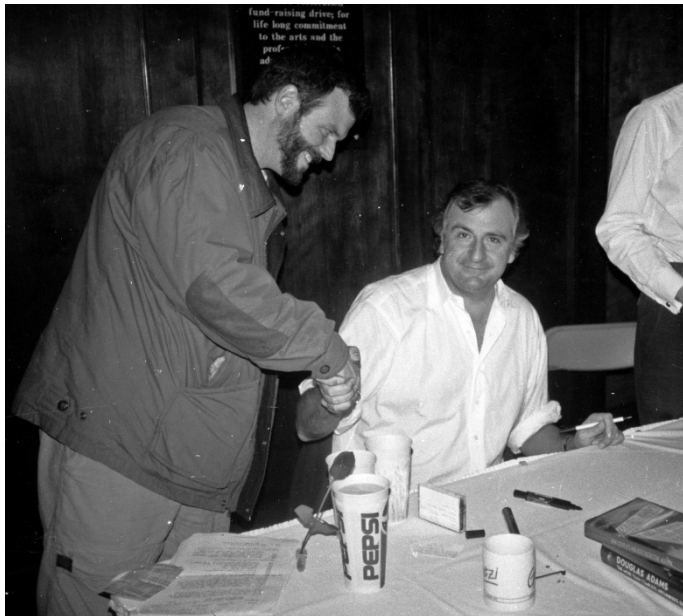


A Hitchhiker's Guide to Azure Mobile Apps



David Giard

Microsoft Technical Evangelist

- @DavidGiard
- DavidGiard.com
- TechnologyAndFriends
- dGiard@microsoft.com



This presentation is dedicated to Bill Fink



@DavidGiard

This presentation is dedicated to Bill Fink



@DavidGard

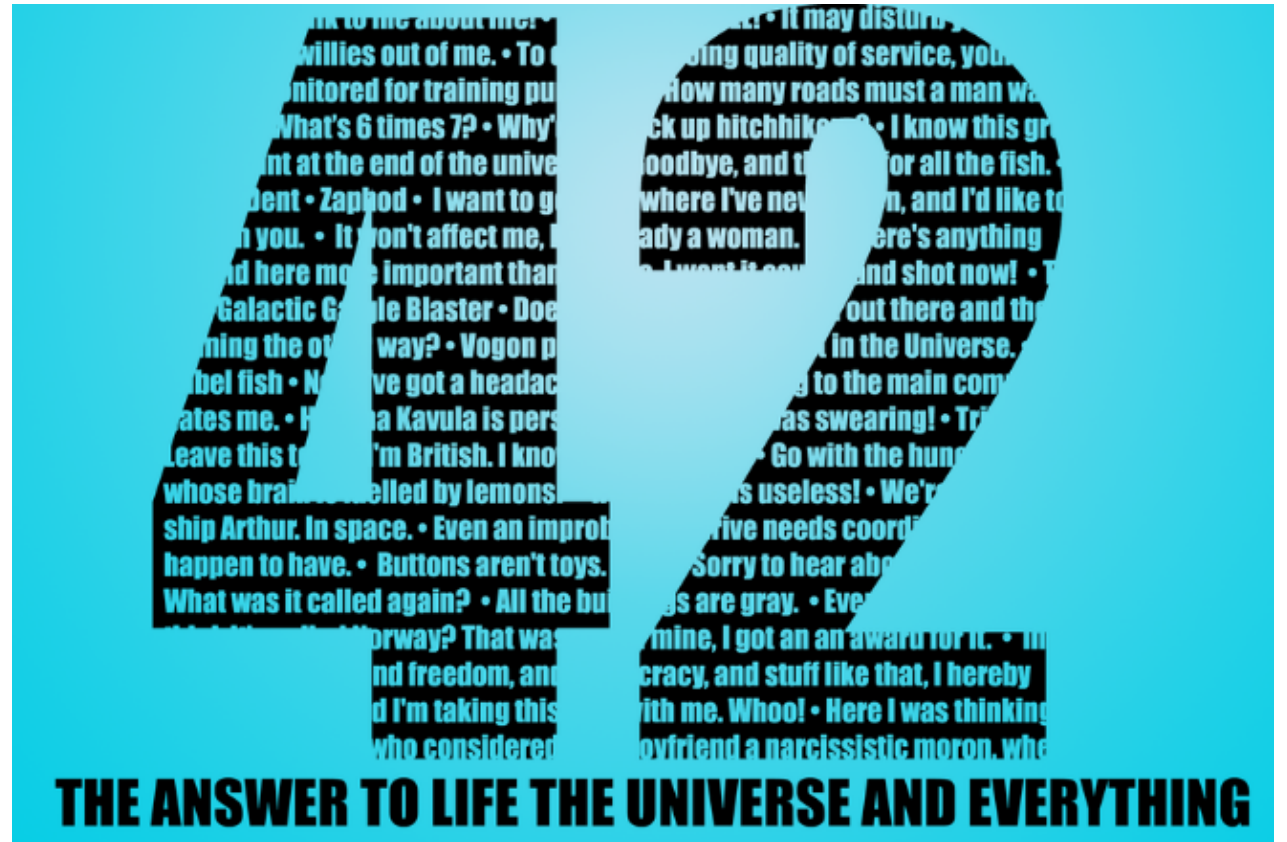
Agenda

- Why Azure Mobile Apps?
- Mobile Architecture
- Common Data Access
- Customization
- Dynamic Data
- Client Code
- Identity and Permissions
- Scaling
- API Scripts
- Notifications



