



# DEV1 – How to Design, Build, Deploy and Manage API's in Microsoft Azure

Presented by Nikolai Blackie  
Adaptiv Integration  
3<sup>rd</sup> of October 2015

*adaptiv*

# Why API's?



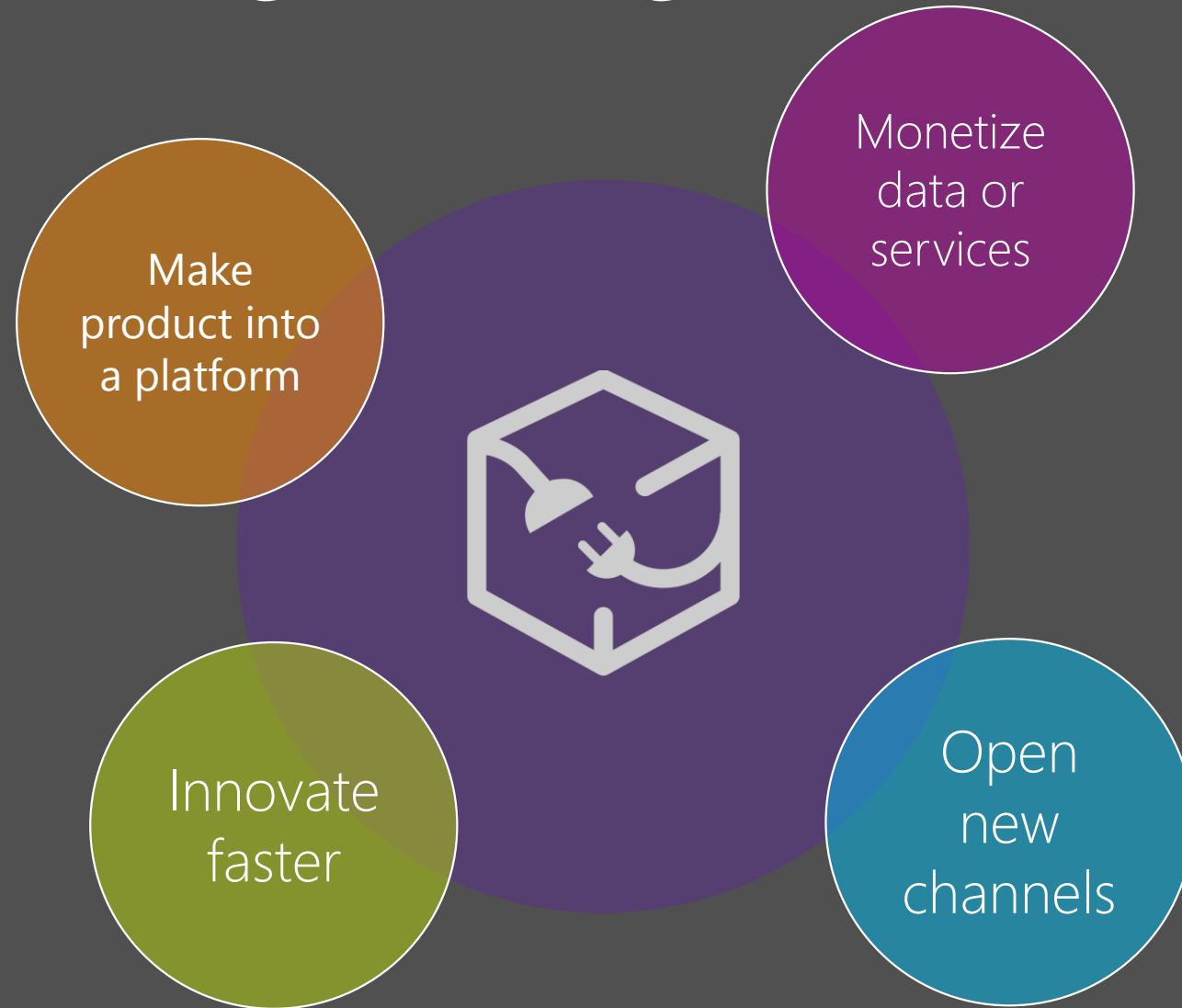
## Code Camp Auckland 2015 Demo Scenario

- Adding an API
- Leveraging Swagger
- Continuous Integration
- Composition via Logic Apps
- Expose via API Management

What Microsoft Azure  
API tools & components  
you can use?



# APIs are the engines of growth



# Common scenarios

## Mobile enablement



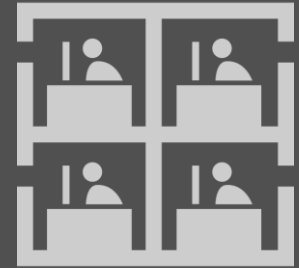
- Optimize and secure backend
- Authorize and throttle requests
- Get visibility into usage and health
- Drive developer adoption

