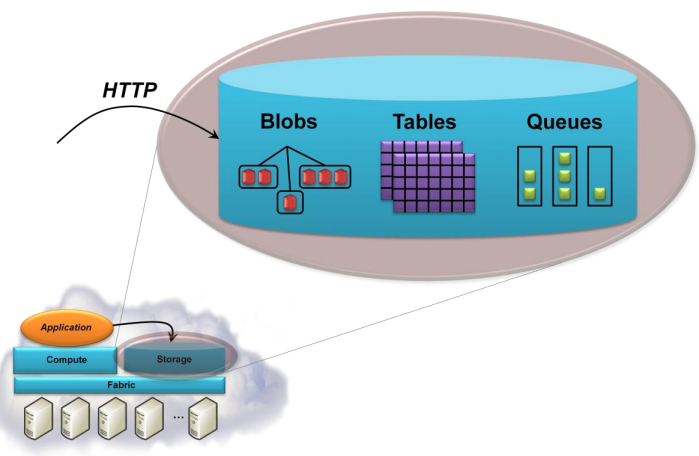
Mỗi Web role chấp nhận yêu cầu HTTP (hay HTTPs) thông qua IIS7. Một Web role có thể được thi hành bằng cách sử dụng ASP.NET, WCF hay một công nghệ .NET Framework mà làm việc trên IIS.

Một thực thể Woker role không thể chấ nhận các yêu cầu trực tiếp từ bên ngoài – Nó không được phép chấp nhận bất kỳ một kết nối mạng nào và IIS cũng không chạy trên máy ảo của nó. Thay vào đó, nó lất dữ liệu đầu vào từ thực thể Web role, thông qua một hàng đợi trong bộ lưu trữ Windows Azure.

Bất cứ khi nào ứng dụng chạy trên thực thể Web role hay Worker rolee, mỗi máy ảo cũng chứa một Windows Azure agent cho phép ứng dụng tác động qua lại với Windows Azure fabric. Agent này trình bày một API được định nghĩa bởi Windows Azure để cho thực thể ghi vào file log Windows Azure, gửi các cảnh báo đến thực thể sở hữu thông qua Windows Azure fabric.

**Truy cập dữ liệu**

Những ứng dụng làm việc với dữ liệu trong nhiều cách khác nhau, đôi lúc là blob, hay dữ liệu cấu trúc. Việc lưu trữ trên Windows Azure cần 3 thành phần như hình sau:



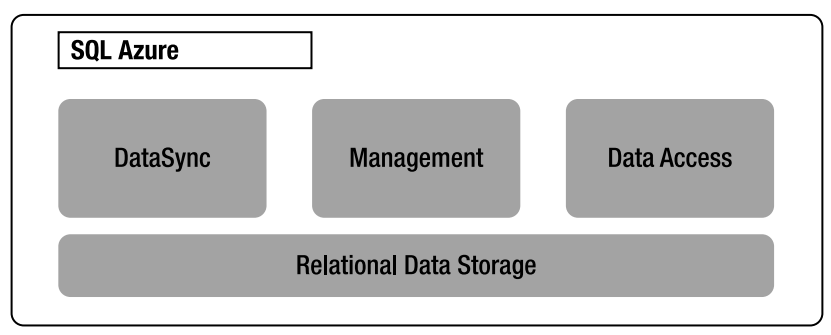
Cách đơn giản nhất để lưu trữ dữ liệu trong Windows Azure là blob. Blob chỉ là lựa chọn đúng với một số loại dữ liệu, nhưng trong nhiều trường hợp chúng không có cấu trúc. Để cho phép các ứng dụng làm việc với dữ liệu trơn tru, Windows Azure storage cung cấp dữ liệu bảng, nhưng đây không phải là bảng dữ liệu quan hệ. Thực ra, mặc dù chúng được gọi là “bảng”, dữ liệu mà chúng chứ thực sữ được lưu trong cấu trúc phân cấp đơn giản của các entity với các thuộc tính, thuộc tính có thể là một số kiểu dữ liệu như int, string,Bool hay DateTime. Ngoài ra nó cũng không sử dụng SQL, ứng dụng truy cập dữ liệu sử dụng ngôn ngữ truy vấn với cấu trúc LINQ. Dữ liệu một bảng có thể rất lớn với hàng nghìn tỉ thực thể được nắm giữ với dữ liệu TB.

