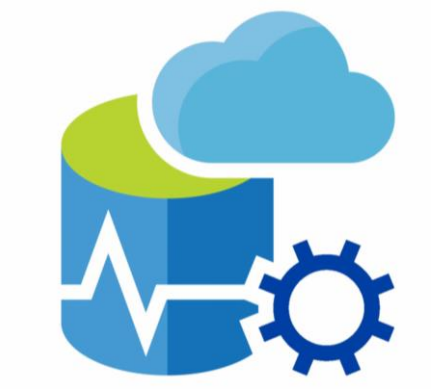
Drew Skwiers-Koballa



Azure Data Studio Extension Development

- **Visit the Sponsor Booths**
- **Lots of Great Raffle Prizes!**
- **Get your parking paid via Sponsor Bingo**

Thank you Sponsors!

Platinum Sponsor:

NUTANIX™

Gold Sponsors:

Concurrency

vmware®

SentryOne®

New Horizons Computer Learning Centers

Analytics8

Tail Wind Informatics

CoveringData CONSULTING

fulton

DELL EMC

PURESTORAGE®

Global Alliance Partners:

Microsoft Azure

PASS

PASSMN – News/Info

passmn
SQL Server User Group

Thanks to all our sponsors of 2019!

We need Speakers & Sponsors for 2020 PASSMN Meetings!

- Sign up to present at one of the monthly meetings!

Monthly Meetup: 3rd Tuesday of Each Month (except Oct) at Microsoft MTC in Edina (food usually provided)

Signup on Meetup: https://www.meetup.com/MN-SQL-Server-User-Group-PASSMN/

Board Member Elections in November/December:

- Your chance to help out the MN SQL community!

PASS SUMMIT2019

Join the brightest data professionals focused on the Microsoft Data Platform!

November 3th Through November 8th

- Pre-Conference Sessions – Monday/Tuesday
- Conference – Wednesday through Friday

SQLSaturday #913 – After Party

Location: 4th Floor of Mall of America
Time: 6:30PM – 10PM

There will be drinks and appetizers as well as free game cards and bowling!

Hang out with some new friends you've made.

Agenda

1. Why Develop an Extension?
2. Extension Development Framework*
3. Building Extension Features*
4. Package and Publish*
5. Wrap Up

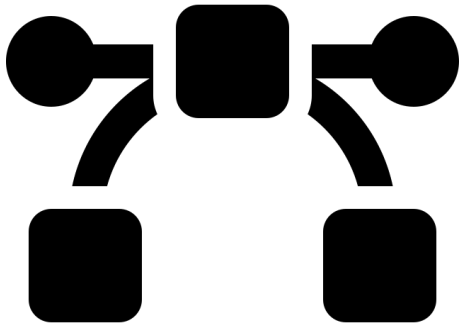
*Demo interspersed in 2-4

Why Develop an Extension?