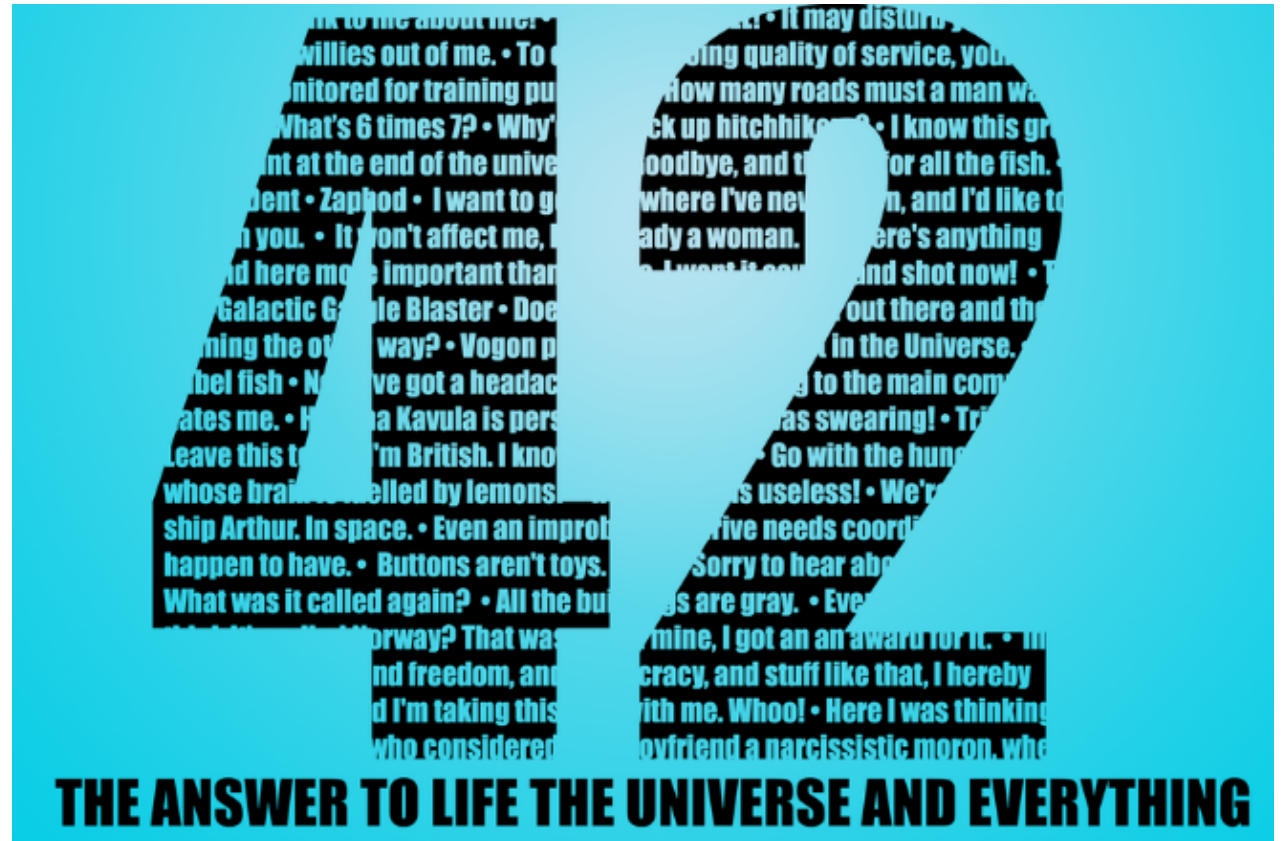
Why Azure Mobile Apps?

- Integrate with
 - Web Apps
 - Logic Apps
 - API Apps
- All features of Web Apps

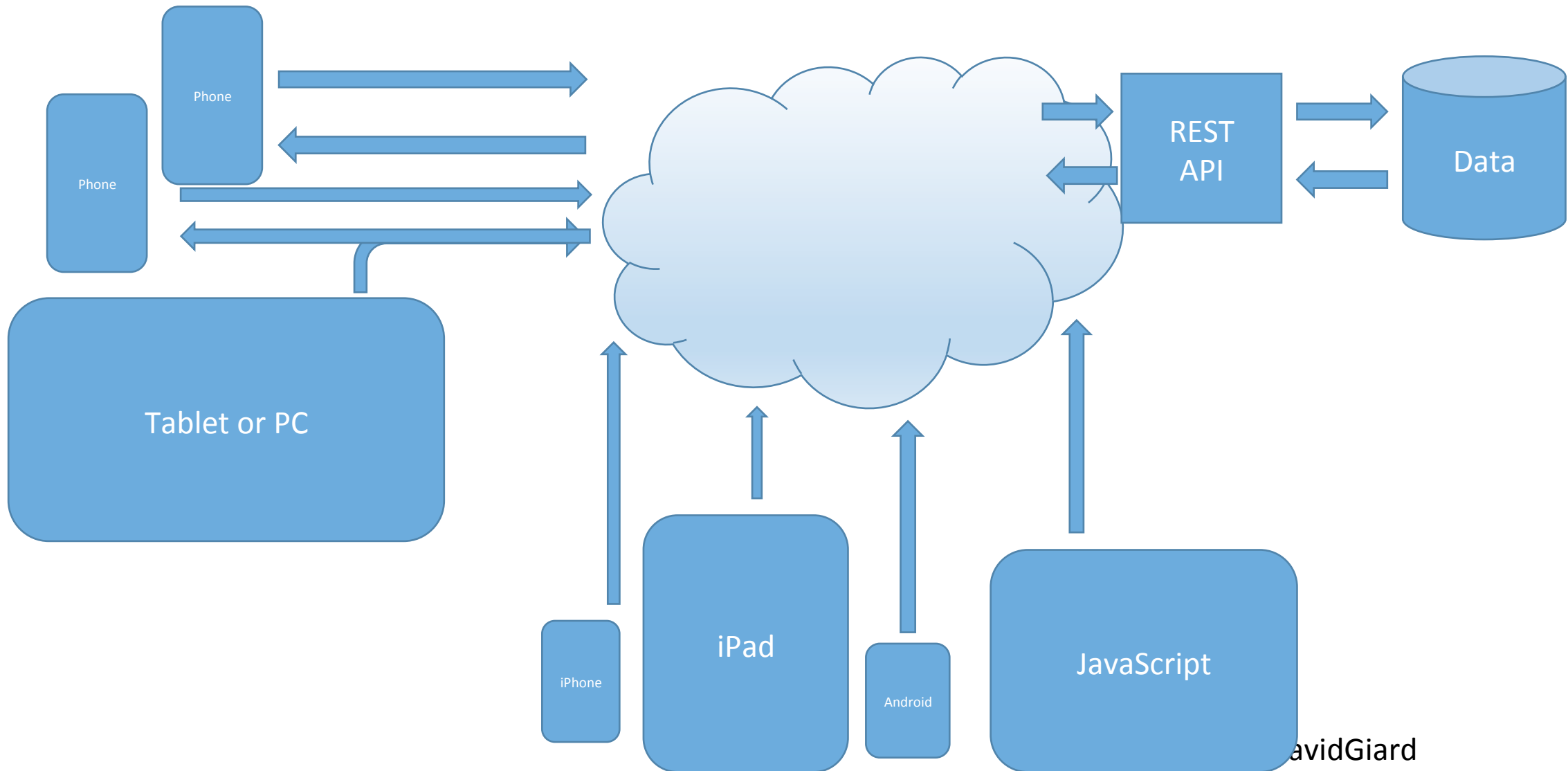


Why Azure Mobile Services?