Blob và bảng đều tập trung trên dữ liệu lưu trữ. Lựa chọn thứ ba trong Windows Azure storage, queue (hàng đợi) lại có mục đích khác. Vai trò chính của hàng đợi là cug cấp cách để các thực thể Web role để giao tiếp với thực thể Worker role. Ví dụ, người dùng có thể submit một yêu cầu để thực hiện một công việc tính toán thông qua một trang Web được thực thi bởi Windowa Azure Web role. Thực thể Web role nhận yêu cầu và ghi lại một tin nhắn vào hàng đợi mô tả công việc cần được thực hiện. Một thực thể Worker role mà chờ đợi trong hàng đợi sẽ đọc tin nhắn và thực hiện công việc được chỉ định.

Windows Azure storage có thể được truy cập bởi một ứng dụng Windows Azure hay bởi ứng dụng đang chạy nơi nào náo. Trong cả hi trường hợp, ba kiểu Windows Azure storage sử dụng quy ước của REST để xác định và trình bày dữ liệu. Mọi thức được đặt tên bằng cách sử dụng URI và được truy cập với chuẩn HTTP.

## 1.3. SQL Azure

SQL Azure là cơ sở dữ liệu quan hệ trong trên nền Windows Azure. Nó cung cấp hệ quản trị cơ sở dữ liệu như là một dịch vụ, và được xây dựng trên nền tảng của SQL Server. Trong phiên bản hiện tại (CTP), các nhà phát triển có thể truy cập SQL Azure bằng cách sử dụng dòng dữ liệu bảng (TDS – Tabular data stream), là cơ chế chuẩn cho việc truy xuất các thực thể SQL Server on-premise thông qua SQL client. SQL client có thể là bất kỳ client nào như ADO.NET, LINQ, ODBC, JDBC, ADO.NET Entity Framework, hay ADO.NET Data Services.



Các dịch vụ chính của SQL Azure bao gồm:

* *Relational Data Storage*: là thành phần cơ bản của SQL Azure dựa trên SQL Server. Thành phần này thể hiện khả năng truyền thồng của SQL Server như table, index, view, store procedure và trigger.
* *Data Sync*: cung cấp sự đồng bộ hóa và sự tập hợp dữ liệu đến và từ SQL Azure đến doanh nghiệp, workstation, cộng sự và các thiết bị tiêu dùng bằng cách sử dụng Microsoft Sync Framework.
* *Management*: thành phần quản lý cung cấp các dịch vụ load-balancing, billing, metering, provisioning tự động và các khả năng bảo mật cho SQL Azure. Dựa trên SLA, mỗi cơ sở dữ liệu được tái tạo trên một server chính và hai server thứ cấp. Trong trường hợp failover, sự chuyển chuyển từ server chính và server thứ cấp được diễn ra tự động mà không có sự gián đoạn nào.
* *Data Access*: thành phần này định nghĩa các phương thức khác nhau cho việc truy xuất SQL Azure. Hiện tại, SQL Azure sẽ hỗ trợ Tabbular Data Stream (TDS), bao gồm trong ADO.NET, Entity Framework, ADO.NET Data Srver, ODBC, JDBC, và LINQ client. Các nhà phát triển có thể truy cập hoặc là trực tiếp từ các ứng dụng on-premise hay thông qua dịch vụ “đám mây” trong Windows Azure. Bạn cũng có thể định vị một Windows Azure compute cluster và một thực thể SQL Azure chung với nhau để truy xuất dữ liệu nhanh hơn.