Improve Your Workflow

Enterprise process

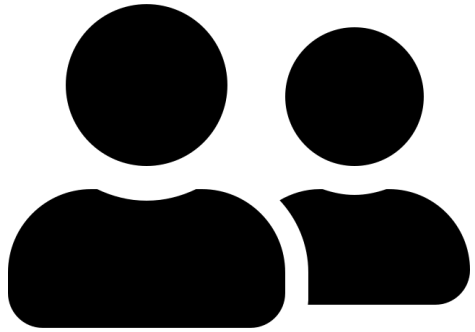
Specific shortcut



Share with the SQL Community

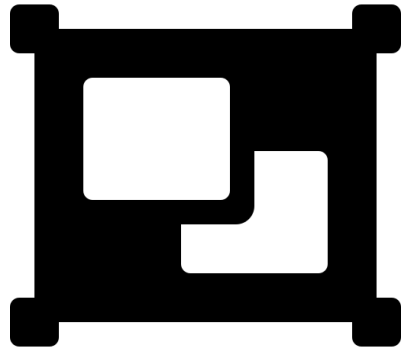
Better tools

Best practices



Close To Your Comfort Zone

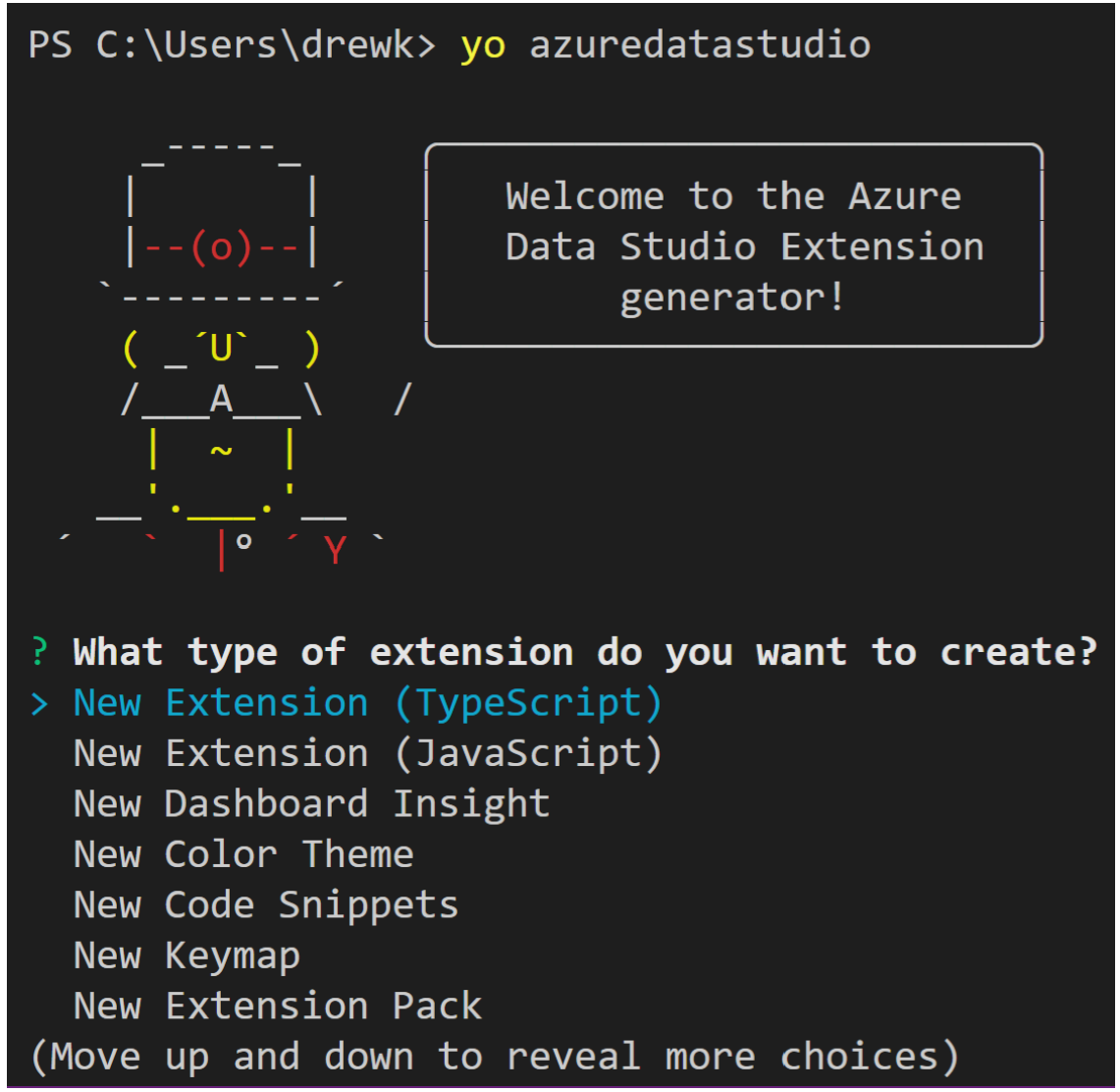
Development related to data platform



Extension Types

Chose One for a Template:

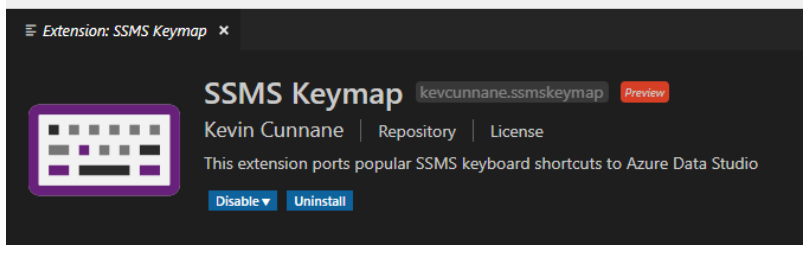
- TypeScript/JavaScript
- Dashboard Insight
- Color Theme
- Code Snippets
- Keymap
- Extension Pack
- Language Pack