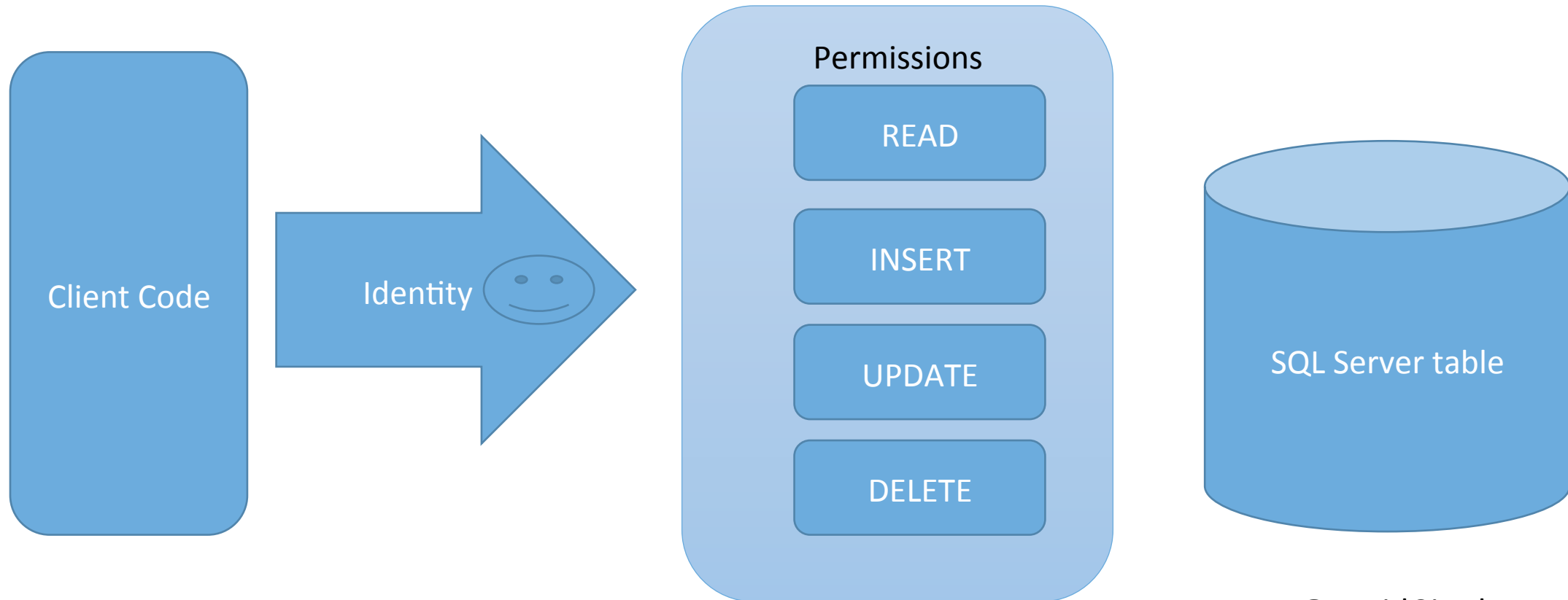
- Frees you from plumbing code
- Handles API Changes
- Cross-Platform solution
- Choose JavaScript or .NET



Mobile Architecture



Pieces of Azure Mobile Apps



Creating a Mobile App

The screenshot displays the Microsoft Azure portal interface for creating a new Mobile App. The breadcrumb navigation at the top reads: Microsoft Azure > New > Web + Mobile > Mobile App.

Left Panel (New): A search bar labeled "Search the marketplace" is at the top. Below it, a "MARKETPLACE" section includes a "See all" link. A list of categories follows: Compute, Web + Mobile (highlighted), Data + Storage, Data + Analytics, Internet of Things, Networking, Media + CDN, Hybrid Integration, Security + Identity, Developer Services, Management, and Container Apps. A "RECENT" section lists "Mobile App" (Microsoft) and "SQL Database" (Microsoft).

Web + Mobile Panel: A "FEATURED APPS" section with a "See all" link lists several options: Web App, Mobile App (highlighted), API App, Logic App (preview), App Service Environment, and API Management.

Mobile App Panel: This panel contains configuration fields for the new app:

- * App Service Name:** A text input field containing "AwesomeMobileApp" with a green checkmark. Below it, ".azurewebsites.net" is displayed.
- * Subscription:** A dropdown menu showing "Giard".
- * Resource Group:** A dropdown menu showing "Default-Web-EastUS" with a "New" link below it.
- * App Service plan/Location:** A dropdown menu showing "Default0(East US)".

At the bottom of the Mobile App panel, there is a checked checkbox for "Pin to dashboard" and a blue "Create" button.

Creating a Mobile App

The screenshot shows the Azure portal interface for a mobile app named "dgtvmobile". The top navigation bar includes icons for Settings, Tools, Browse, Stop, Swap, Restart, Delete, Get publish..., and More commanc... The main content area is divided into three sections: Essentials, Monitoring, and Settings.

Essentials

Resource group	dgtv	URL	http://dgtvmobile.azurewebsites.net
Status	Running	App Service plan/pricing tier	dgtvSvcPlan (Standard: 1 Small)
Location	North Central US	FTP/Deployment username	dgtvmobile\dgiard
Subscription name	Giard	FTP hostname	ftp://waws-prod-ch1-001.ftp.azurewebsites...
Subscription id	b5a9a948-b91f-418c-9a14-c0d81fb5e34c	FTPS hostname	https://waws-prod-ch1-001.ftp.azurewebsites...