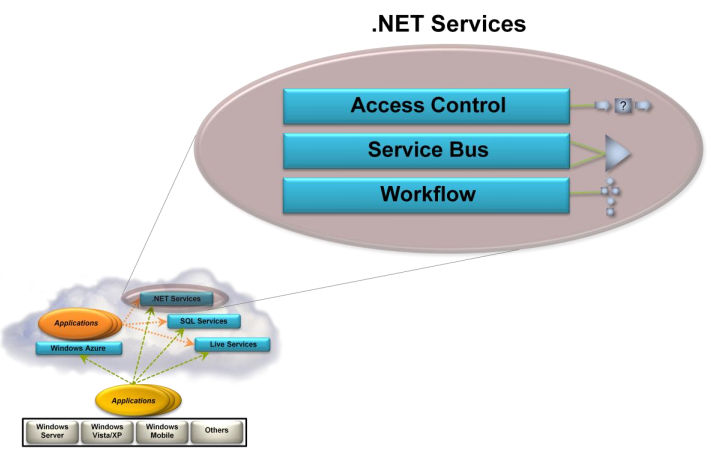
## 1.4. .NET Services

**Kiến trúc**

.NET Service là cỗ máy middleware của nền tảng Windows Azure cung cấp dịch vụ access control, service bus và workflow.

AppFabrichas là một kiến trúc hướng đối tượng và cho phép tạo ra access control và tin nhắn phân tán thông qua “đám mây” và enterprise. AppFabricto là sự tích hợp cở cốt lõi của nền Windows Azure, bởi vì nó cung cấp khả năng kết nới và khả năng truyền tin nhắn giữa các ứng dụng phân tán. Nó cũng cung cấp các khả năng chi các ứng dụng tích hợp và quy trình nghiệp vụ không chỉ giữa các dịch vụ “đám mây” mà còn các ứng dụng on-premise.

AppFabricalso cung cấp mội trường phát triển được tích hợp vào Visual Studio .NET 2008 SP1 và các phiên bản mới hơn. Các nhà phát triển có thể xây dựng WCF – giống các dịch vụ trong Visual Studio .NET và xuất bản endpoint lên “đám mây” bên trong môi trường thiết kế Visual Studio .NET. Hình sau minh họa các dịch vụ cốt lõi của .NET Services:



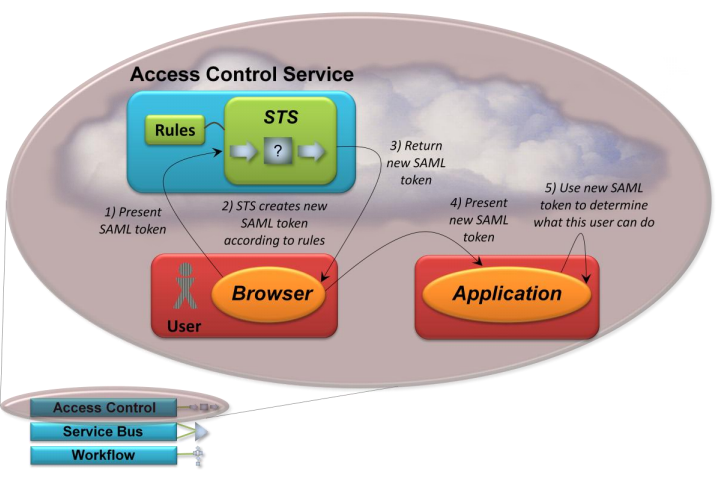
Ba thành phần chính của AppFabricare:

* *Access Control*: đây là thành phần cung cấp rules-driven, access control dựa trên yêu cầu cho các ứng dụng phân tán. Bạn có thể định nghĩa các quy tắc dựa trên yêu cầu và các vai trò ủy quyền trong “đám mây” cho việc truy cập ứng dụng on-premise cũng như các dịch vụ “đám mây”.
* *Service Bus*: là dịch vụ service bus .NET chung trên Internet. Nó giống như Enterprise Service Bus (ESB). Không giống như ESB, AppFabricServiceBus được thiết kế cho phạm vi Internet và sự truyền tin nhắn với viễn cảnh cross-enterprise và cross-cloud. Service bus cung cấp mô hình truyền tin nhắn như publish/subscribe, point-to-point và các hàng đợi cho tin nhắn tra đổi qua các ứng dụng phân tán trong “đám mây” cũng như on-premise.
* *Workflow:* tạo các ứng dụng kép, như các ứng dụng doanh nghiệp, đòi hỏi sự logic mà phối hợp sự tương tác qua lại trong số các phần khác nhau. Được xây dựng trên Windows Workflow Foundation (WF), dịch vụ Workflow cho phép chạy loại logic này trên “đám mây”.

**Access Control Service**

Làm việc với identity là một phần nền tảng của hầu hết các ứng dụng phân tán. Dựa trên thông tin identity của người dùng, mà ứng dụng quyết định người dùng được cho phép làm những gì. Để truyền thông tin, những ứng dụng có thể dựa trên các dấu hiệu được định nghĩa sử dụng Security Assertion Markup Language (SAML). Các dấu hiệu được tạo ra bởi Security Token Service (STS).

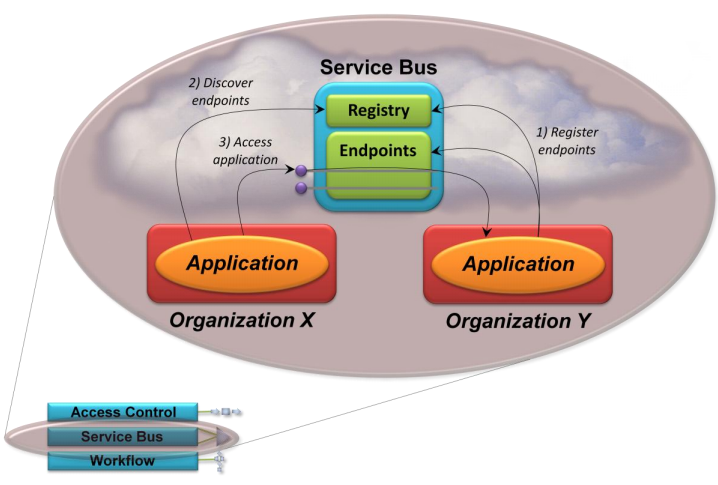
Một khi một client (giống như Web browser) có một dấu hiệu riêng, nó có thể biệu hiện đấu hiệu này cho ứng dụng. Ứng dụng sau đó sẽ sử dụng dấu hiệu này để xem xét người dùng được làm những gì.



**Service Bus**

Giả sử rằng có một ứng dụng chạy bên trong tổ chức của bạn và bạn muốn ứng dụng này khả dụng đối với phần mềm trong các tổ chức khác thông qua Internet. Nhìn thoáng qua, đây có thể dừng như là một khó khăn. Thừa nhận rằng ứng dụng của bạn cung cấp tính năng như là Web service, bạn chỉ có thể làm cho Web services có thể nhìn thấy được đối với bên ngoài. Nhưng khi bạn thực sự cố gắng làm điều này, một số vấn đề có thể xuất hiện.

