

# Getting Started with SharePoint 2016



# Student Introductions

- Basic Info
  - What's your name?
  - Where do you work? (optional)
  - How long have you been a developer?
  - Have you used SharePoint? Which versions?
- List the skills with which you already feel comfortable
  - .NET programming in Visual Studio with C# or VB.NET
  - Development with ASP.NET and ASP.NET MVC
  - SharePoint solution development
  - JavaScript and jQuery
  - REST and OData



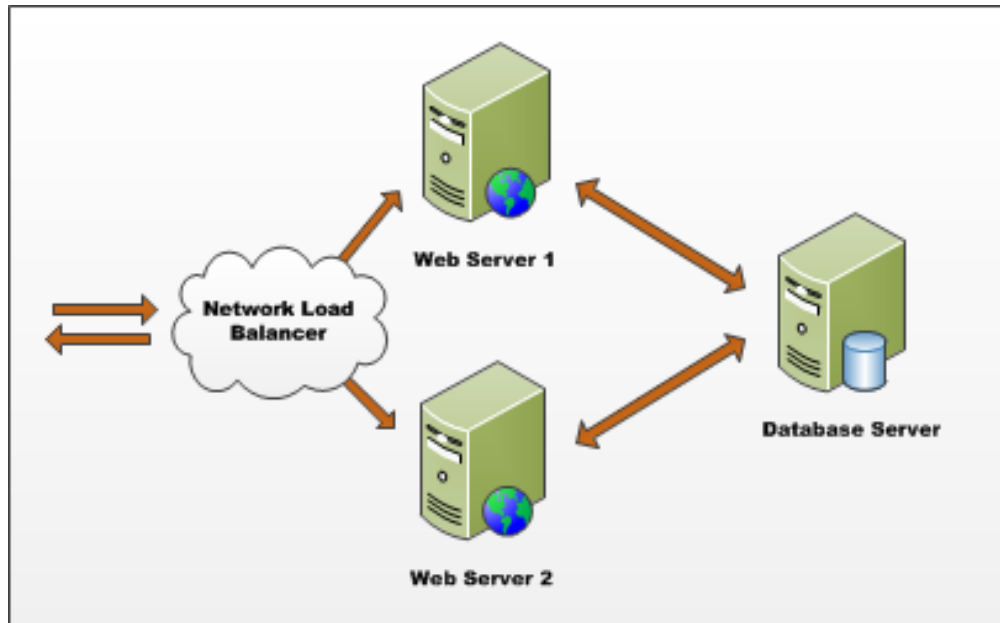
# Agenda

- SharePoint Architecture and Topology
- SharePoint Development Strategies
- SharePoint Developer Tools and Utilities
- Creating a SharePoint Development Environment



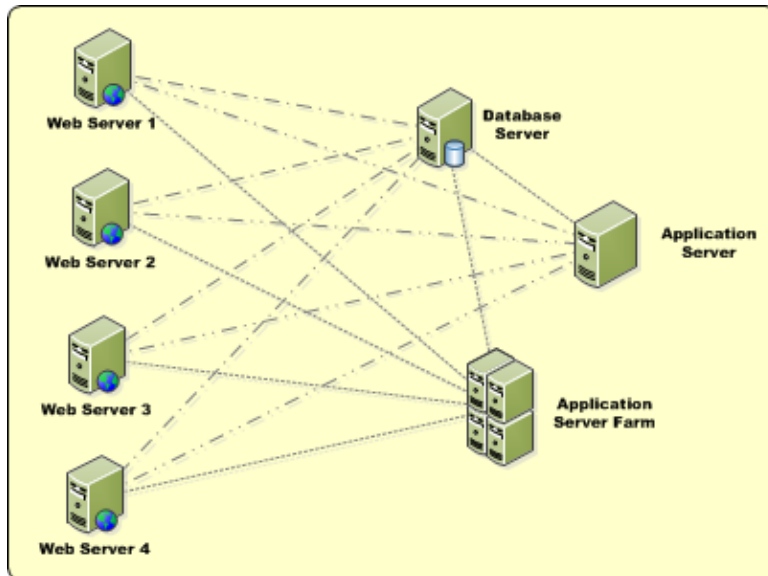
# SharePoint Farms

- SharePoint farms created for on-premises deployments
  - Farm requires Web server(s) and database server
  - Farm can be single server or multi-server
  - Each farm has exactly one configuration database
  - Single-server farm used for development environments



# Service Applications

- Services applications facilitate resource sharing
- Service apps can run on WFE or Application Servers
- Service apps can be used across farms