[All settings](#) →

Monitoring

Requests and errors

100
80
60
40
20
0

11:45 PM JAN 7 12:15 AM 12:30 AM

■ HTTP SERVER ERRORS ■ REQUESTS

Settings

Search settings

SUPPORT & TROUBLESHOOTING

- Check health
- Troubleshoot
- New support request

GENERAL

- Quick start
- Properties
- Application settings

APP SERVICE PLAN

- App Service Plan
- Scale Up (App Service Plan)
- Scale Out (App Service Plan)

MOBILE

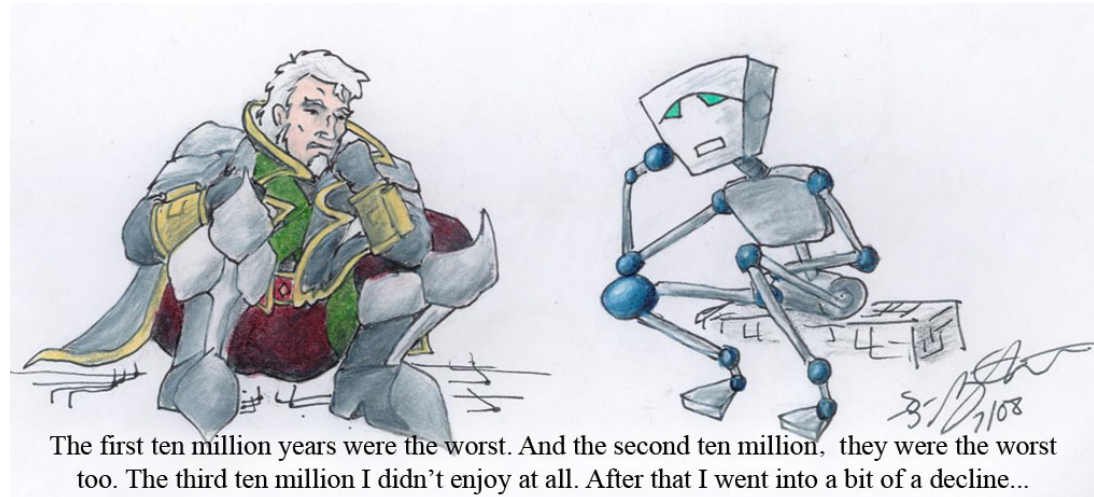
- Easy tables

REST

<https://Giard.azurewebsites.net/Tables/Table1>

<https://mymobileapp-code.azurewebsites.net/Tables/Table1>

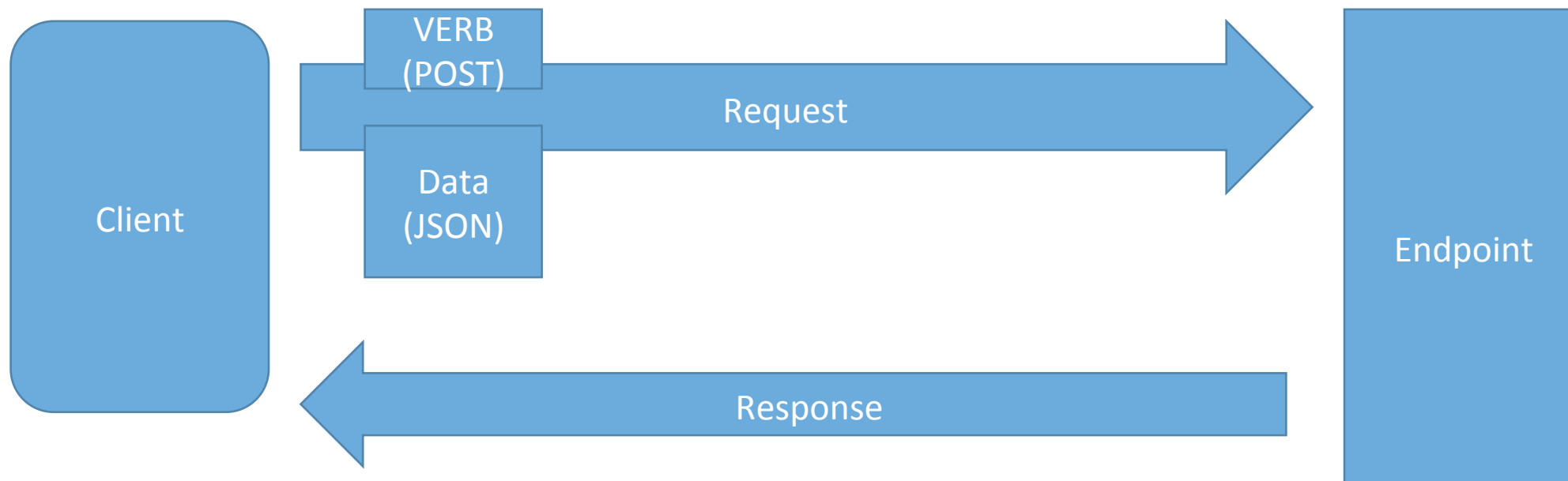
- GET, PUT, POST, DELETE, PATCH
- Extend GET with oData Query syntax