## Partner ecosystem



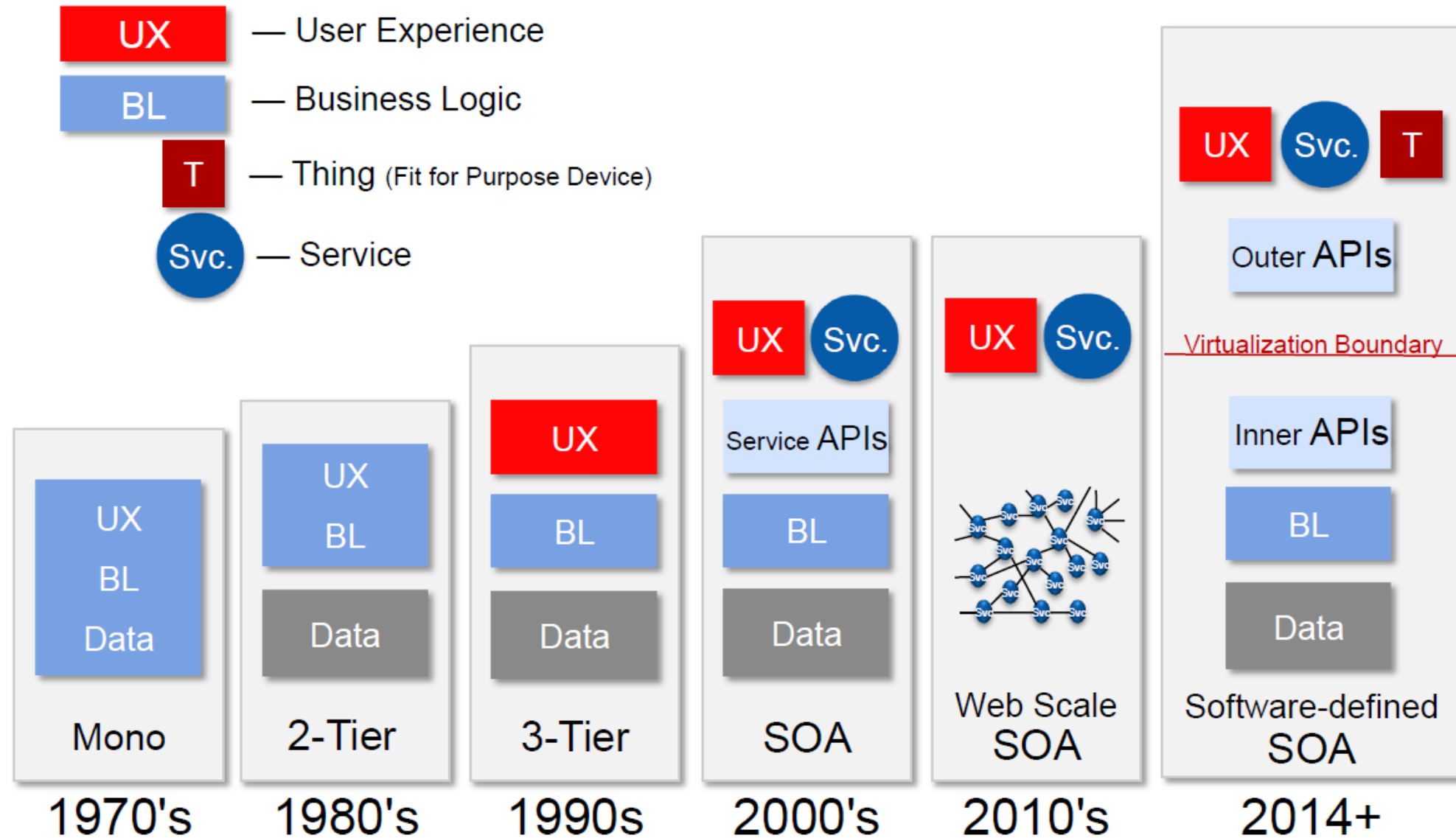
- Modernize legacy backends
- Package APIs into tiers
- Enforce usage limits
- Provide support and receive feedback

## Business agility



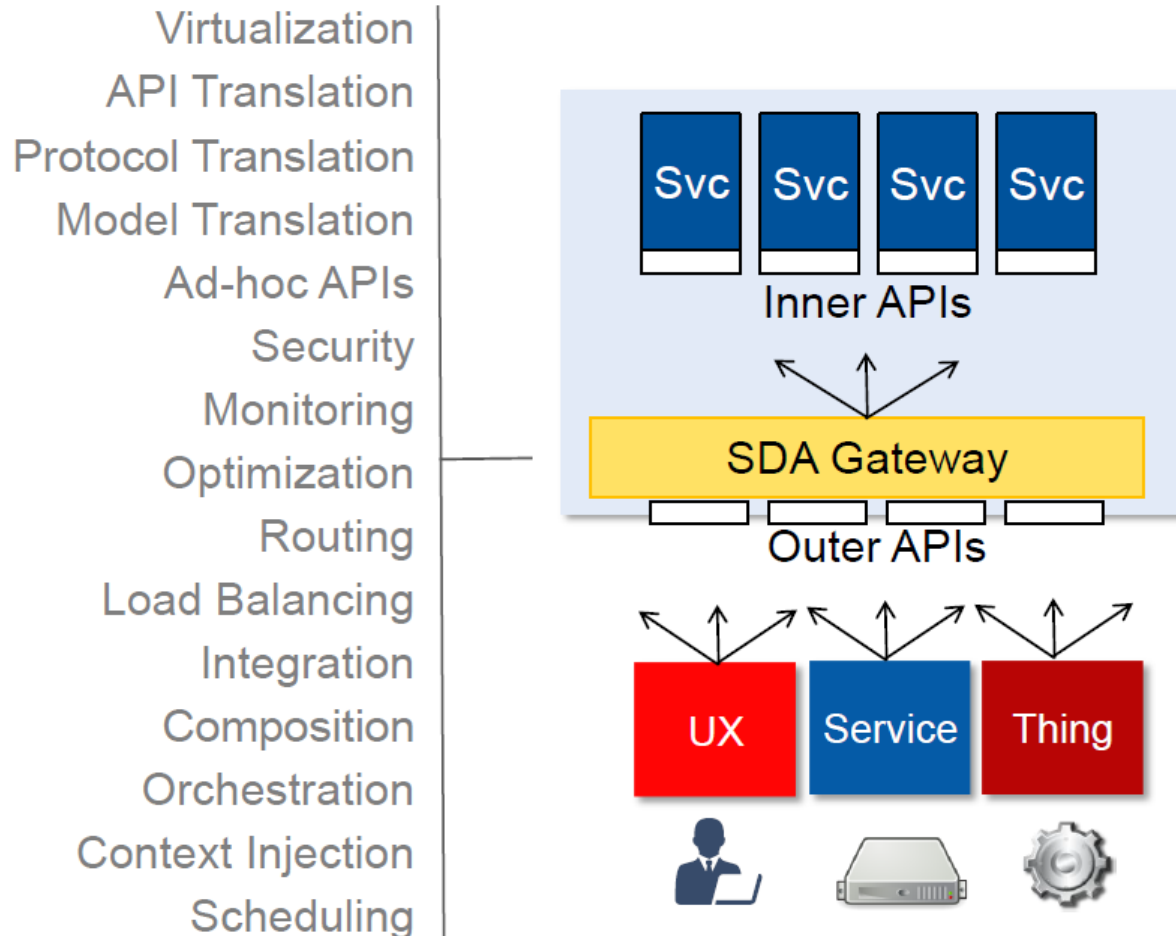
- Normalize legacy backends
- Build the catalog of APIs
- Accept organizational ID
- Respect org structure