Đầu tiên, làm thế nào ứng dụng ở tổ chức khác tìm thấy các endpoint mà có thể kết nối đến dịch vụ của bạn? Có thể định vị được ứng dụng của bạn bằng một số loại registry. Và một lần nữa khi tìm thấy ứng dụng này, thì làm thế nào những yêu cầu từ phần mềm của của tổ chức khác vược qua đến ứng dụng của bạn? Network Address Translation (NAT) có thể giúp bạn điều này. Và khi đó câu hỏi lại được đặt ra, làm sao các yêu cầu có thể vượt qua firewall? Việc mở các cổng firewall để cho phép truy cập ứng dụng của bạn là điều khả thi nhưng hầu hết các quản trị mạng điều kho6g chấp nhận điều này. Service bus là mô hình có thể giải quyết vấn đề này:



Để bắt đầu, ứng dụng của bạn đăng ký một hay nhiều endpoint với Service bus. Services bus cấp cho ứng dụng của bạn một URI root. Ứng dụng của bạn cũng phải mở một kết nối với Service bus cho mỗi endpoint mà nó hiện ra. Serivce bus giữ cho kết nối này được mở, mà sẽ giải quyết được hai vấn đề. Thứ nhất, NAT không còn là một trở ngại, từ đó sự lưu thông trên kết nối này với Service bus sẽ luôn luôn được định tuyến đối với ứng dụng của bạn. Thứ hai, bởi vì kết nối này được khởi đầu từ bên trong firewall, nên sẽ không có vấn đề gì trong việc truyền thông tin trở lại ứng dụng – firewall sẽ không khóa sự lưu thông này.

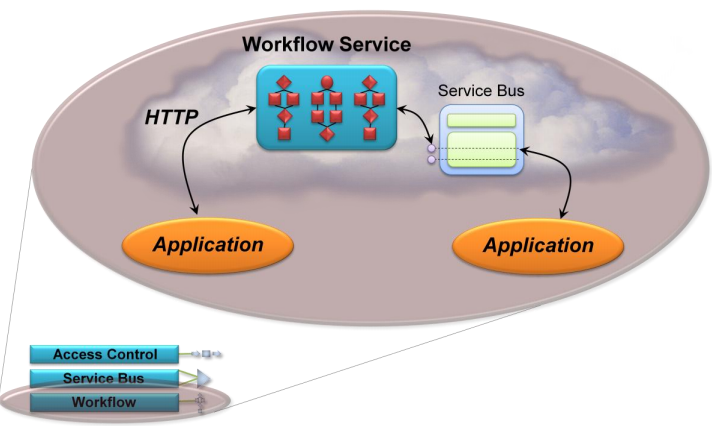
Khi ứng dụng trong tổ chức khác muốn truy cập ứng dụng của bạn, nó sẽ liên hệ với registry trên Service bus. Yêu cầu này sử dụng Atom Publishing Protocol, và nó trả về một văn bản AtomPub service với tham chiếu đến endpoint của ứng dụng của bạn. Khi các bước trên được thực hiện, nó có thể gọi các dịch vụ được cung cấp thông qua endpoint. Mổi yêu cầu được nhận bởi Service bus, sau đó sẽ truyền đến ứng dụng của bạn, và các phản hồi sẽ truyền ngược lại. Mặc dầu không được hiển thị trên hình, Service bus đã thiết lập một kết nối trực tiếp giữa một ứng dụng và client của nó bất cứ khi nào có thể, làm cho việc giao tiếp hiệu quả hơn.

Ngoài việc làm việc giao tiếp dễ dàng hơn, Service bus còn nâng cao tính bảo mật. Bởi vì các client chỉ thấy được một địa chỉ IP được cung cấp bởi Service Bus, nên không cần hiển thị bất kỳ địa chỉ IP nào bên trong tổ chức của bạn. Service Bus hành động giống như một DMZ bên ngoài, cung cấp một lớp hàn động gián tiếp để ngăn cản các cuộc tấn công. Và cuối cùng, Service Bus được thiết kế để được sử dụng với Access Control Service, cho phép sự vận chuyển yêu cầu dựa trên quy luật. Thực ra, Service Bus chỉ chấp nhận các ký hiệu được phát ra bởi Access Control Service STS.