administrated through Central Admin

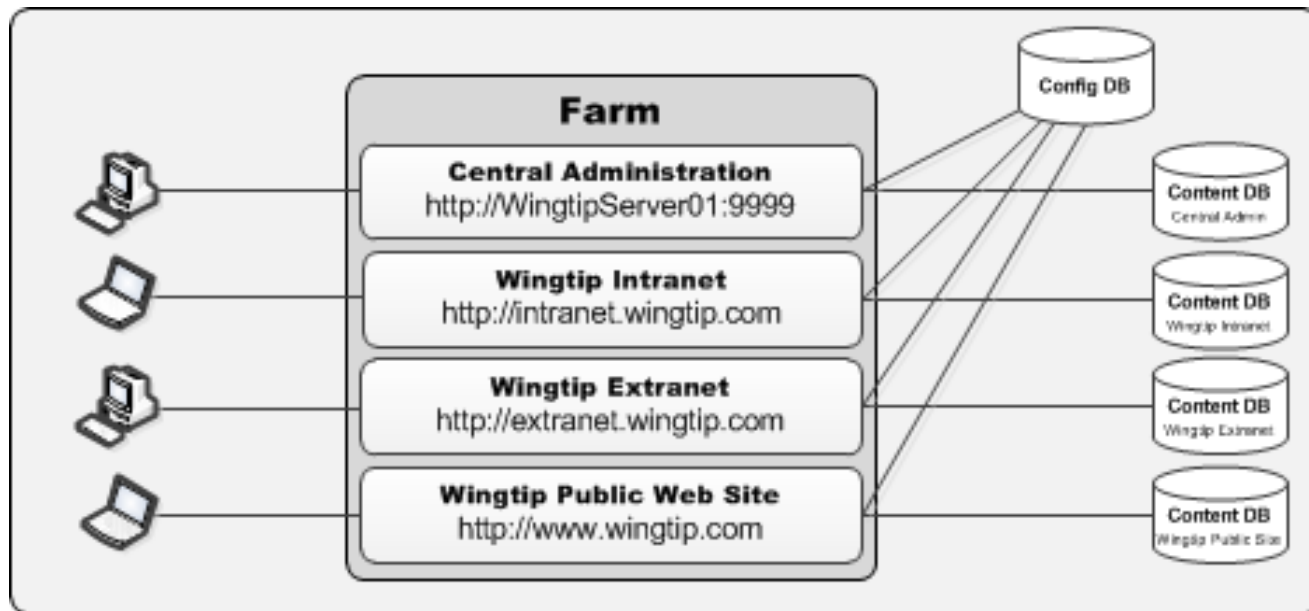
The screenshot shows the 'Manage Service Applications' page in the SharePoint Central Administration console. The page lists various service applications and their status.

Name	Type	Status
Access Services	Access Services Web Service Application	Started
Access Services	Access Services Web Service Application Proxy	Started
App Management Service Application	App Management Service Application	Started
App Management Service Application Proxy_a67adc2b-644d-4361-ac95-81067ba7b763	App Management Service Application Proxy	Started
Application Discovery and Load Balancer Service Application	Application Discovery and Load Balancer Service Application	Started
Application Discovery and Load Balancer Service Application Proxy_a67adc2b-644d-4361-ac95-81067ba7b763	Application Discovery and Load Balancer Service Application Proxy	Started
Business Data Connectivity Service	Business Data Connectivity Service Application	Started
Business Data Connectivity Service	Business Data Connectivity Service Application Proxy	Started
Excel Services Application	Excel Services Application Web Service Application	Started
Excel Services Application	Excel Services Application Web Service Application Proxy	Started
Machine Translation Service	Machine Translation Service	Started
Machine Translation Service	Machine Translation Service Proxy	Started
Managed Metadata Service	Managed Metadata Service	Started
Managed Metadata Service	Managed Metadata Service Connection	Started
PowerPoint Conversion Service Application	PowerPoint Conversion Service Application	Started
PowerPoint Conversion Service Application	PowerPoint Conversion Service Application Proxy	Started
Secure Store Service	Secure Store Service Application	Started
Secure Store Service	Secure Store Service Application Proxy	Started



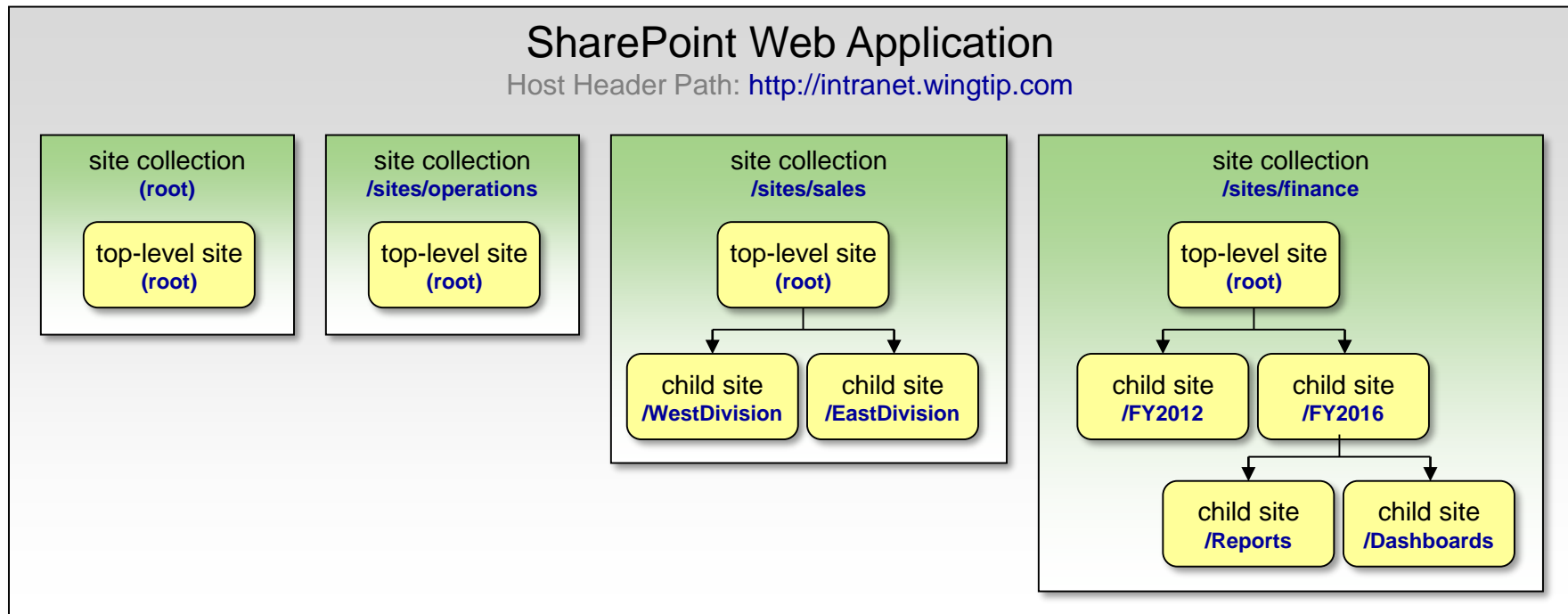
# Web Applications

- Web Applications provide HTTP entry points
  - Web Applications based on IIS Web sites
  - Web Application defines one or more URL spaces
  - Web Application security configured independently