# Software-defined Architecture for Applications



# Software-defined Architecture for Applications

## – Microsoft Technologies



API Management  
App Services  
API Apps  
Logic Apps  
Azure Service Bus  
WebAPI & WCF

# Integration Platform as a Service

## Azure App Services



=



### WEB APPS

Web apps that scale with your business



### MOBILE APPS

Build Mobile apps for any device



### LOGIC APPS

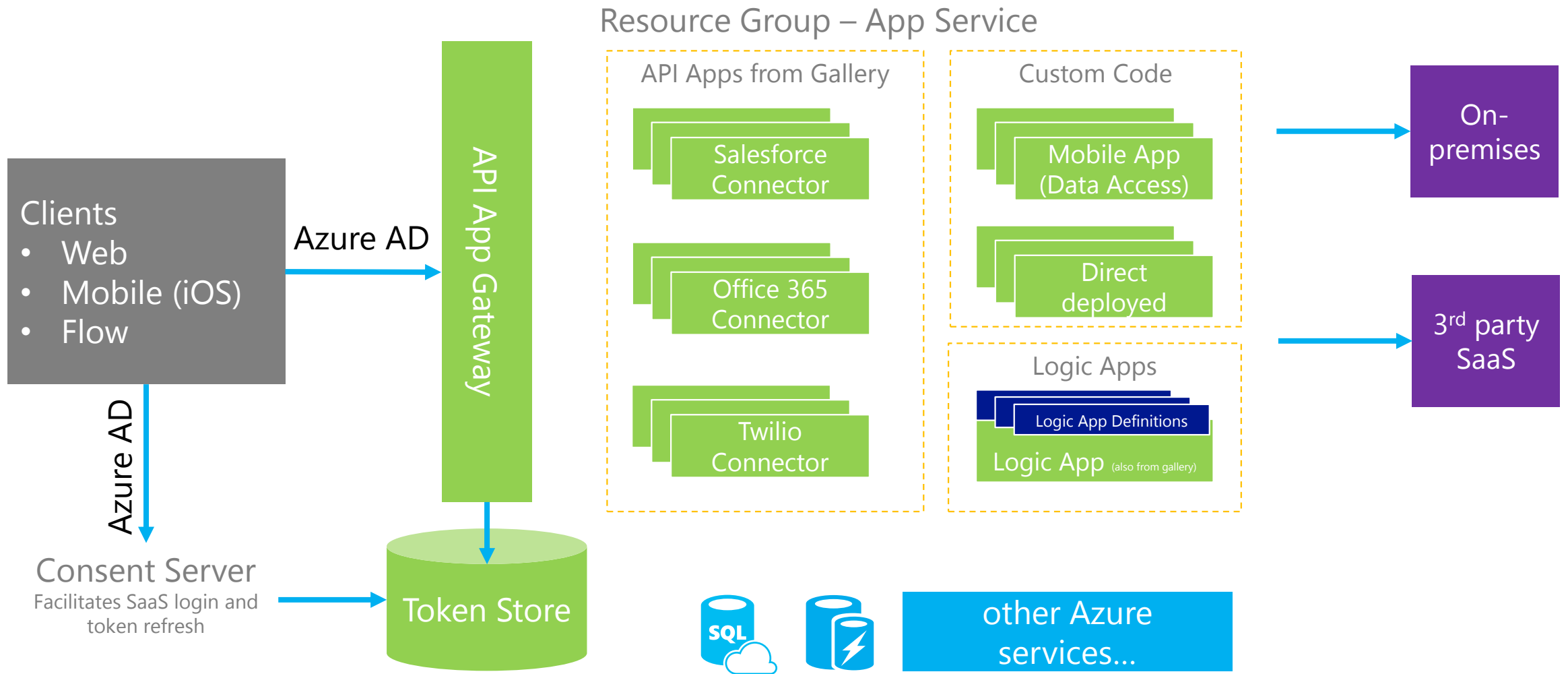
Automate business process across SaaS and on-premises