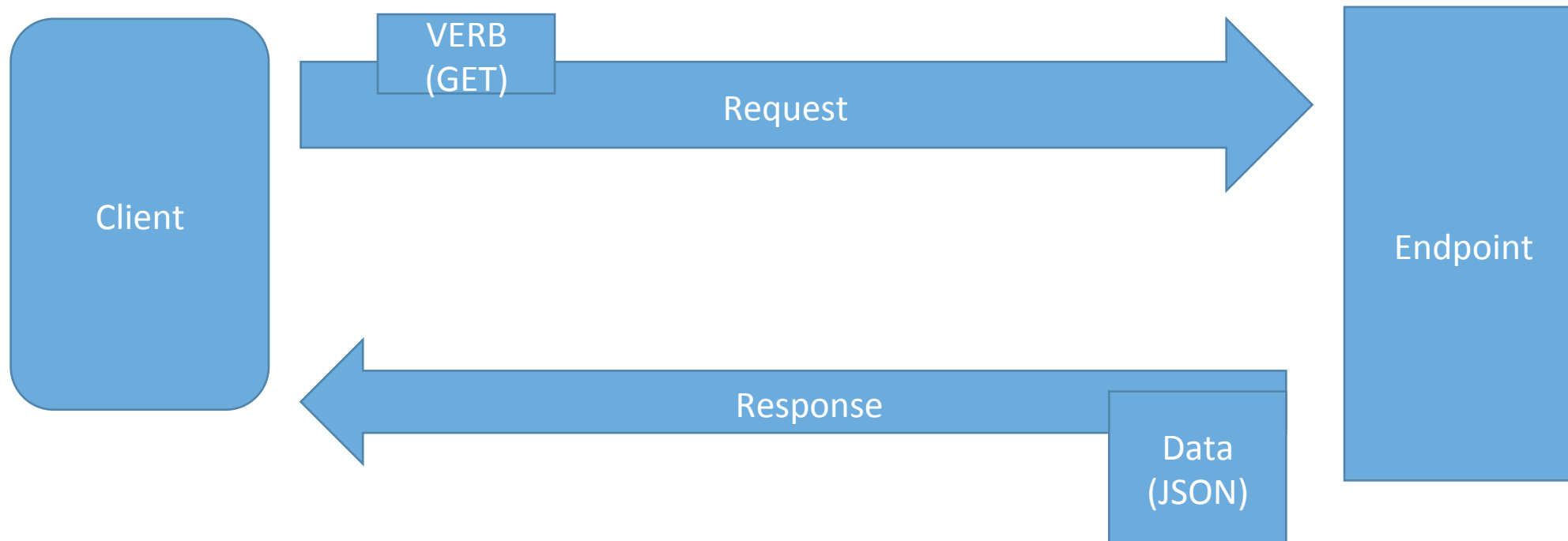
REST

HTTP Verb	Action	SQL
GET	Read Data	SELECT...
POST	Create Data	INSERT...
DELETE	Delete Data	DELETE...
PUT or PATCH	Update Data	UPDATE...

REST



REST

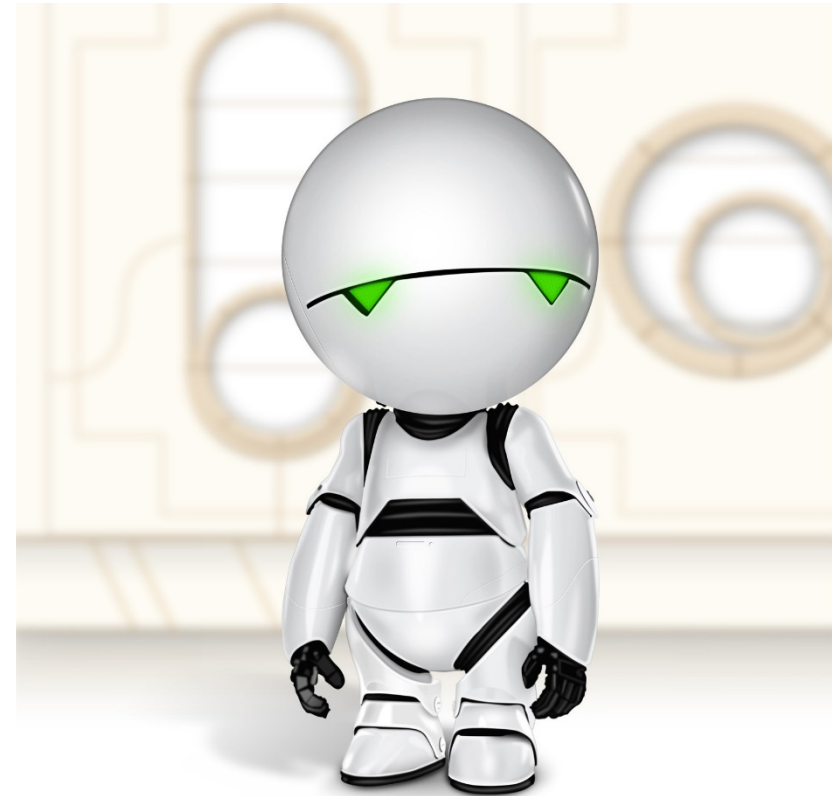


oData Query Syntax

Extension	Description
\$filter	WHERE clause
\$inlinecount	# items in table
\$orderby	SORT clause
\$select	Columns to return
\$skip	#records to skip
\$top	#records to return

../Tables/Table1?\$filter=state eq IL

../Tables/Table1?\$filter=state eq IL&\$orderby=LastName





Dynamic Schema

- Automatically adds columns if matching data submitted
- Useful during development
- Turn off during production

