

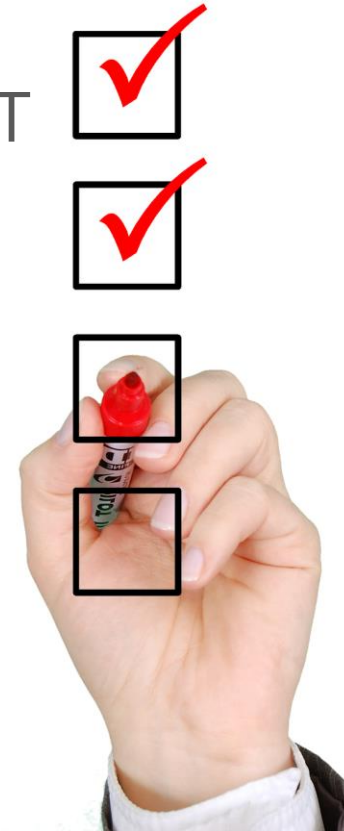
GrapeCity Spread Services: Spreadsheet Document Server Component for .NET Core



Presenter:
Sean Lawyer
Product Manager
sean.lawyer@grapecity.com

Agenda

- What is Spread Services - Overview
- Platforms and Targets Supported
- API power: Fast Purely-Managed VSTO in .NET
 - Load and Save document as XLSX or SSJSON
 - Generate and process documents for reports & dashboards, calculating complex models, building decision tables, etc.
 - GrapeCity Spread.Sheets back-end processing
- Demo Time
 - Web Demo Sample and General API Overview
 - How Fast is it? (REALLY FAST!)
- Licensing and Evaluation
- Q&A



What is Spread Services - Overview

- Spreadsheet Document object model based on MS Excel VSTO
- Pure Managed Code built for .NET Core and .NET Standard
- Ultra-fast calculations using Emit to generate dynamic IL (except iOS)
- Load any XLSX document and access all its content
- Integrate directly with any .NET platform (v4.6+)
- Special Support for Spread.Sheets: Load/Save SSJSON server-side
- Create web services or web applications using .NET Core for generating XLSX or SSJSON content, or performing server-side calculations and returning the results

Platforms and Targets Supported

- Supports .NET Core 1.0+
- Works in all platforms supported by .NET Standard 1.4+:

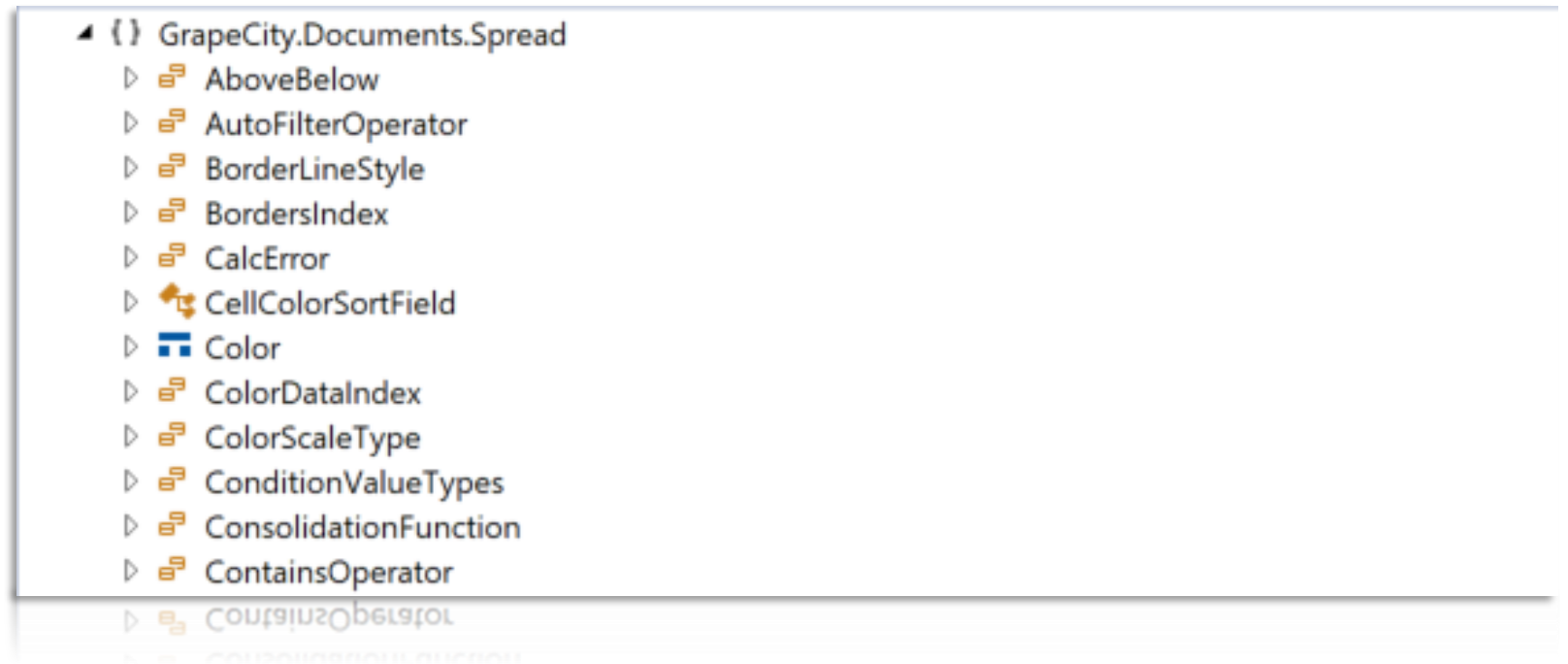
- .NET Core
- .NET Framework
- Mono
- Xamarin.iOS
- Xamarin.Mac
- Xamarin.Android
- UWP