# Site Collections and Sites

- Sites always created in scope of a site collection
  - Site collections created at web application scope in on-prem farm
  - Site collections created at tenancy scope in SharePoint on-line
  - User can be configured to be site collection administrator



# Managing SharePoint

- Central Administration
  - Available in On-Premises deployments
  - Manage servers, services, jobs, etc.
  - Create Web Applications, site collections
- Site Settings
  - Available in On-Premises & Hosted deployments
  - Manage site features, lists, users, permissions, etc.
  - Dual-purpose management site for sites & site collections
    - Manage site collection from top-level site's site settings page
    - When in a non-top-level site, only managing current site





# Managing Features

- Site collection administrator can activate features
  - Some features activate at site (aka web) level
  - Other features activate at site collection level

The image shows a sequence of three screenshots from the SharePoint interface, illustrating the path to manage features. The first screenshot on the left shows the 'Site Actions' menu with 'Manage site features' and 'Site collection features' highlighted. Arrows point from these items to the subsequent screenshots. The middle screenshot shows the 'Site Settings > Site Features' page, which lists features like 'Access App', 'BICenter Data Connections Feature', and 'Class My Site Host Content'. An arrow points from this page to the final screenshot on the right, which shows the 'Site Settings > Site Collection Features' page. This page lists features such as 'Content Deployment Source Feature', 'Content Type Syndication Hub', 'Cross-Farm Site Permissions', 'Cross-Site Collection Publishing', 'Custom Site Collection Help', 'Disposition Approval Workflow', 'Document ID Service', and 'Document Sets'. Each feature has a description and an 'Activate' button, some of which are already active.

**Site Actions**

- Manage site features
- Save site as template
- Enable search configuration export
- Reset to site definition
- Delete this site

**Site Collection Administration**

- Recycle bin
- Search Result Sources
- Search Result Types
- Search Query Rules
- Search Schema
- Search Settings
- Search Configuration Import
- Search Configuration Export
- Site collection features
- Site hierarchy
- Site collection audit settings
- Audit log reports
- Portal site connection
- Content Type Policy Templates

**Site Settings > Site Features**

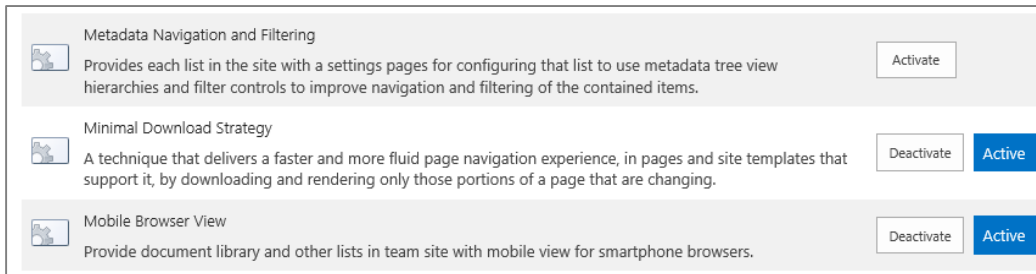
Name	Status
Access App Access web app.	
BICenter Data Connections Feature BICenter Data Connections Feature	
Class My Site Host Content Adds class and group content to S	
Class Web Types Adds required content types to th	
Community Site Feature This feature adds community func also provisions community site pa	
Content Management Interoperabi Make the data stored in this Share	
Content Organizer Create metadata based rules that	
External System Events This feature enables Alerts and Ev	
Following Content Enable users to follow documents	

**Site Settings > Site Collection Features**

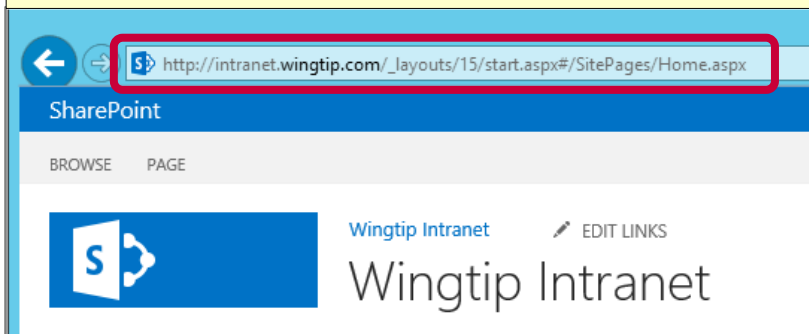
Name	Status
Content Deployment Source Feature Content Deployment Source feature enables content deployment specific checks on source site collection and enables setting up content deployment from the site collection to a target site collection.	Activate
Content Type Syndication Hub Provisions a site to be Enterprise Metadata hub site.	Deactivate Active
Cross-Farm Site Permissions Use the cross-farm site permissions feature to allow internal SharePoint applications to access websites across farms.	Activate
Cross-Site Collection Publishing Enables site collection to designate lists and document libraries as catalog sources for Cross-Site Collection Publishing.	Activate
Custom Site Collection Help Creates a Help library that can be used to store custom help for this site collection.	Activate
Disposition Approval Workflow Manages document expiration and retention by allowing participants to decide whether to retain or delete expired documents.	Deactivate Active
Document ID Service Assigns IDs to documents in the Site Collection, which can be used to retrieve items independent of their current location.	Activate
Document Sets Provides the content types required for creating and using document sets. Create a document set when you want to manage multiple documents as a single work product.	Deactivate Active