Permissions

Change permissions

Save Discard

Insert permission

Allow anonymous access
Authenticated access only
Disabled

Allow anonymous access

Delete permission

Allow anonymous access

Read permission

Allow anonymous access

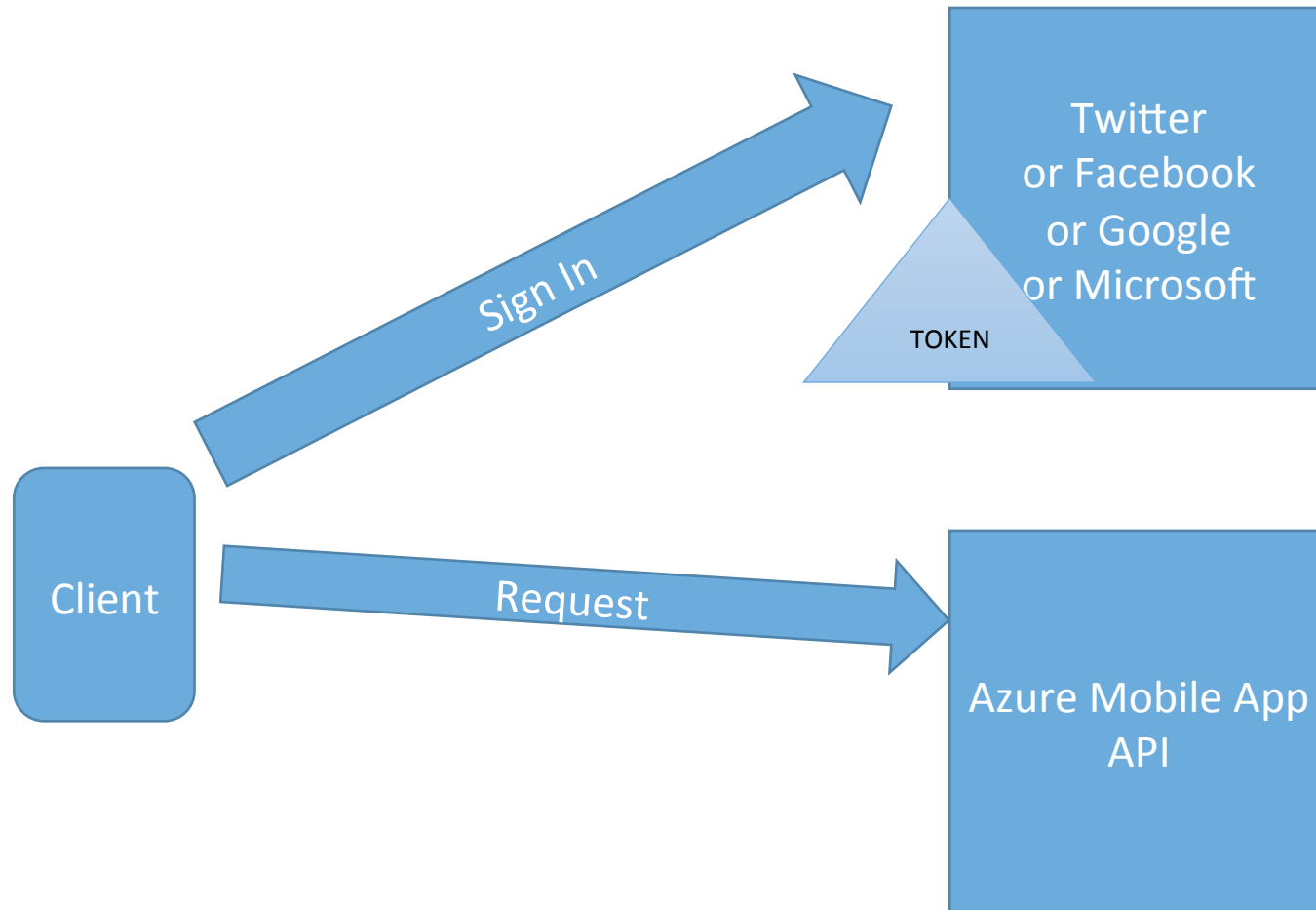
Undelete permission

Allow anonymous access

Demo



Identity – Single Sign-On



Identity

 DASHBOARD DATA API SCHEDULER PUSH **IDENTITY** CONFIGURE SCALE LOGS



Your mobile service was created.
Now let's connect it to an app.



@DavidGiard

Identity

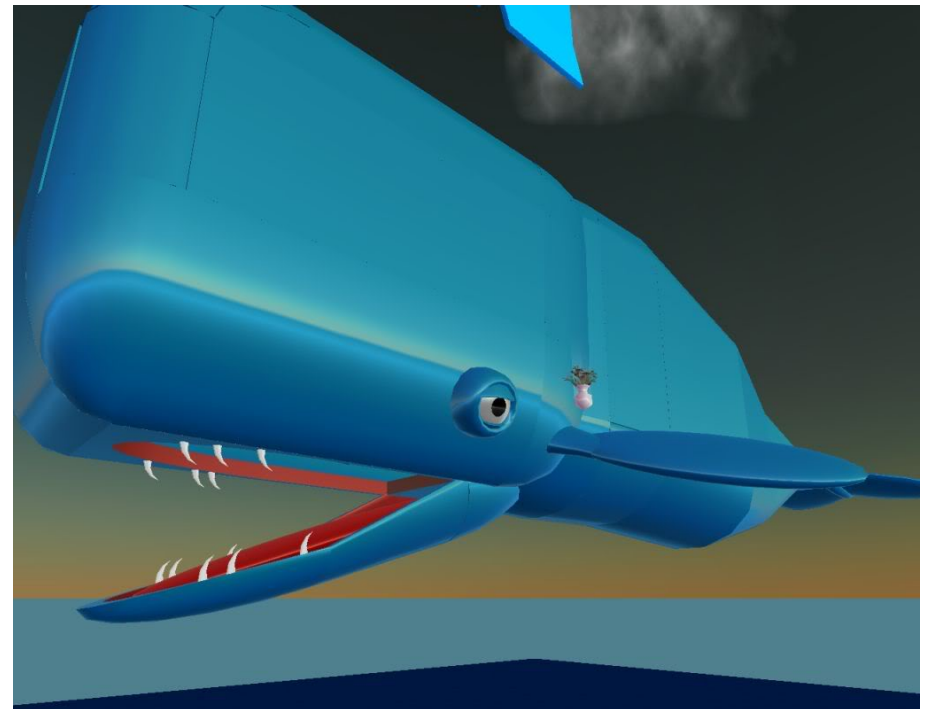
```
function insert(item, user, request) {  
  item.CreatedBy = user.userId;  
  request.execute();  
}
```

```
function read(query, user, request) {  
  query.where ({CreatedBy:user.userId});  
  request.execute();  
}
```

Demo



Scaling



Scaling

Save Discard

1

0.8

0.6

0.4

0.2

0

Sep 16 Sep 17 Sep 18 Sep 19 Sep 20 Sep 21 Sep 22

AVERAGE INSTANCES

1

* Scale by

Description Create your own set of rules. Create a schedule that adjusts your instance counts based on time and performance metrics.

Default, scale 1 - 1

CPU Percentage > 80 (increase count by 1)

Settings CPU Percentage < 60 (decrease count by 1)


Add Rule



Add Profile

Notifications for Scale Actions

Email Administrator and CoAdministrators

API Scripts

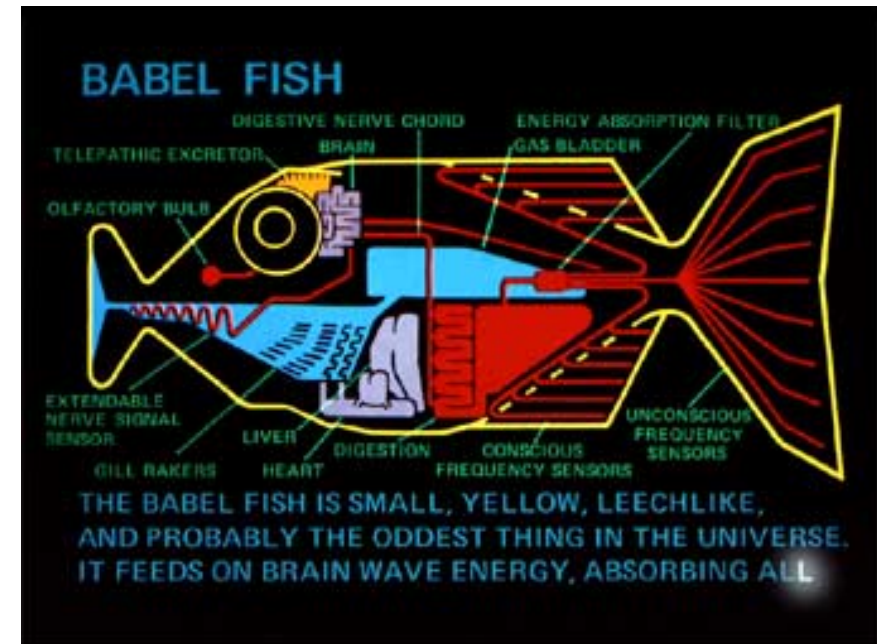
 API Definition
dgtvmobile

 Save  Discard



API definition lets you configure the location of the Swagger 2.0 metadata describing your API. This makes it easy for others to discover and consume your API. Note: the URL can be a relative or absolute path, but must be publicly accessible.