.NET Standard	<u>1.4</u>	<u>1.5</u>	<u>1.6</u>	<u>2</u>
.NET Core	1	1	1	2
.NET Framework (with .NET Core 1.x SDK)	4.6.1	4.6.2		
.NET Framework (with .NET Core 2.0 SDK)	4.6.1	4.6.1	4.6.1	4.6.1
Mono	4.6	4.6	4.6	5.4
Xamarin.iOS	10	10	10	10.14
Xamarin.Mac	3	3	3	3.8
Xamarin.Android	7	7	7	8
Universal Windows Platform	10	10.0.16299	10.0.16299	10.0.16299

- Deploy .NET Core Serverless Apps to Azure Functions or to Amazon AWS Lambda

Full Featured High-Performance VSTO Excel API

- 146 interfaces
- 112 enumerations
- 14 classes (3 sealed)
- 1 attribute
- 273 total types
- Mirrors Excel VSTO
- Fast Performance: can load ~ *1.5M cells/second*
- Fast Calculation: uses Emit to compile formulas to IL for fast recalculation
- Memory efficient: storage model uses minimal memory for fast performance

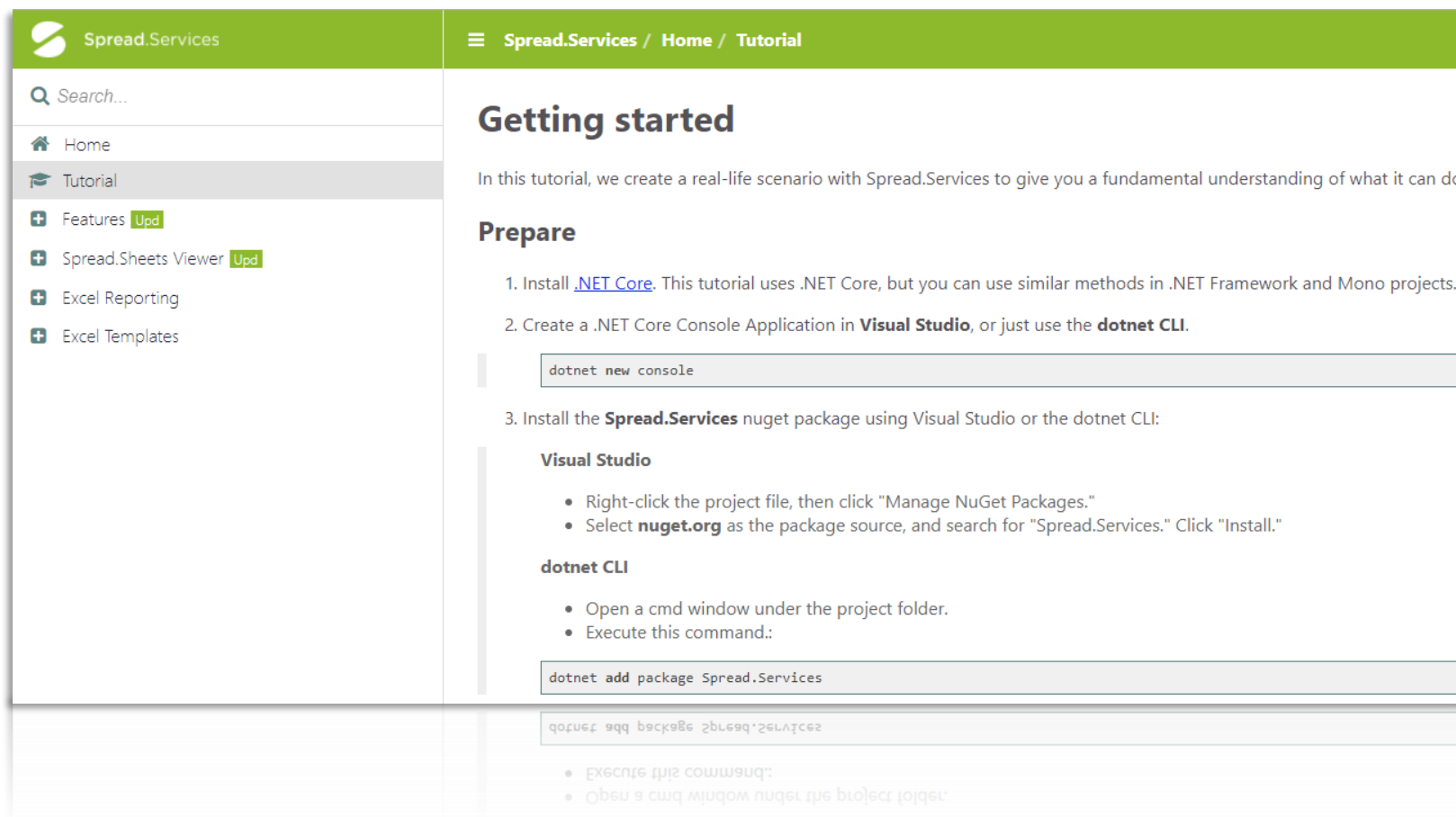


Common Use Cases

- Create .NET Core Web API for:
 - Dashboards/Reporting/Modeling/Calculations
 - Create/modify decision tables, split business intelligence logic from application
 - Document processing – combine/split workbooks
 - Spreadsheet automation – implement back-end services to automate tasks, port VBA from XLSM to VB.NET or C#
 - Spread.Sheets web application back-end processing (e.g. server-side recalculation)
- Create .NET Core web applications
 - Integrate with Spread.Sheets using SSJSON
 - Use API to render content directly, or to create JSON for client-side data sources
- Integrate with any .NET 4.6+ application for back-end spreadsheet document processing (Windows Forms, ASP.NET, WPF, UWP, etc.)

Demo Time – Let's Build Some Spreadsheets!

- Feature Demos
- Spread.Sheets Viewer
- Excel Reporting
- Excel Templates
- Getting Started Tutorial
- Benchmark Demo
- Angular Demo
- React Demo



The screenshot shows the Spread.Services website interface. The left sidebar contains a navigation menu with the following items: Home, Tutorial (highlighted), Features (with an 'Upd' badge), Spread.Sheets Viewer (with an 'Upd' badge), Excel Reporting, and Excel Templates. The main content area is titled 'Getting started' and includes the following text: 'In this tutorial, we create a real-life scenario with Spread.Services to give you a fundamental understanding of what it can do.' Below this is a 'Prepare' section with three numbered steps: 1. Install .NET Core. 2. Create a .NET Core Console Application in Visual Studio or use the dotnet CLI. 3. Install the Spread.Services nuget package using Visual Studio or the dotnet CLI. The 'dotnet CLI' section includes a code block:

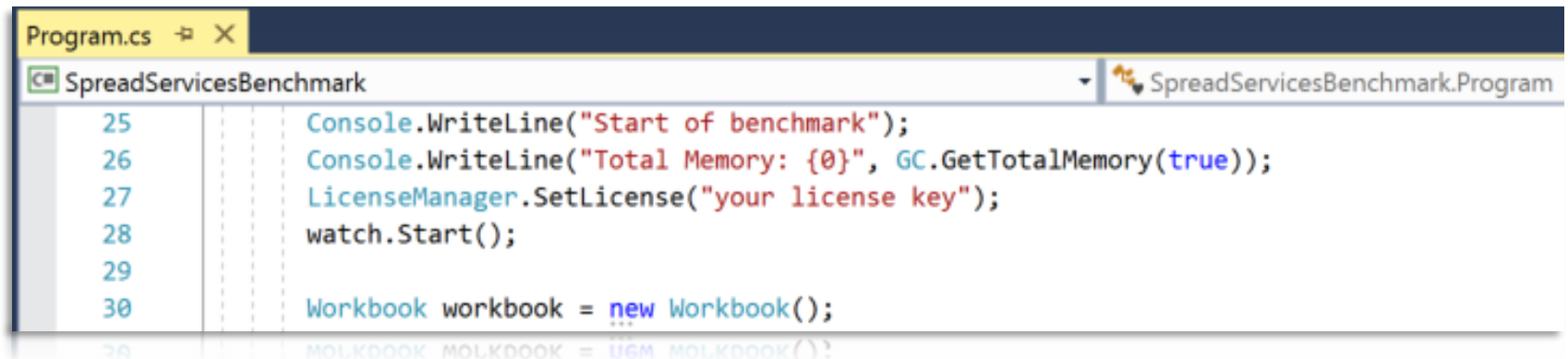
```
dotnet new console
```

 and the 'Visual Studio' section includes a list of instructions: 'Right-click the project file, then click "Manage NuGet Packages."' and 'Select nuget.org as the package source, and search for "Spread.Services." Click "Install."'.

Licensing and Evaluation

- Requires a Developer license + Distribution License Key (one included)
 - 1 distribution license required per separate distribution (“server hostname, named mobile, or desktop application” – [GrapeCity EULA](#))
 - Included Distribution License Key is Annual (requires annual renewal fee)
 - Perpetual Distribution License Keys also available (valid only for that version, upgrade to new version requires new developer license)

➤ Example of License Code:



```
Program.cs [X]
SpreadServicesBenchmark SpreadServicesBenchmark.Program
25 Console.WriteLine("Start of benchmark");
26 Console.WriteLine("Total Memory: {0}", GC.GetTotalMemory(true));
27 LicenseManager.SetLicense("your license key");
28 watch.Start();
29
30 Workbook workbook = new Workbook();
31 MOLKBOOK MOLKBOOK = UGM MOLKBOOK();
```

- Without the Distribution License Key, some limitations apply:
 - Maximum 100 files opened/saved + Evaluation Version Notice Worksheet in saved files
- Evaluation Distribution License Keys available on request
 - Full Function during trial period, contact sales.us@grapecity.com to request a key

Wrap up

- The ultimate .NET Core tool for:
 - *Fast XLSX*
 - *Fast Calculation*
 - *Fast Excel VSTO API*
- Create Web API, Web Applications and more
 - Deploy to serverless Azure Functions or AWS Lambda
 - Spreadsheet Automation/processing, Decision tables,
 - Reporting/dynamic spreadsheet generation
- Integrate with Spread.Sheets (sold separately)
 - Implement important logic server-side for code security
 - Server-side calculation
 - Easy to use with React, Angular, etc.



Next Steps

- Start a trial of Spread Services. - <https://www.grapecity.com/en/download/spreadsheets>
- Download demo samples from github. - <https://github.com/grapecity/Spread.Services-for-.NET>
- Contact support, participate in forums, get your questions answered. - <https://www.grapecity.com/en/support/contact>
- Contact sales for more licensing information or to schedule an exclusive demo. us.sales@grapecity.com
- Register for upcoming webinar topics. - <http://bit.ly/GCWebinars>
- Stay in touch!
 - <http://bit.ly/GCContact>
 - Twitter: @GrapeCityUS



Thank You!



Sean Lawyer
Product Manager
Sean.Lawyer@grapecity.com
<https://www.grapecity.com/en/spreadsheets/>
@GrapeCityUS