### API APPS

Easily build and consume APIs in the cloud

# API Apps Architecture Scenario





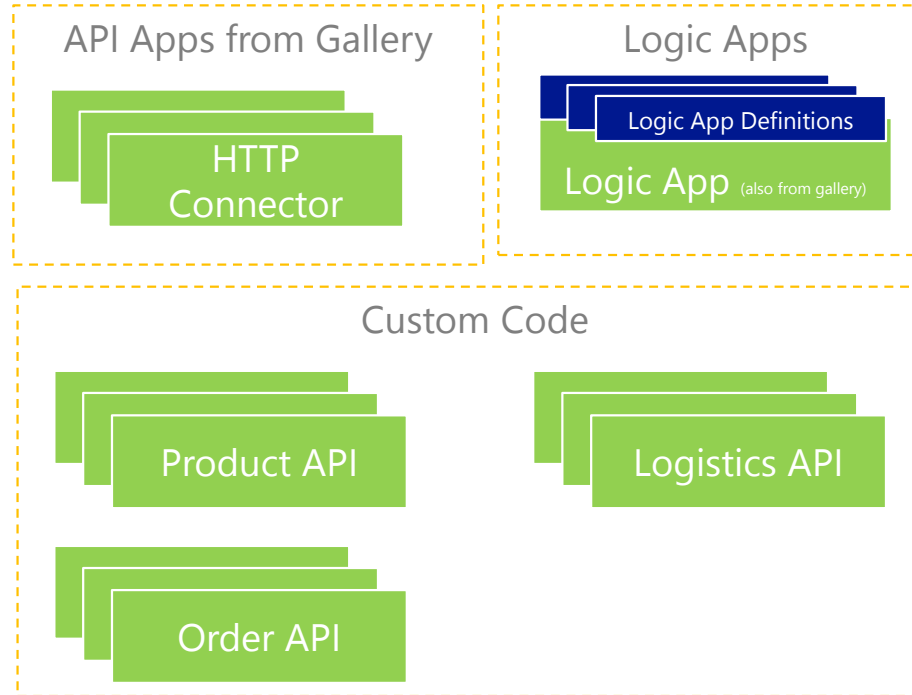
# API Apps Code Camp Demo Scenario

## Clients

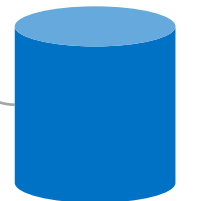
- Web
- Mobile (iOS)
- Flow

API App Gateway

## Resource Group – App Service



App Insights



Source Control

Azure API Apps provide a rich platform and ecosystem for building, consuming, and distributing APIs in the cloud and on-premises.

# Built-in API Connectors



## Connectors

- Box
- Chatter
- Delay
- Dropbox
- Azure HD Insight
- Marketo
- Azure Media Services
- OneDrive
- SharePoint
- SQL Server
- Office 365
- Oracle
- QuickBooks
- SalesForce
- Sugar CRM
- SAP
- Azure Service Bus
- Azure Storage
- Timer / Recurrence
- Twilio
- Twitter
- IBM DB2
- Informix
- Websphere MQ
- Azure Web Jobs
- Yammer
- Dynamics CRM
- Dynamics AX
- Hybrid Connectivity

## Protocols

- HTTP, HTTPS
- File
- Flat File
- FTP, SFTP
- POP3/IMAP
- SMTP
- SOAP + WCF

## BizTalk Services

- Batching / Debatching
- Validate
- Extract (XPath)
- Transform (+Mapper)
- Convert (XML-JSON)
- Convert (XML-FF)
- X12
- EDIFACT
- AS2
- TPMOM
- Rules Engine



# API App Settings for Custom Applications

The screenshot displays the Azure portal interface for an API App named 'AdtvDemo.WUS.API.Products'. The left pane shows the 'Essentials' section with various settings. Three specific settings are highlighted with numbered callouts:

- 1) URL: <https://microsoft-apiapp8794af517534a5...>
- 2) API app host: [AdtvDemo.WUS.API.Products](#)
- 3) Gateway: [AdtvDemo-WUS-RG04f39fb1101e4fd2b35...](#)

The right pane shows the 'API definition' section with a table of operations:

OPERATION	HTTP METHOD	LOGIC APP TRIGGER
/api/Product/ValidateProducts	POST	
/api/Product/GetProducts	GET	
/api/Product/{id}	GET	

- 1) URL for external access
- 2) Host that contains API settings
- 3) Shared Resource Group Gateway

# Demo: API Apps

How to make your own

# Swagger

# Generate Document

## AdtvDemo.WUS.API.Products

### Product

Show/Hide | List Operations | Expand Operations

GET /api/Product/GetProducts Gets all Products available

GET /api/Product/{id} Gets Product by id

### Response Class (Status 200)

Model | Model Schema

```
{
  "Id": 0,
  "Name": "string",
  "Price": 0
}
```

Response Content Type

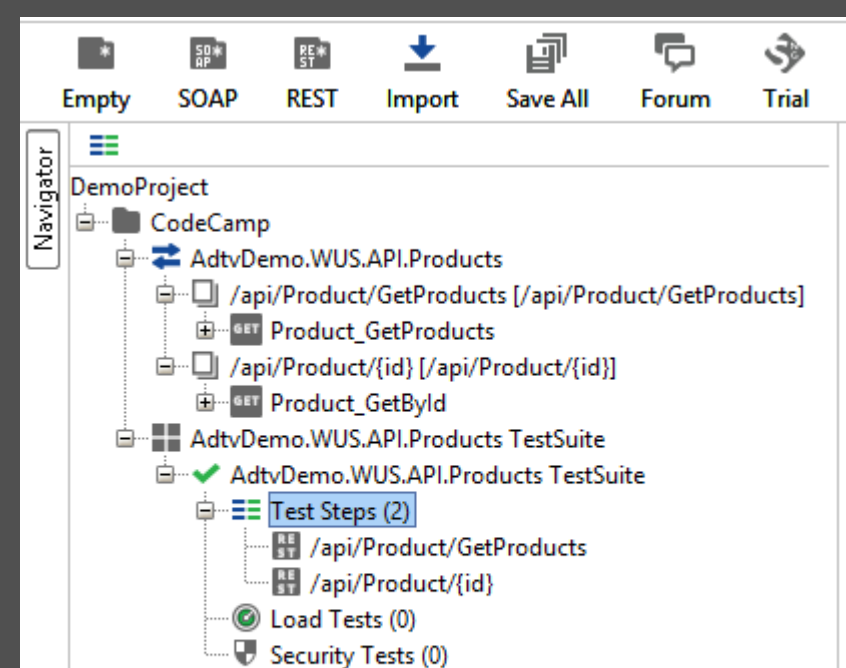
### Parameters

Parameter	Value	Description	Parameter Type
id	<input type="text" value="(required)"/>	id of product	path

### Response Messages

HTTP Status Code	Reason	Response Model
404	If product not found	
500	Internal Server Error	

Try it out!



```
{
  "swagger": "2.0",
  "info": {
    "version": "v1",
    "title": "AdtvDemo.WUS.API.Products"
  },
  "host": "localhost:52043",
  "schemes": ["http"],
  "paths": {
    "/api/Product/GetProducts": {
      "get": {
        "tags": ["Product"],
        "summary": "Gets all Products available",
        "operationId": "Product_GetProducts",
        "consumes": [],
        "produces": ["application/json",
                     "text/json",
                     "application/xml",
                     "text/xml"],