# Minimal Download Strategy (MDS) Feature

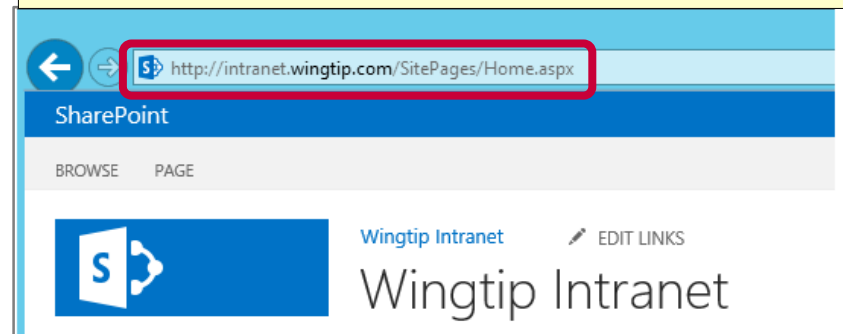
- MDS features used to smooth page transitions
  - Implemented with site-scoped feature
  - MDS features is activated in Team Site by default
  - MDS feature is disabled (and not supported) in Publishing Sites



URL structure to site page with MDS feature activated



URL structure to site page with MDS feature deactivated







**DEMO**

## **Creating a Team Site using Central Administration**

# Agenda

- ✓ SharePoint Architecture and Topology
- SharePoint Development Strategies
  - SharePoint Developer Tools and Utilities
  - Creating a SharePoint Development Environment



# SharePoint Environments

- SharePoint On-Premises Farms
  - SharePoint installed and managed by company
  - Access to 100% of SharePoint's features & capabilities
- Office 365 and SharePoint Online
  - SharePoint installed and managed by Microsoft in cloud
  - Some on-premises features not available in the cloud
- Hybrid Environments
  - Mix of the two other environments
  - Very scenario driven on customer-by-customer basis



# SharePoint Development Strategies

- Farm Solutions (aka Full Trust Solutions)
  - Packaged and deployed using farm solution packages
  - Deployment requires farm administrator
  - Works in on-premises farms but not Office 365
  - Heavily used since SharePoint 2007
- Sandboxed Solutions
  - Introduced in SharePoint 2010 with very limited adoption
  - Deprecated by Microsoft in SharePoint 2013
- SharePoint ~~Apps~~ Add-ins
  - Introduced with SharePoint 2013
  - Designed for Office 365 and on-premises farms
  - Does not allow server-side code to run in SharePoint
  - Requires breadth of client-side development skills





# SharePoint Server-Side Object Model

- Accessible through **Microsoft.SharePoint.dll**
  - In-process Assembly DLL for .NET clients
  - Oldest & most mature API for SharePoint
  - Available in solution packages but not SharePoint apps
  - Farm solutions have full access to server-side API
  - Sandbox solutions can access only a subset



# Client-Side Object Model (CSOM)

- CSOM provides client-side API for SharePoint
  - Introduced in SharePoint 2010
  - Accessible using .NET, Silverlight and JavaScript
- CSOM expanded in SharePoint 2013
  - Search
  - Managed Metadata
  - User Profiles and Social Feeds
  - Business Connectivity Service (BCS)
  - Workflow
  - Publishing



# SharePoint REST API

- SharePoint 2016 provides REST API
  - Great alternative to CSOM when coding in JavaScript
  - Can also be called from server-side .NET code
  - Accessible from non-Windows platforms as well
  - Unlike CSOM, SharePoint REST API accessible to all
- Creating URIs for the SharePoint REST API
  - URIs created based on principles of REST and ODATA
  - [Target Site URL] + `_api` + [Target SharePoint object]

[http://intranet.wingtip.com/\\_api/web/lists/getByTitle\('Customers'\)](http://intranet.wingtip.com/_api/web/lists/getByTitle('Customers'))