API definition location



@DavidGard

API Scripts

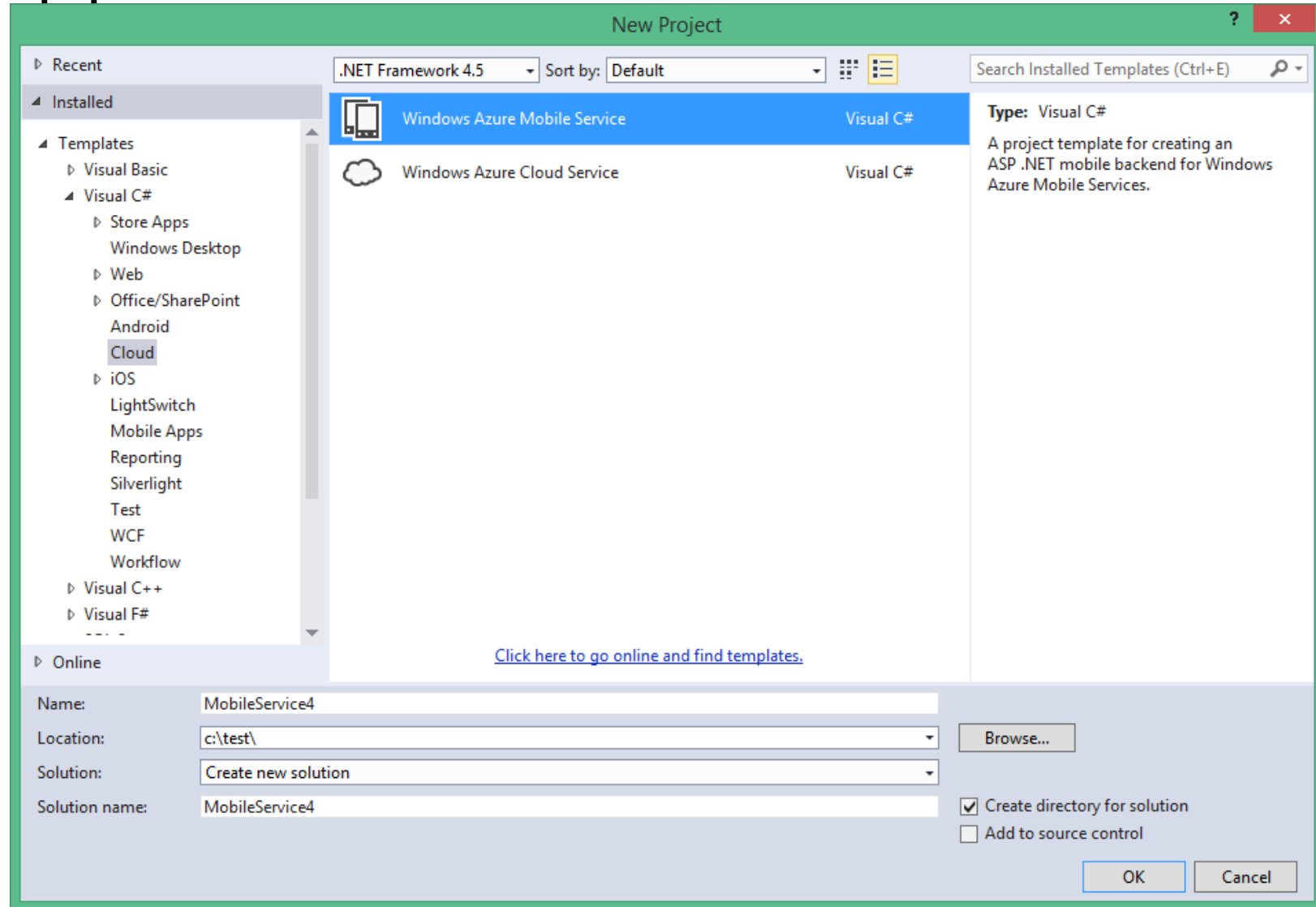
<Mobile Service URL>/api/<API Name>

e.g.,

<https://giard.azure-mobile.net/api/MyApi>

```
exports.post = function(request, response) {  
  // Use "request.service" to access features of your mobile service, e.g.:  
  // var tables = request.service.tables;  
  // var push = request.service.push;  
  
  response.send(statusCodes.OK, { message : 'Hello World!' });  
};  
  
exports.get = function(request, response) {  
  response.send(statusCodes.OK, { message : 'Hello World!' });  
};
```