```

# Swagger Configuration

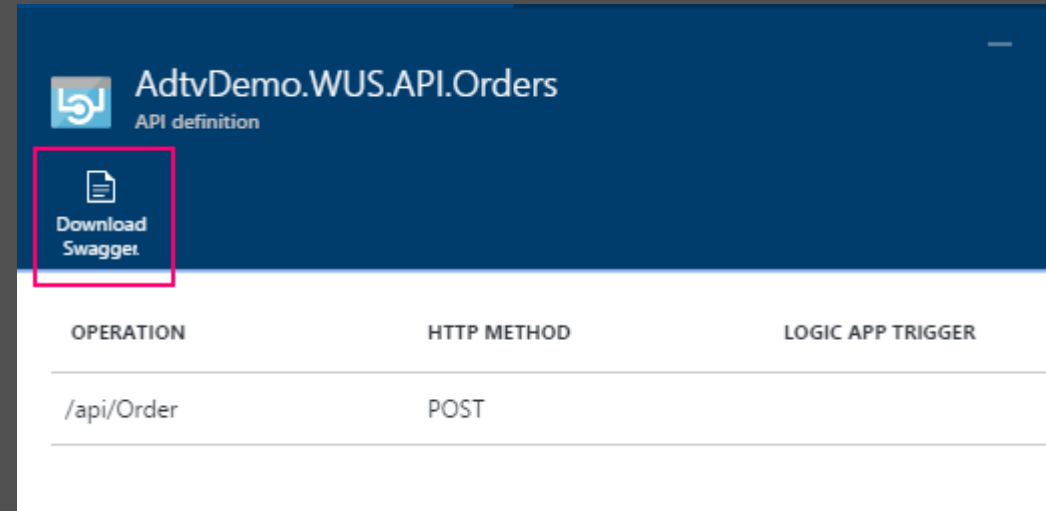
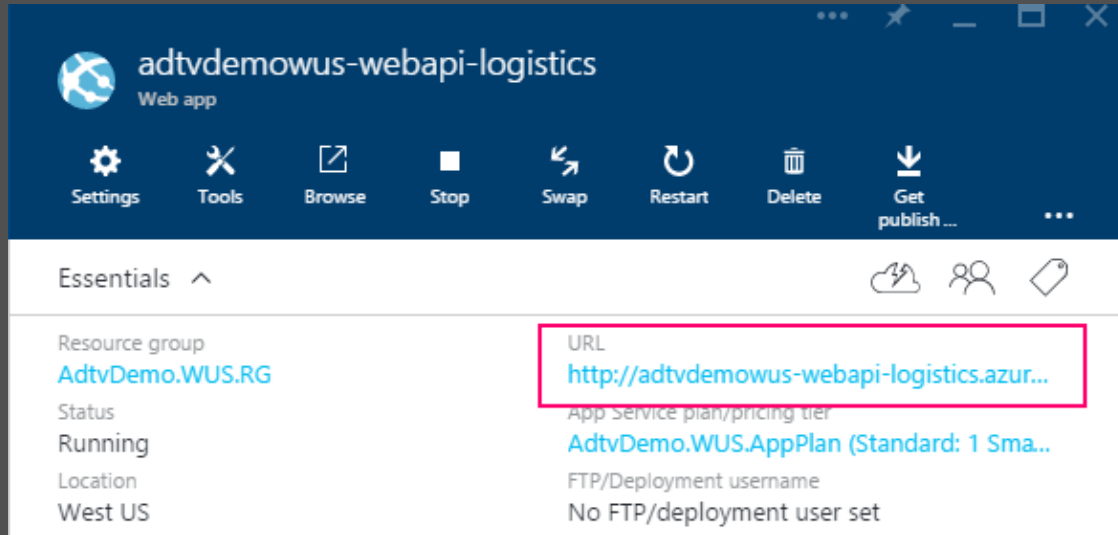
The image shows a Visual Studio interface with two panes. The left pane displays the `SwaggerConfig.cs` file with the following code:

```
//  
//c.OrderActionGroupsBy(new DescendingAlphabeticComparer());  
  
// If you annotate Controllers and API Types with  
// Xml comments (http://msdn.microsoft.com/en-us/library/b2s063f7\(v=vs.110\).aspx), you can incorporate  
// those comments into the generated docs and UI. You can enable this by providing the path to one or  
// more Xml comment files.  
//  
c.IncludeXmlComments(string.Format(@"{0}\bin\XmlComments.xml", System.AppDomain.CurrentDomain.BaseDirectory));
```

The right pane shows the Build configuration for 'All Configurations' on 'Any CPU' platform. The 'Configuration' dropdown is set to 'All Configurations' and the 'Platform' dropdown is set to 'Active (Any CPU)'. The 'Platform target' is 'Any CPU'. The 'Optimize code' checkbox is checked. The 'Warning level' is set to 4. The 'Suppress warnings' field is empty. The 'Treat warnings as errors' section has 'None' selected. The 'Output' section has 'Output path' set to 'bin\' and 'XML documentation file' checked, with the file path 'bin\XmlComments.xml' entered.

- 1) Set XML documentation file 'bin\XmlComments.xml'
  - Note All Configurations
- 2) Update SwaggerConfig IncludeXmlComments to
  - `string.Format(@"{0}\bin\XmlComments.xml", System.AppDomain.CurrentDomain.BaseDirectory)`