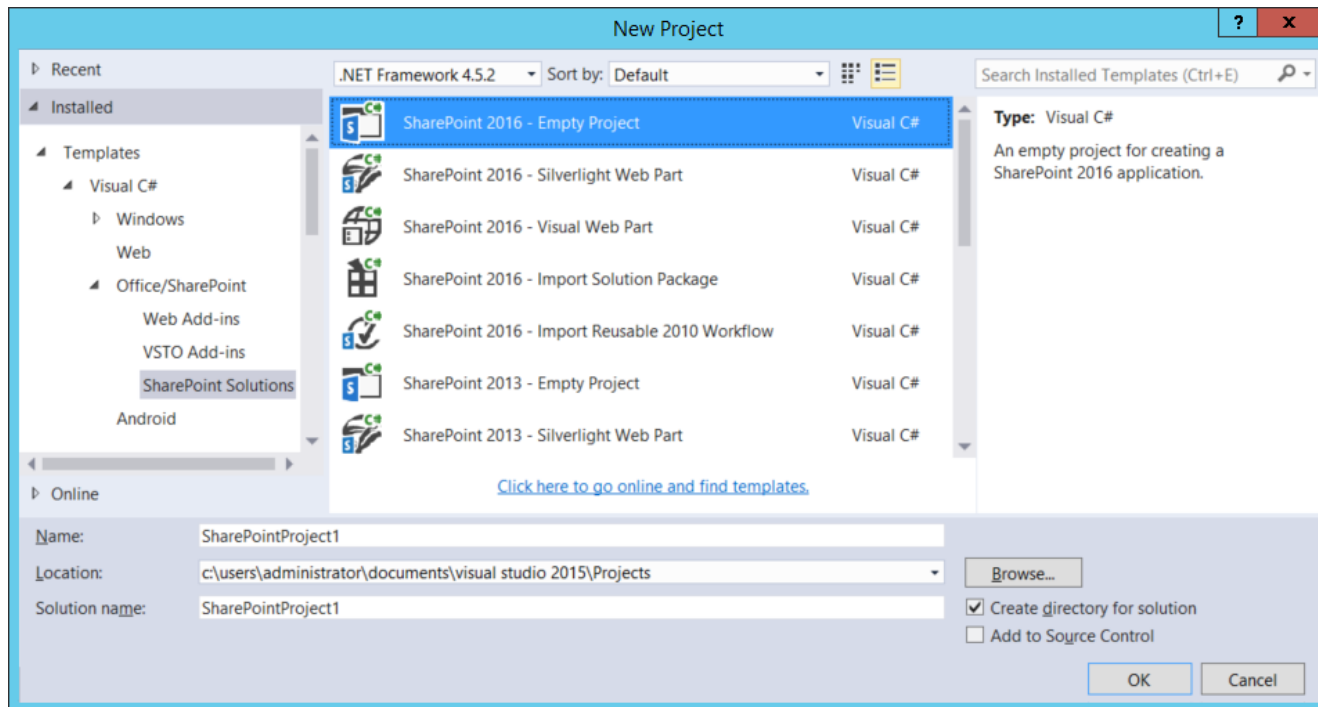
# Agenda

- ✓ SharePoint Architecture and Topology
- ✓ SharePoint Development Strategies
- SharePoint Developer Tools and Utilities
  - Creating a SharePoint Development Environment



# Visual Studio 2015

- Visual Studio supports SharePoint development
  - Project templates for SharePoint solutions and apps
  - Use Visual Studio 2015 with Update 2 or later
  - Install latest Visual Studio updates for SharePoint 2016



# Web Essentials

- Web Essentials 2015
  - Visual Studio 2015 Extension in Online Gallery
  - Additional IntelliSense for CSS3
  - Warnings & helpers for browser compatibility issues
  - Selector IntelliSense for HTML elements, classes, IDs
  - Web Essentials includes **JSHint**
  - Includes JSHint which detects problems in JavaScript





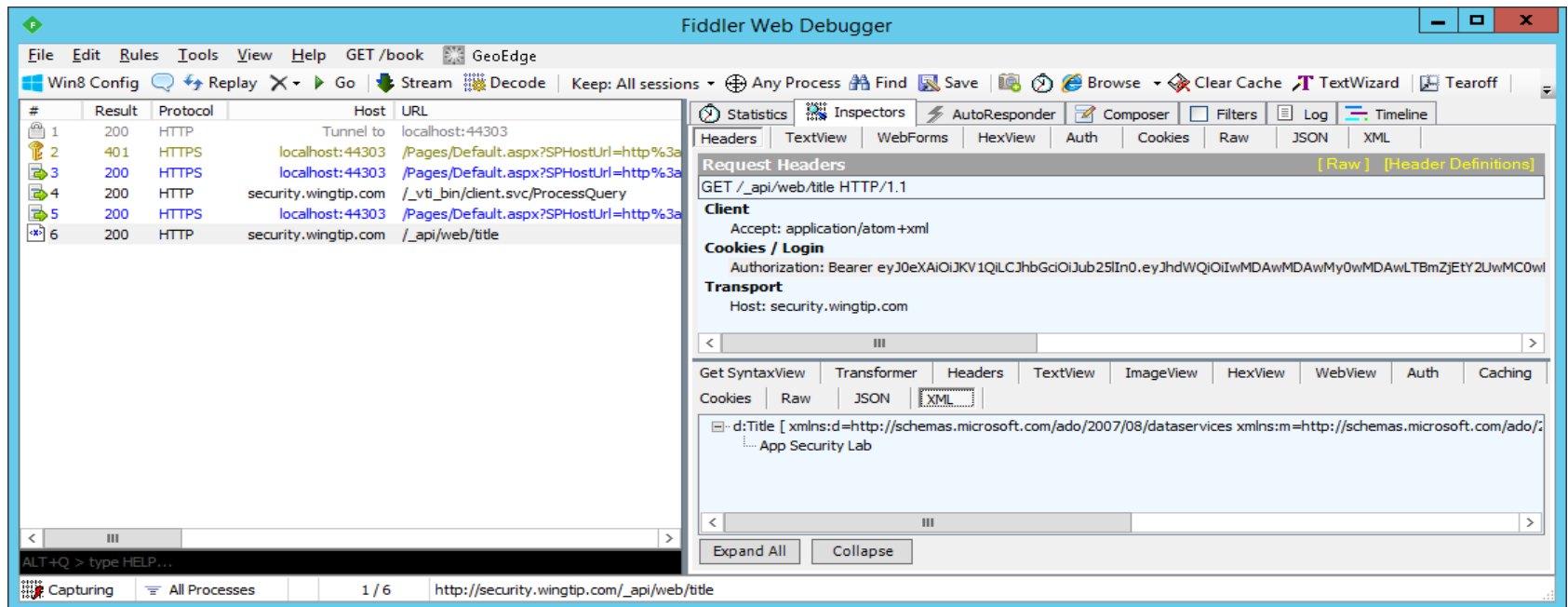
# Developer Tools provided by the Browser

- Browser-based development tools
  - View HTML
  - View & modify CSS
  - View & debug JavaScript
- **Internet Explorer**
  - Developer Tools: included
- **FireFox**
  - FireBug: extra download
- **Google Chrome**
  - Developer Tools: included