**Workflow Service**

Windows Workflow Foundation là một công nghệ tổng quan cho việc tạo ra các ứng dụng dựa trên workflow. Các ứng dụng WF có thể là một lựa chọn tốt cho việc phối hợp với nhiều loại công việc.

Bằng cách cung cấp một tiến trình chủ cho ứng dụng dựa trên nền WF 3.5, nó cho phép các nhà phát triển tạo ra các workflow chạy trên “đám mây”. Hình sau minh họa workflow service:



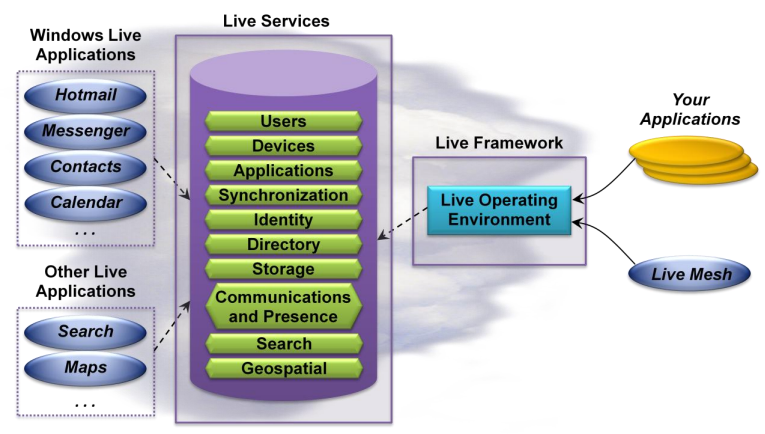
Mỗi WF workflow được thực thi bằng cách sử dụng một số “hoạt động”, thể hiện mà đỏ trong hình. Mỗi hoạt động thực hiện một hành động được định nghĩa, như gửi và nhận tin nhắn, thực thi một câu lệnh If, hay điều khiển một vòng lập While. WF cung cấp một tập các hành động chuẩn được gọi là Base Activity Library (BAL), và Workflow Service cho phép các ứng dụng mà nó chạy sử dụng các hành động con của BAL. Dịch vụ cũng cung cấp một vài hoạt động của chính nó. Ví dụ, các ứng dụng mà nó làm chủ có thể giao tiếp các phần mềm khác sử dụng hoặc là giao thức HTTP hay Service Bus. Workflow Service cũng cung cấp các hoạt động cho việc làm việc với các tin nhắn XML, một đòi hỏi cơ bản cho ứng dụng tích hợp.

Tuy nhiên việc chạy trên “đám mây” có một vài giới hạn. Ví dụ như, các ứng dụng dựa trên WF chạy trên Workflow Service chỉ có thể sử dụng mô hình luồng làm việc dãy của WF.

## 1.5. Live Services

Microsoft Live Service là một tập hợp các ứng dụng trung tâm và các nền tảng như Identity Management, Search, Geospatial, Communication, Storage và Synchronization. Live Services và Live Framework là điều thường hay gây bối rối. Live Framework là mô hình phát triển hợp nhất cho việc xây dựng các ứng dụng cho Live Service. Live Framework cung cấp các khả năng cho việc xây dựng các ứng dụng đồng bộ được gọi là ứng dụng Mesh cho dữ liệu đồng bộ qua nhiều máy tính và thiết bị di động.

Live Framework cũng cung cấp một môi trường phát triển local được tích hợp vào trong Visual Studio .NET và môi trường runtime local cho sự phát triển và kiểm chứng ứng dụng on-premise. Hình sau là các thành phần cốt lõi của Live service:



Những thành phần được mô tả sau:

* *Mesh Services*: dịch vụ này cung cấp sự truy xuất đến người dùng, thiết bị, ứng dụng và sự đồng bộ dữ liệu.
* Dịch vụ ngườ dùng cung cấp sự quản lý và chia sẻ tài nguyên qua các thiết bị và người dùng khác.
* Dịch vụ thiết bị cung cấp sự quản lý, truy xuất, chia sẻ và bảo mật các thiết bị người dùng qua Internet.
* Dịch vụ ứng dụng cung cấp sự triển khai, cấu hình qua Live Mesh.
* Dịch vụ đồng bộ dữ liệu cung cấp sự đồng bộ dữ liệu và metadata qua nhiều thiết bị, ứng dụng và “đám mây” Live mesh.
* *Identity Service*: cung cấp sự phản lý id và ủy quyền sự xác thực qua Live Service. Ví dụ như Windows Live Identiy Provider.
* *Directory Service*: quản lý các quan hệ của các User, Identies, Devices, Application và mạng kết nối. Ví dụ như Relationship giữa các user và thiết bị trên Live Mesh.
* *Storage*: quản lý việc lưu trữ dữ liệu tạm thời và cứng cho User, Device, và Application trong Mesh. Ví dụ Windows Live Skydrive và Live Mesh storage.
* Communication & Presence: cung cấp cơ sở hạ tầng giao tiếp giữa các thiết bị và ứng dụng, quản lý thông tin hiện các kết nối và hiển thị, ví dụ Windows Live Messenger và Notification API.
* *Search*: cung cấp các khả năng tìm kiếm cho người dùng, web site và các ứng dụng, ví dụ Bing.
* *Live Framework*: là mô hình lập trình thống nhất Live Service thông qua platform, các ngôn ngữ lập trình và thiết bị. Có ba thành phần chính của dịch vụ Live Framework là:
* Live Operating Environment (LOE): là môi trường thời gian thực cho các đối tượng Live Service trong “đám mây” cũng như môi trường thiết bị cục bộ. Các nhà phát triển có thể kết nối đến LOE trên đám mây thiết bị cụ bộ. LOE cung cấp khả năng hoạt động offline đến Live Mesh. LOE cũng làm cho Live Mesh có thể đồng bộ dữ liệu hoạt động offline với LOE “đám mây” và các thiết bị khác.
* Resource Model: thể hiện ra các đối tượng tài nguyên LOE như Data, Application, Communication, Mesh và Identities thông qua các giao tức và định dạng HTTP, XML, ATOM, và RSS. Bạn có thể sử dụng bất kỳ ngôn ngữ lập trình nào hỗ trợ các giao thức để truy cập tài nguyên LOE.
* Live Programming Model: cung cấp mô hình lập trình chung từ JavaScript, Silverlight và ngôn ngữ lập trình .NET. Bạn có thể xây dựng Client, Web, và Rich Internet Application sử dụng bất kỳ ngôn ngữ nào được đề cập ở trên và triển khai chúng trên Live Mesh.

# 2. Nền tảng Windows Azure cho developer

Tất cả Windows Azure, SQL Azure, .NET Services và Live Services SDKs tách biệt, nhưng Visual Studio .NET và .NET Framework là công cụ lập trình phổ biến được sử dụng xây dựng các ứng dụng cho tất cả các thành ph6a2n của Windows Azure. Windows Azure SDK và Live Famework SDK có các bộ khung phát triển cục bộ cùng với môi trường “đám mây”. Các developer dùng môi trường .NET hiện có cho việc phát triển các dịch vụ trên Windows Azure Platform.

## 2.1. Vai trò của Developer

Các nhà phát triển khác nhau có thể quan tấm đến sự phát triển các ứng dụng Windows Azure sau:

* *Nhà phát triển doanh nghiệp*: Các nhà phát triển trong doanh nghiệp làm việc đặc trưng trong lĩnh vực IT hay các ban kinh kinh doanh của doanh nghiệp. Họ có trách nhiệm phát triển các ứng dụng nâng cao năng suất kinh doanh. Các yêu cầu kinh doanh đến trực tiếp từ các nhóm kinh doanh bên trong doanh nghiệp. Các nhà phát triển trong doanh nghiệp có thể tạo ra các dịch vụ “đám mây” trên Windows Azure platform.
* *Nhà phát triển ISV:* Independent software vender (ISV) phát triển các giải pháp kinh doanh trên nền tảng hiện có. ISV tập trunng nhiều hơn trong thị trường hẹp (vertical market) như tài chính, sản xuất, dầu hỏa và chăm sóc sức khỏe. ISV có thể tận dụng nền tảng Windows Azure để triển khai các dịch vụ mà mục tiêu được nhắm đến là các khách hàng trong thị trường hẹp. Họ có thể thiết kế các kiến trúc trên nền tảng Windows Azure platform, đặt biệt dành cho nền công nghiệp hẹp mà họ nhắm tới.
* *Nhà phát triển tiêu dùng (Consumer developer):* các nhà phát triển tiêu dùng làm việc cho các công ty dịch vụ trực tuyến giống như MSN Live, Yahoo, Apple, Google, Facebook, MySpace và họ cung cấp các dịch vụ mail, collaboration, mạng xã hội và dịch vụ di động trực tiếp đến các nhà tiêu dùng. Các nhà phát triển tiêu dùng có thể xây dựng các dịch vụ tiêu dùng trên Windows Azure platform và cung cấp dịch vụ này đến người tiêu dùng trên khắp thế giới.

## 2.2. Các kiến thức cần thiết cho Developer

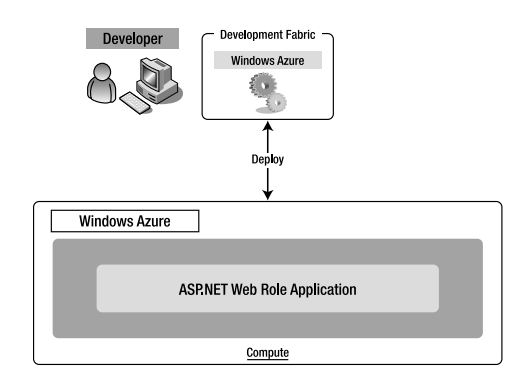
Việc hiểu rõ các công nghệ sau đây và những nguyên lý là kỹ năng chính cần thiết cho phát triển ứng dụng với Windows Azure:

* Visual Studio .NET 2008/2010
* .NET Framework 3.5
* Windows Communication Foundation
* ADO.NET
* ADO.NET Data Services
* Web services (REST, SOAP)
* XML
* ASP.NET
* .NET security
* SQL Server database development

# 3. Khái quát về sự phát triển trên Windows Azure platform

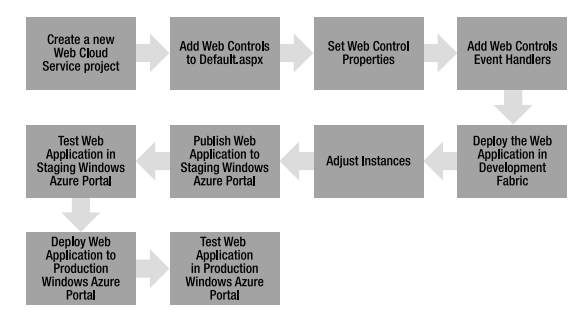
## 3.1. Kiến trúc dịch vụ

Kiến trúc dịch vụ bao gồm một dịch vụ “đám mây” web role Windows Azure được phát triển trong môi trường cục bộ bởi developer và sau đó được tải lên portal của Windows Azure. Hình sau cho thấy các developer phát triển dịch vụ “đám mây” on-premise và sau đó triển khai lên nền tảng đám mây Windows Azure.



## 3.2. Workflow cho developer

Hình sau minh họa luồng làm việc của developer cho việc phát triển dịch cụ đám mây web role Windows Azure.



Luồng làm việc trên minh họa các bước cần thiết cho phát triển, xây dựng và triển khai một dịch vụ web đơn giản trên “đám mây”.