# Swagger Document Access



## 1) WebAPI via web app URL

- /swagger/ui/index – UI documentation
- /swagger/docs/v1 – Swagger DOC

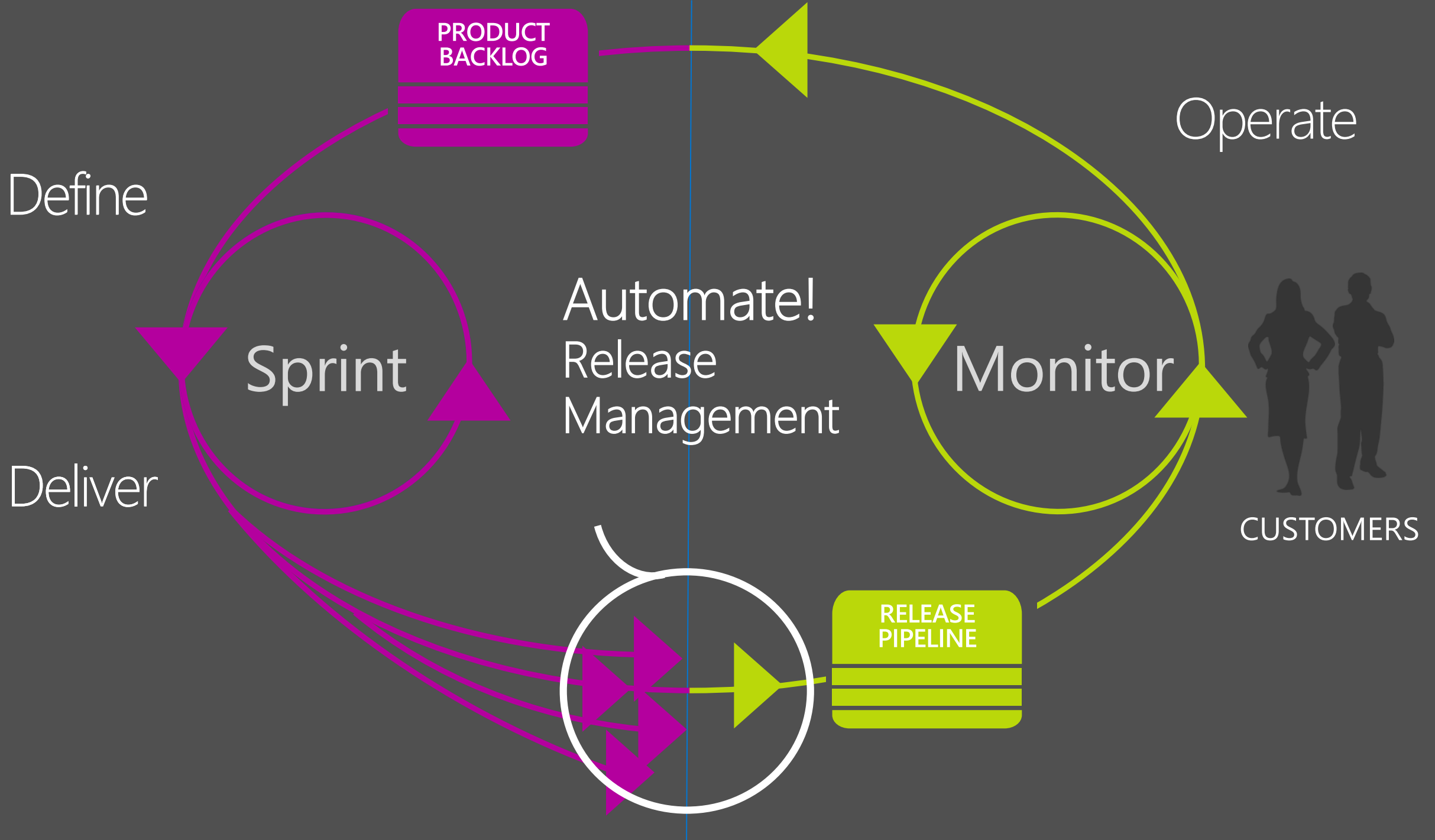
## 2) API App

- Download swagger
- API Host URL + /swagger/docs/v1 – Swagger DOC



# Demo: Swagger

Spend more time coding, less time documenting




# Visual Studio API Build Solution Configuration

Definitions / AklCodeCamp2015 Code Build | Builds


**Build** Options Repository Variables Triggers General Retention History

Save Queue build... Undo

+ Add build step...

 Visual Studio Build  
Build solution **\*\*\\*.sln**

 Visual Studio Test  
Test Assemblies **\*\*\\$(BuildConfiguration)\\*test\*.dll;-.;\*\*\obj\\*\***

 Index Sources & Publish Symbols  
Publish symbols path:

## Build solution **\*\*\\*.sln**

Solution	<b>**\*.sln</b>	...	i
MSBuild Arguments	<b>/p:DeployOnBuild=true /p:WebPublishMethod=Packag</b> i		
Platform	<b>\$(BuildPlatform)</b>		
Configuration	<b>\$(BuildConfiguration)</b>		
Clean	<input checked="" type="checkbox"/>		
Restore NuGet Packages	<input checked="" type="checkbox"/>		
Visual Studio Version	Visual Studio 2013 i		

MSBuild Arguments to support web deployment

```
/p:DeployOnBuild=true /p:WebPublishMethod=Package  
/p:PackageAsSingleFile=true /p:SkipInvalidConfigurations=true  
/p:PackageLocation="$(build.stagingDirectory)"
```

# Visual Studio API Deploy Solution Configuration

The screenshot shows the Visual Studio Build and Deploy configuration interface. On the left, a list of build steps is shown, with 'Azure Web App Deployment' selected. The main panel displays the configuration for 'Azure Deployment: adtvdemowus-webapi-logistics'.

**Build Steps:**

- Visual Studio Build  
Build solution \*\*\\*.sln
- Visual Studio Test  
Test Assemblies \*\*\\$(BuildConfiguration)\\*test\*.dll;~:\*\*\obj\\*\*
- Index Sources & Publish Symbols  
Publish symbols path:
- Publish Build Artifacts  
Publish Artifact: drop
- Azure Web App Deployment**  
Azure Deployment: adtvdemowus-webapi-logistics

**Azure Deployment: adtvdemowus-webapi-logistics**

Azure Subscription: [Dropdown] [Manage](#) ⓘ

Web App Name: adtvdemowus-webapi-logistics

Web App Location: West US ⓘ

Slot: stagingbuild ⓘ

Web Deploy Package: \$(build.stagingDirectory)\\*\*\AdtvDemo.WUS.WebAPI.L... ⓘ

Additional Arguments: [Text Box]