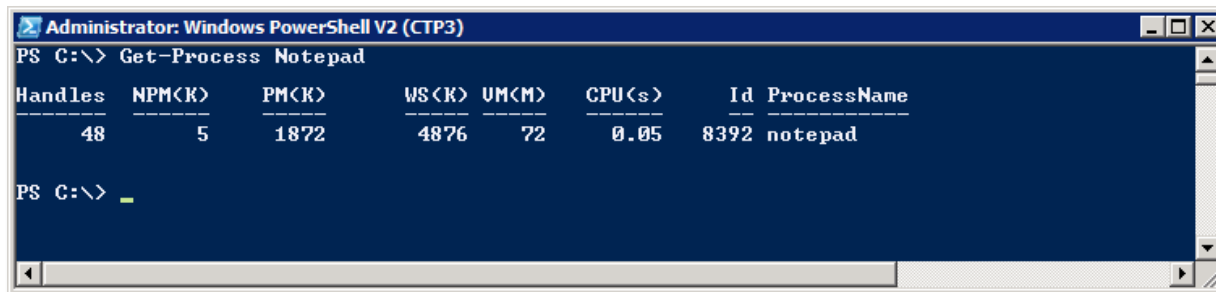
# Debugging HTTP request with Fiddler

- Fiddler is a HTTP debugging proxy
  - It helps you inspect HTTP request & response
  - Useful in debugging client-side JavaScript code
  - Useful in debugging SharePoint Workflows



# Working with Windows PowerShell

- SharePoint developers should learn PowerShell
  - Scripting environment for SharePoint administration
  - Used to create and manage test sites
  - Used to configure SharePoint environment
- Windows PowerShell fundamentals
  - Cmdlets (e.g. `Get-Process` and `Stop-Process`)
  - Pipelining and formatting features
  - Provider-based model for accessing resources



A screenshot of a Windows PowerShell V2 console window titled "Administrator: Windows PowerShell V2 (CTP3)". The prompt is "PS C:\>". The command "Get-Process Notepad" has been executed, resulting in a table of process information for Notepad.

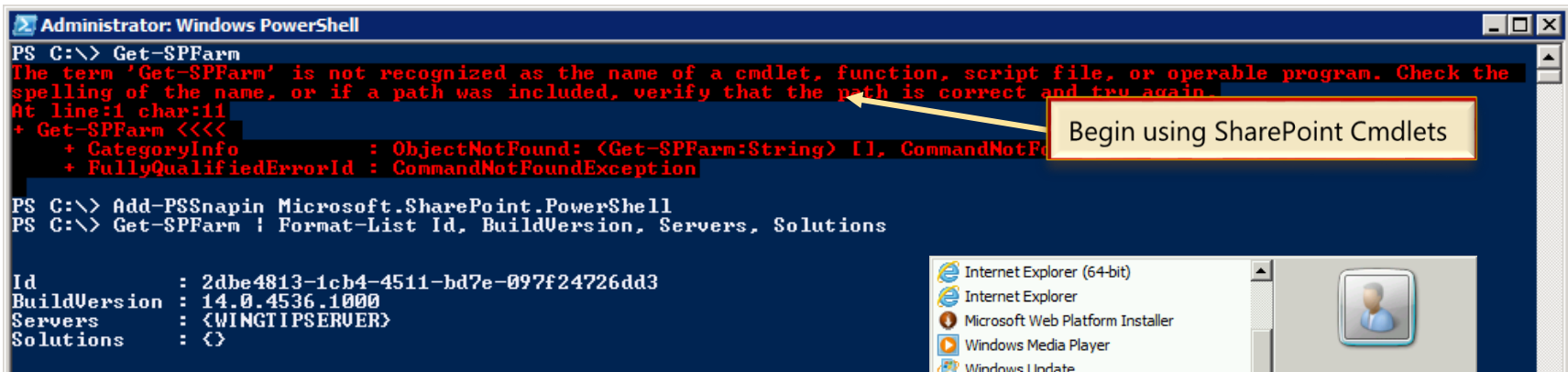
Handles	NPM(K)	PM(K)	WS(K)	UM(M)	CPU(s)	Id	ProcessName
48	5	1872	4876	72	0.05	8392	notepad

The prompt "PS C:\> \_" is visible below the table.



# The Microsoft.SharePoint.PowerShell Snapin

- Explicitly load SharePoint Windows PowerShell snap-in from console or script



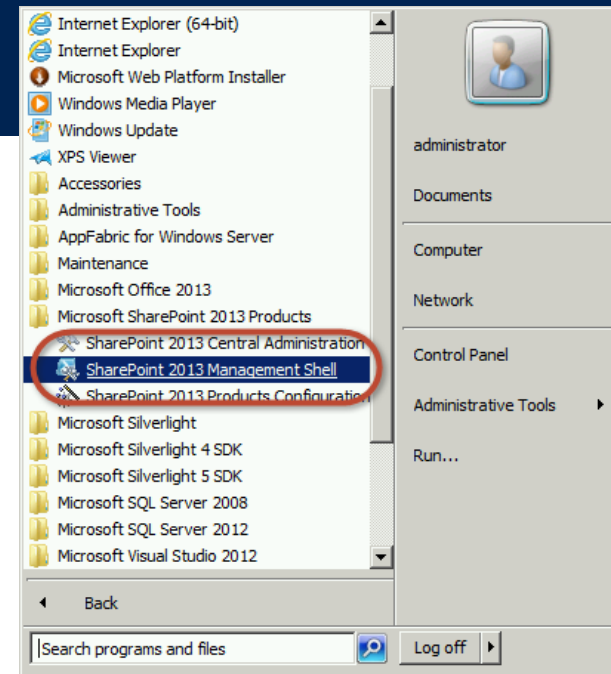
```
Administrator: Windows PowerShell
PS C:\> Get-SPFarm
The term 'Get-SPFarm' is not recognized as the name of a cmdlet, function, script file, or operable program. Check the spelling of the name, or if a path was included, verify that the path is correct and try again.
At line:1 char:11
+ Get-SPFarm <<<<
+ ~~~~~
+ CategoryInfo          : ObjectNotFound: (Get-SPFarm:String) [], CommandNotFoundException
+ FullyQualifiedErrorId : CommandNotFoundException