Mobile Apps in .NET



Mobile Apps in .NET

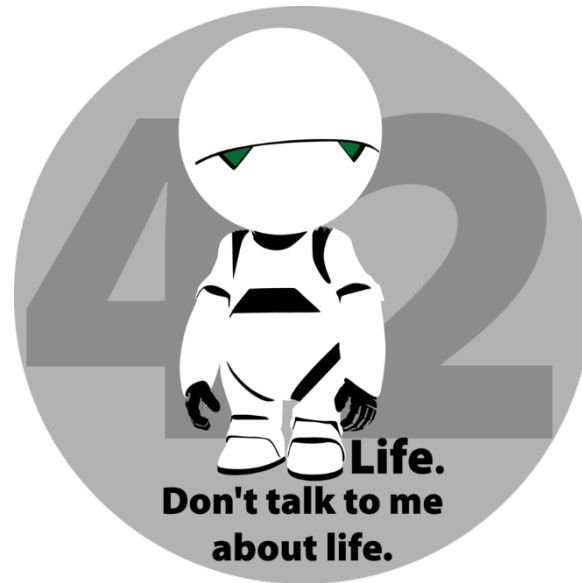
```
public class TodoItemController : TableController<TodoItem>
{
    protected override void Initialize(HttpContext controllerContext)
    {
        base.Initialize(controllerContext);
        MobileService4Context context = new MobileService4Context(Services.Settings.Schema);
        DomainManager = new EntityDomainManager<TodoItem>(context, Request, Services);
    }

    // GET tables/TodoItem
    public IQueryable<TodoItem> GetAllTodoItems()
    {
        return Query();
    }

    // GET tables/TodoItem/48D68C86-6EA6-4C25-AA33-223FC9A27959
    public SingleResult<TodoItem> GetTodoItem(string id)
    {
        return Lookup(id);
    }

    // PATCH tables/TodoItem/48D68C86-6EA6-4C25-AA33-223FC9A27959
    public Task<TodoItem> PatchTodoItem(string id, Delta<TodoItem> patch)
    {
        return UpdateAsync(id, patch);
    }
}
```

Demo



Push Notifications

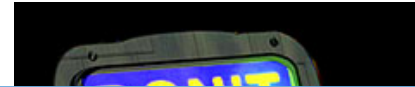
 DASHBOARD DATA API SCHEDULER **PUSH** IDENTITY CONFIGURE SCALE LOGS



Your mobile service was created.
Now let's connect it to an app.



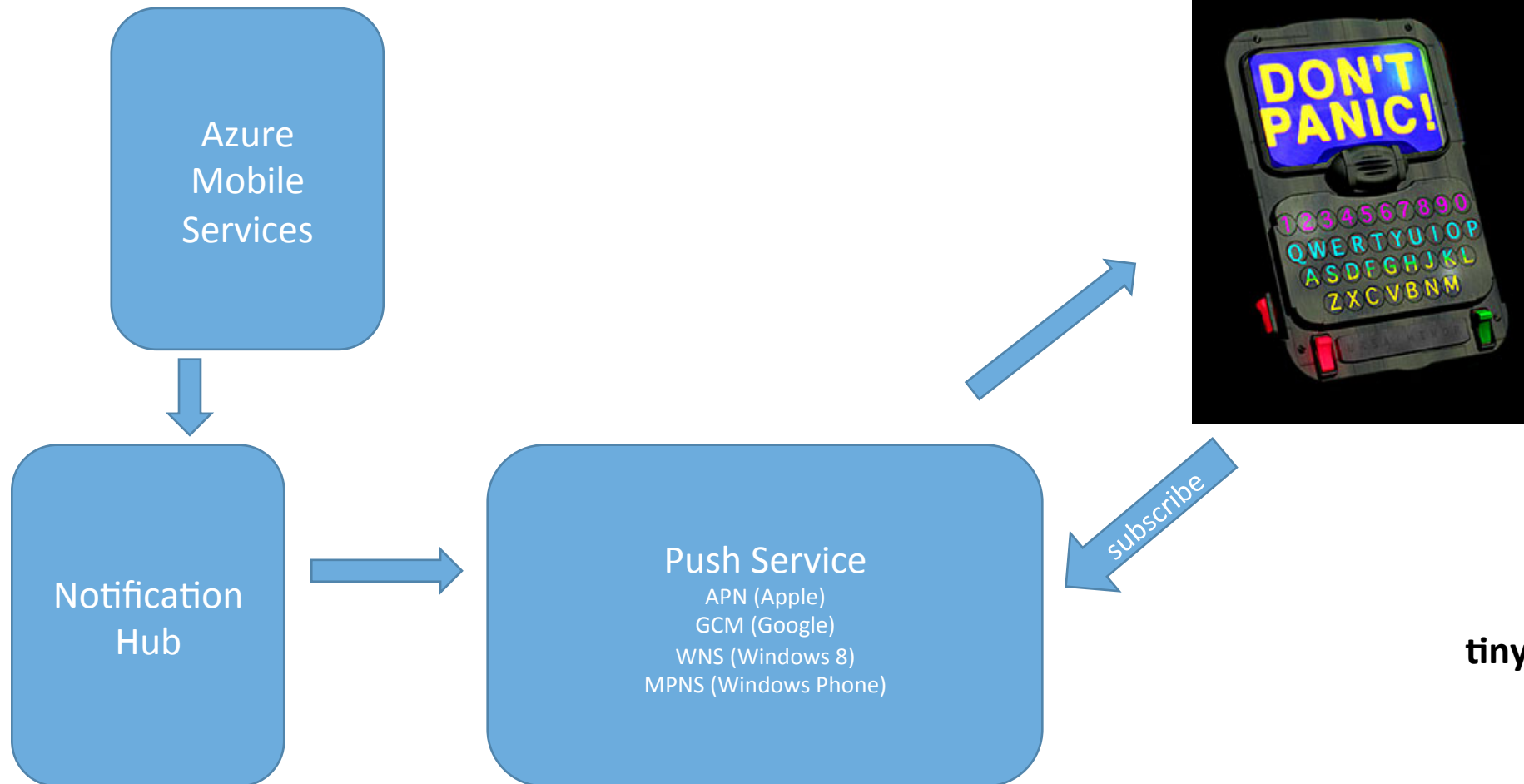
Push Notifications



```
var payload =
  '<?xml version="1.0" encoding="utf-8"?><toast><visual><binding template="ToastText01">' +
  '<text id="1">Sample Toast</text></binding></visual></toast>';
var push = request.service.push;

push.wns.send(null,
  payload,
  'wns/toast', {
    success: function (pushResponse) {
      console.log("Sent push:", pushResponse);
    }
  });
}
```

Push Notifications



tinyurl.com/PushNoti

@DavidGiard

Push Notification on Windows 8 & WP8

1. Create Mobile Service
2. Write Client App
 1. Associate app with store
 2. Get Package SID and Client ID from Live Services. Copy to Mobile Service.
 3. Register notifications channel in OnLaunched (App.xaml.cs)
 4. Enable Toast notifications (Package.appxmanifest)
3. Update service to send Push Notification

tinyurl.com/Win8Push

So Long and Thanks!

David Giard

- @DavidGiard
- DavidGiard.com
- TechnologyAndFriends.com
- channel9.msdn.com/niners/dgiard



@DavidGiard