**Control Options**

- Enabled ☒
- Continue on error ☐
- Always run ☐

- Deploy automatically to slots or live
- Web Deploy Package (based on MSBuild Arguments)
  - \$(build.stagingDirectory)\\*\*\AdtvDemo.WUS.WebAPI.Logistics.zip
  - \$(build.stagingDirectory)\\*\*\AdtvDemo.WUS.API.Products.zip

# Demo: API CI/CD

How to automate via TFS 2015 Online



## **LOGIC APPS**

Automate SaaS and  
on-premises systems

## Logic Apps for easy automation

- No code designer for rapid creation
- Dozens of pre-built templates to get started
- Out of box support for popular SaaS and on-premises apps
- Use with custom API apps of your own
- Biztalk APIs for expert integration scenarios

# App Logic Design Surface



## Triggers and actions

AdtvDemoWUS-Logic-Fulfilment



Save



Discard



Designer



Code view



Zoom in



Zoom out



Language  
Reference



Help



HTTP Listener



← Receive HTTP Request ⓘ

Method ⓘ

Post

body ⓘ

Request ID ⓘ

Content ⓘ

...



adtvdemo.wus.api.products



← Validate Products exist ⓘ

productsToValidate ⓘ

@parse(triggers().outputs.body.Content)....

body ⓘ



adtvdemo.wus.api.orders



← This accepts and order ⓘ

OrderID ⓘ

@parse(triggers().outputs.body.Content)....

DispatchID ⓘ

@{triggers().outputs.body.RequestId}

ProductDetails ⓘ

@parse(triggers().outputs.body.Content)....

body ⓘ



HTTP



← POST ⓘ

URI ⓘ

http://adtvdemo.wus-webapi-logistics.az...

Headers ⓘ

{ "Content-Type": "application/json" }

Authentication ⓘ

Body ⓘ

{ "DispatchId": "@{triggers().outputs.body.R..." }

headers ⓘ

body ⓘ

## API Apps

Search

In AdtvDemo.WUS.RG resource group



AdtvDemo.WUS.API.Orders

AdtvDemo.WUS.API.Orders



AdtvDemo.WUS.API.Products

AdtvDemo.WUS.API.Products



HTTP Listener

Microsoft  
(AdtvDemoWUS.Logic.Fulfilment.HttpPOST)



BizTalk JSON Encoder


Microsoft













HTTP Listener

Microsoft  
(AdtvDemoWUS.Logic.Fulfilment.HTTP)

# App Logic Operational Insight

 AdtvDemoWUS-Logic-Fulfilment  
Logic app

 Settings  Run Now  Enable  Disable  Edit  Delete  Move

Essentials   

Resource group  
[AdtvDemo.WUS.RG](#)

Status  
Enabled

Location  
West US

Subscription Name  
[Microsoft Partner Network](#)

Subscription ID  
04f39fb1-101e-4fd2-b353-1916362cfe8f

Endpoint  
<https://westus.logic.azure.com:443/subscri...>

App Service plan/pricing tier  
[AdtvDemo.WUS.AppPlan \(Standard: 1 Sma...](#)

Definition  
[1 trigger, 3 actions](#)

Last run  
[2 d ago](#)


Runs today  
[0 successful, 0 failed](#)

[All settings →](#)


Summary

All runs


	START TIME	IDENTIFIER	DURATION
✓	4/10/2015 11:54 p.m.	08587576508032829951	843 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508033104296	765 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508034012883	814 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508034206890	621 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508035675599	1.05 Seconds
✓	4/10/2015 11:54 p.m.	08587576508039723203	1.12 Seconds
✓	4/10/2015 11:54 p.m.	08587576508040201788	926 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508040829463	690 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508044064543	1.1 Seconds
✓	4/10/2015 11:54 p.m.	08587576508045102542	1 Second
✓	4/10/2015 11:54 p.m.	08587576508045799146	475 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508045966639	750 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508046450650	4.68 Seconds
✓	4/10/2015 11:54 p.m.	08587576508051788011	852 Milliseconds

 All runs  
AdtvDemoWUS-Logic-Fulfilment

	START TIME	IDENTIFIER	DURATION
✓	4/10/2015 11:54 p.m.	08587576508032829951	843 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508033104296	765 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508034012883	814 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508034206890	621 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508035675599	1.05 Seconds
✓	4/10/2015 11:54 p.m.	08587576508039723203	1.12 Seconds
✓	4/10/2015 11:54 p.m.	08587576508040201788	926 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508040829463	690 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508044064543	1.1 Seconds
✓	4/10/2015 11:54 p.m.	08587576508045102542	1 Second
✓	4/10/2015 11:54 p.m.	08587576508045799146	475 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508045966639	750 Milliseconds
✓	4/10/2015 11:54 p.m.	08587576508046450650	4.68 Seconds
✓	4/10/2015 11:54 p.m.	08587576508051788011	852 Milliseconds

 Logic app run  
08587576508032829951

Cancel



400MS  
300MS  
200MS  
100MS  
0MS

11:54:42.456 11:54:42.556 11:54:42.656 11:54:42.756 11:54:42.856 11:54:42.956

**3** SUCCEEDED **-** FAILED **-** RUNNING **-** SKIPPED

ACTION	STATUS	DURATION
adtvdemo.wus.api.orders	Succeeded	98 Milliseconds
adtvdemo.wus.api.products	Succeeded	358 Milliseconds
adtvdemo.wus.webapi.logis...	Succeeded	72 Milliseconds