PS C:\> Add-PSSnapin Microsoft.SharePoint.PowerShell
PS C:\> Get-SPFarm | Format-List Id, BuildVersion, Servers, Solutions

Id                : 2dbe4813-1cb4-4511-bd7e-097f24726dd3
BuildVersion      : 14.0.4536.1000
Servers           : {WINGTIPSERVER}
Solutions         : {}
```

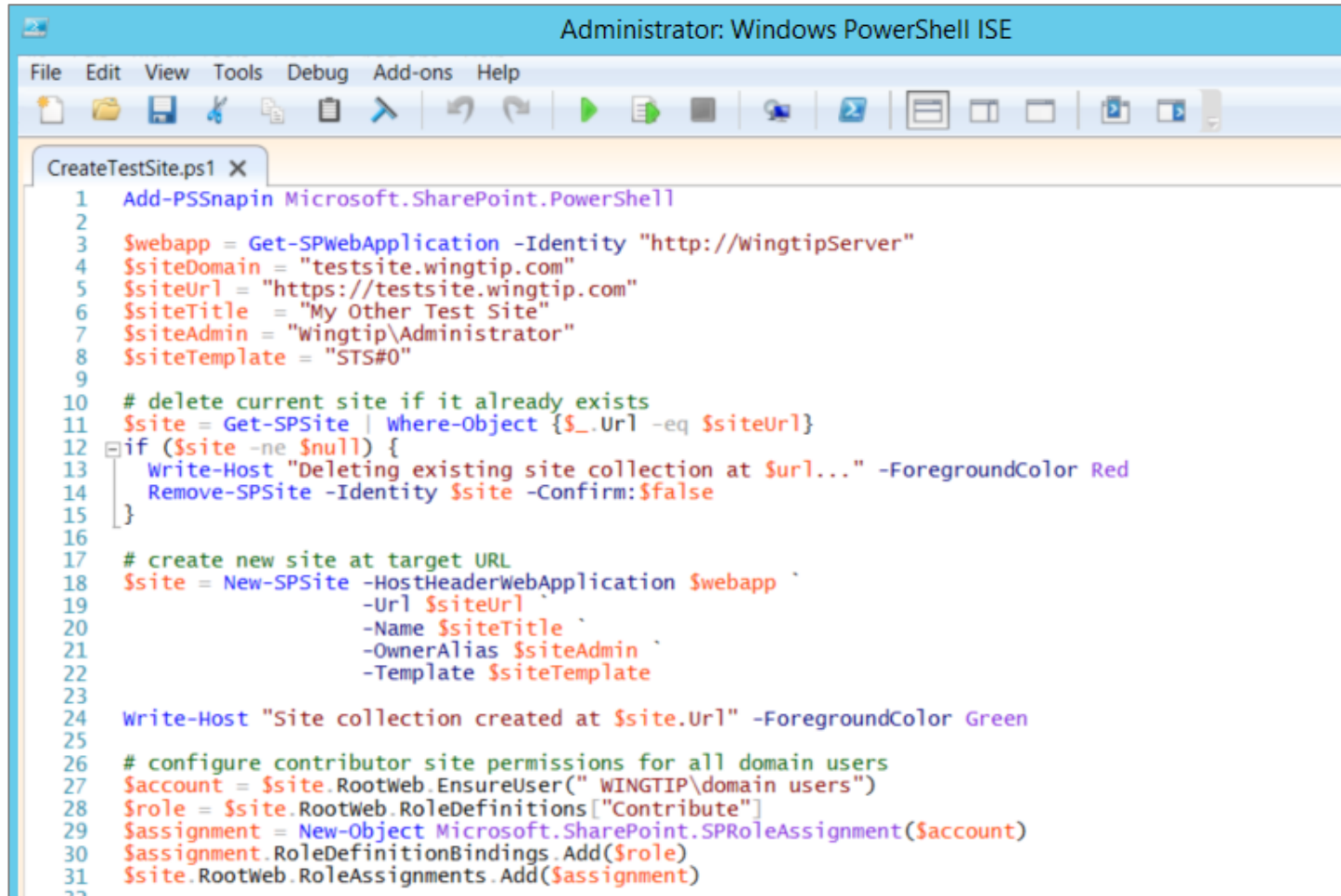
Begin using SharePoint Cmdlets

- Implicitly load snap-in by using link in Windows Start menu



# Windows PowerShell ISE

- Supports color-coding, IntelliSense and debugging



The screenshot displays the Windows PowerShell ISE (Integrated Scripting Environment) window. The title bar reads "Administrator: Windows PowerShell ISE". The menu bar includes "File", "Edit", "View", "Tools", "Debug", "Add-ons", and "Help". The toolbar contains various icons for file operations, editing, and execution. The main editor area shows a script named "CreateTestSite.ps1" with the following content:

```
1 Add-PSSnapin Microsoft.SharePoint.PowerShell
2
3 $webapp = Get-SPWebApplication -Identity "http://wingtipServer"
4 $siteDomain = "testsite.wingtip.com"
5 $siteUrl = "https://testsite.wingtip.com"
6 $siteTitle = "My Other Test Site"
7 $siteAdmin = "Wingtip\Administrator"
8 $siteTemplate = "STS#0"
9
10 # delete current site if it already exists
11 $site = Get-SPSite | Where-Object {$_.Url -eq $siteUrl}
12 if ($site -ne $null) {
13     Write-Host "Deleting existing site collection at $url..." -ForegroundColor Red
14     Remove-SPSite -Identity $site -Confirm:$false
15 }
16
17 # create new site at target URL
18 $site = New-SPSite -HostHeaderWebApplication $webapp `
19     -Url $siteUrl `
20     -Name $siteTitle `
21     -OwnerAlias $siteAdmin `
22     -Template $siteTemplate
23
24 Write-Host "Site collection created at $site.Url" -ForegroundColor Green
25
26 # configure contributor site permissions for all domain users
27 $account = $site.RootWeb.EnsureUser(" WINGTIP\domain users")
28 $role = $site.RootWeb.RoleDefinitions["Contribute"]
29 $assignment = New-Object Microsoft.SharePoint.SPRoleAssignment($account)
30 $assignment.RoleDefinitionBindings.Add($role)
31 $site.RootWeb.RoleAssignments.Add($assignment)
32
```



# Troubleshooting Errors with ULS

- ULS is Unified Logging Service
  - SharePoint's log files located at `.. \15\Logs`
  - Configure level of logging for different categories:
    - Central Administration → Monitoring → Configure Diagnostic Logging
- Developer Tools for inspecting ULS logs
  - Merge-SPLLogFile cmdlet in PowerShell
  - ULS Log Reader Utility (ULSViewer.exe)
  - Developer Dashboard





# Developer Dashboard

- A utility for inspecting per-request diagnostics
  - Introduced in SharePoint 2010
  - Much improved in SharePoint 2013
  - Shows requests from start of dashboard session

The screenshot displays the SharePoint Developer Dashboard interface. At the top, a list of requests is shown, with the third request highlighted in blue:

- Request (GET:http://bcs.wingtip.com/\_layouts/15/devdash.aspx)
- Request (GET:http://bcs.wingtip.com/?AjaxDelta=1)
- Request (GET:http://bcs.wingtip.com/SitePages/DevHome.aspx?AjaxDelta=1)**
- Request (POST:http://bcs.wingtip.com/\_vti\_bin/client.svc/ProcessQuery)

Below the request list is a navigation bar with tabs: Server Info, Scopes, SQL, SPRequests, Asserts, Service Calls, **ULS** (selected), and Cache Calls.

The main content area shows a table of events:

Time	Event	Process	Thread	Level	Category	Area	Message
14:07:04.072	77a3	w3wp.exe (0x0DCC)	9416	Verbose	Logging Correlation Data	SharePoint Foundation	Starting correlation.
14:07:04.072	xmrv	w3wp.exe (0x0DCC)	9416	Medium	Logging Correlation Data	SharePoint Foundation	Name=Request (GET:http://bcs.wingtip.com/SitePages/Dev

At the bottom of the table, a partial message is visible: "Write Debug Scope start event Request".



# Enabling & Using Developer Dashboard

- Developer Dashboard must be enabled
  - Typically enabled using PowerShell script

ToggleDeveloperDashboard.ps1 X

```
Add-PSSnapin Microsoft.SharePoint.PowerShell
```

```
$contentService = [Microsoft.SharePoint.Administration.SPWebService]::ContentService  
$dashboardSettings = $contentService.DeveloperDashboardSettings
```

```
if ($dashboardSettings.DisplayLevel -eq "OnDemand"){  
    # developer dashboard is on - turn it off  
    $dashboardSettings.DisplayLevel = "Off"  
    $dashboardSettings.Update()  
    Write-Host "Developer dashboard disabled." -ForegroundColor Gray  
} else {  
    # developer dashboard is off - turn it on  
    $dashboardSettings.DisplayLevel = "OnDemand"  
    $dashboardSettings.Update()  
    Write-Host "Developer dashboard enabled." -ForegroundColor Gray  
}
```



# Agenda

- ✓ SharePoint Architecture and Topology
- ✓ SharePoint Development Strategies
- ✓ SharePoint Developer Tools and Utilities
- Creating a SharePoint Development Environment



# The SharePoint 2016 VM Setup Guide

- Use the SharePoint 2016 VM Setup Guide
  - <https://github.com/CriticalPathTraining/SharePoint2016VmSetupGuide>
- Guide has docs and scripts to build student VM
  - Windows server 2012 R2
  - Active Directory Domain Services
  - SQL Server 2016
  - SharePoint Server 2016
  - Support for SharePoint 2013 Workflows
  - SharePoint Designer 2013
  - Visual Studio 2015 with SharePoint 2016 Support



# Student files for GSA2016

- Student files for GSA2016 are kept in GitHub
  - <https://github.com/CriticalPathTraining/GSA2016>

The screenshot shows the GitHub repository page for **CriticalPathTraining / GSA2016**. The browser address bar displays the URL <https://github.com/CriticalPathTraining/GSA2016/tree/master/Student/Modules>. The repository has 3 watches, 2 stars, and 4 forks. The **Code** tab is selected, showing the file structure of the **Student / Modules** directory. A red arrow points from the **01\_GettingStarted** folder in the left sidebar to a dropdown menu that lists the files: **Demo**, **Lab**, **Lab.pdf**, **Lab.xps**, **Slides.pdf**, and **Slides.xps**. The main table shows the following files and their commit history:

File	Commit	Time
01_GettingStarted	updates	2 days ago
02_FullTrustSolutions	updates	2 days ago
03_SharePointHostedAddins	updates	2 days ago
04_SharePointRestAPI	updates	2 days ago
05_Angular	updates	2 days ago

# Summary

- ✓ SharePoint Architecture and Topology
- ✓ SharePoint Development Strategies
- ✓ SharePoint Developer Tools and Utilities
- ✓ Creating a SharePoint Development Environment