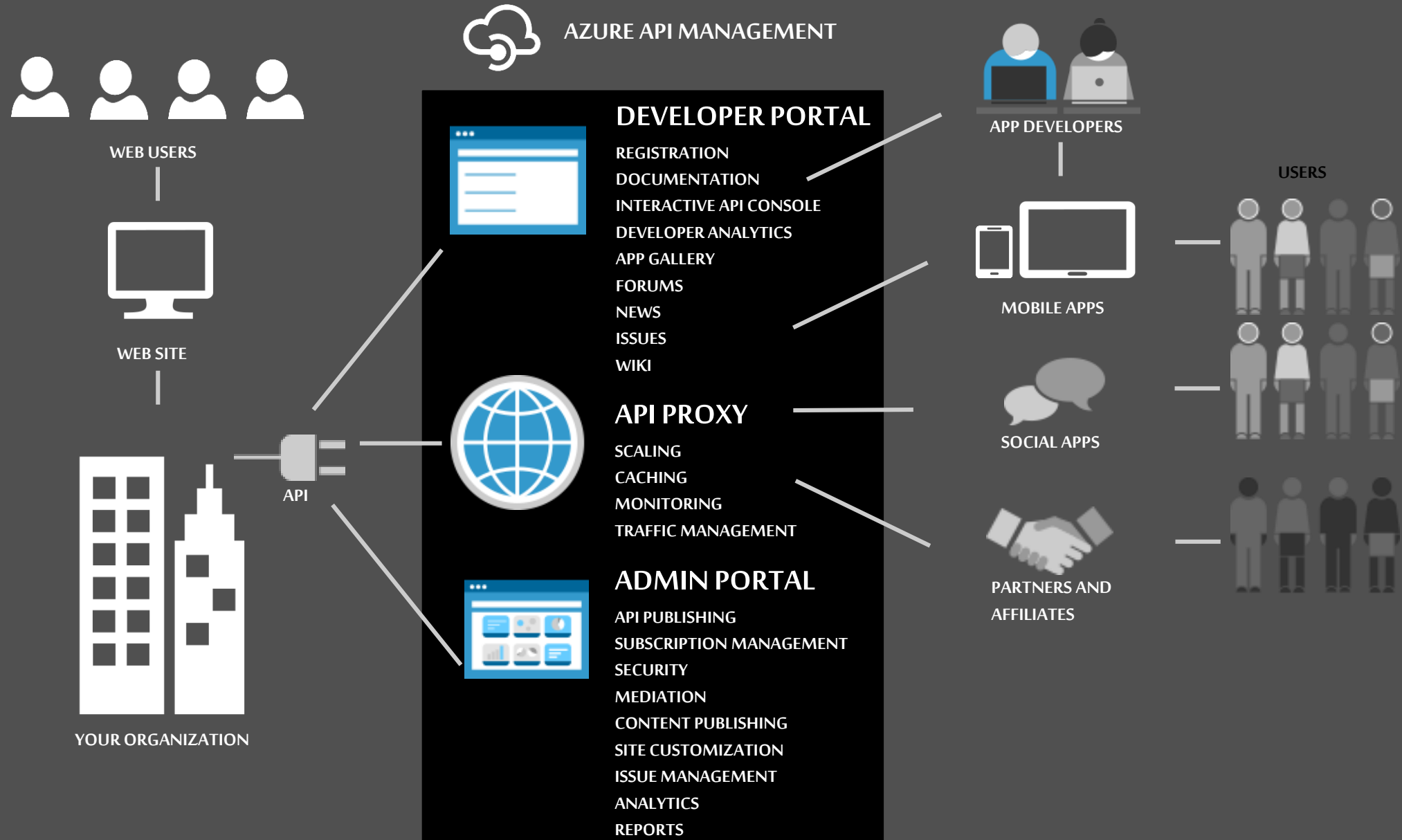
OUTPUTS LINK <https://flowprodcu01by01.blob.core.windows.net/fl...>



# Demo: Logic Apps

How to make your own

# Azure API Management



# Azure API Management – Import Swagger

The image displays two screenshots of the Azure API Management console interface.

**Left Screenshot (APIs page):**

- API MANAGEMENT** sidebar: Dashboard, APIs, Products, Policies, Analytics, Users.
- APIs** main content area:
  - Buttons: **+ ADD API** and **➔ IMPORT API** (highlighted with a red box).
  - Fulfilment** API: [http\(s\)://adtvdemowusapimanager.azure-api.net/Fulfilment](http(s)://adtvdemowusapimanager.azure-api.net/Fulfilment)
  - Logistics** API: <https://adtvdemowusapimanager.azure-api.net/logistics>

**Right Screenshot (Products page):**

- API MANAGEMENT** sidebar: Dashboard, APIs, Products, Policies, Analytics, Users, Groups, Notifications.
- Products** main content area:
  - Button: **+ ADD PRODUCT**
  - Search bar: *Search products* [Q]
  - Starter** product:
    - Description: Subscribers will be able to run 5 calls/minute up to a maximum of 100 calls/week.
    - Permissions: Administrators, Developers, Guests
    - Actions: Open, Published, DELETE
  - Unlimited** product:
    - Description: Subscribers have completely unlimited access to the API. Administrator approval is required.
    - Requires approval
    - Permissions: Administrators, Developers, Guests
    - Actions: Protected, Published, DELETE

- Import API /WebAPI App via Swagger
  - /swagger/docs/v1 – Swagger DOC
  - Or download Swagger definition
- Add API to one or more API products

# Azure API Management - Policies



- Utilise Policies to enforce security or modify API
  - Authentication
  - Format Conversion
  - Cache etc

# Demo: Expose APIs

How to make your own



Q & A

T: @nikolaiblackie

E: nikolai@adaptiv.co.nz

*adaptiv*

# Resources

Find out more

# Swagger

- [Swashbuckle](#)
- [ASP NET WebAPI documentation using Swagger](#)
- [Swashbuckle Customization](#)
- [Swagger & ASP NET WebAPI](#)



# TFS 2015 Builds

- <https://msdn.microsoft.com/Library/vs/alm/Build/overview>
- <https://github.com/Microsoft/vso-agent-tasks/blob/master/README.md>

# Logic Apps

- [Logic App EAI Tutorial](#)
- [Author Logic App definitions](#)
- [Logic App Workflow Definition Language](#)
- [Logic App Workflow Actions and Triggers](#)
- [Logic App Management API](#)
- [Logic Apps Documentation](#)

# Azure Powershell

- <https://github.com/Azure/azure-powershell/releases>

## API Apps

- [API Apps Documentation](#)

# Sample Code

- <https://github.com/nikolaiblackie/AKL2015CodeCampAppServices/blob/master/README.md>