Type Examples

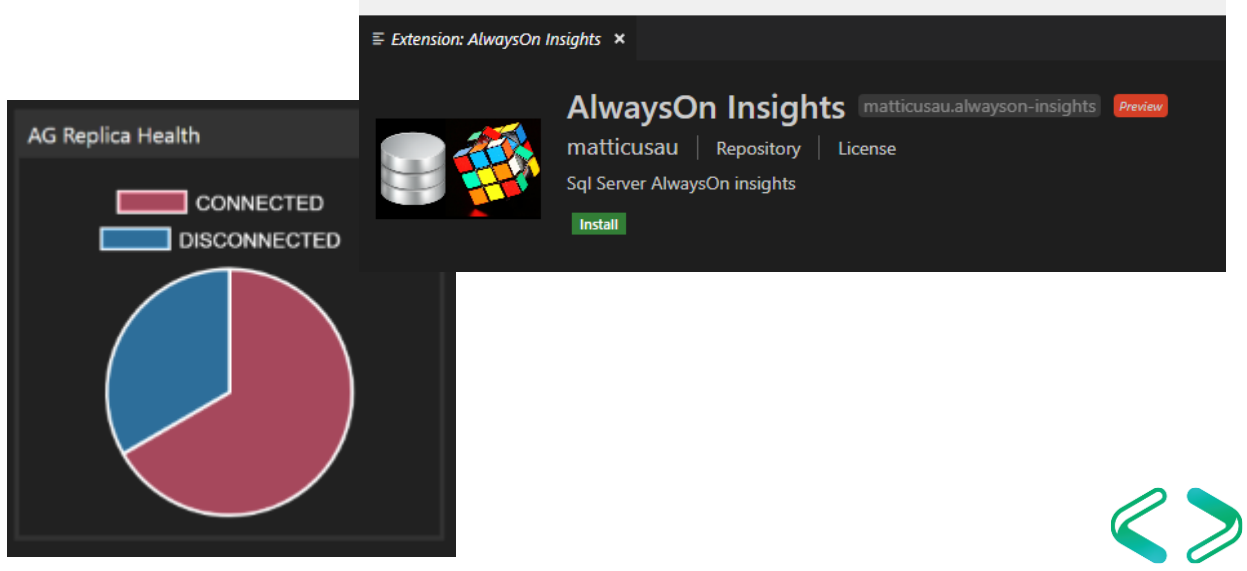
Keymap

Ties keystrokes to commands



Dashboard/Insight

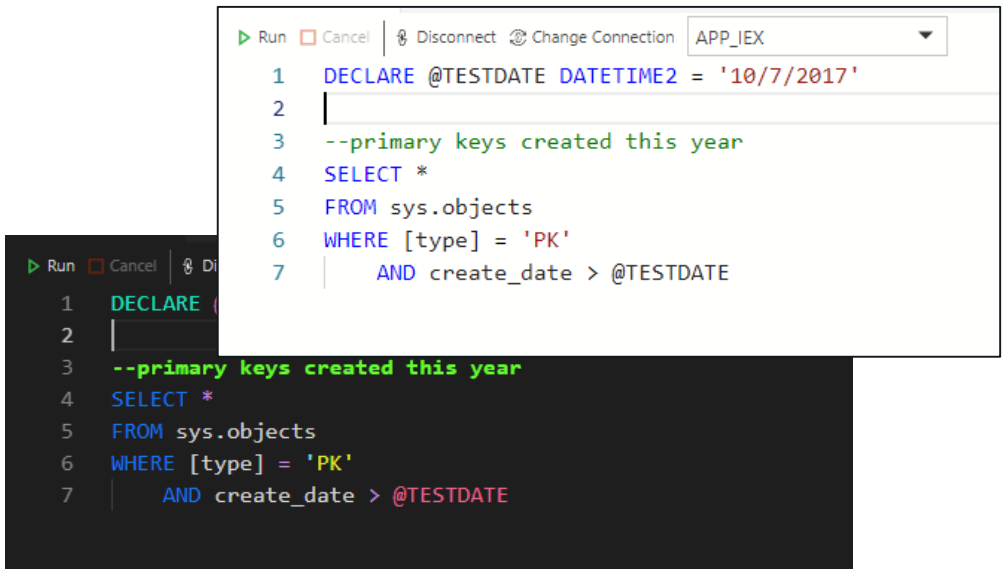
Packaged visualizations for database or server dashboards



Type Examples

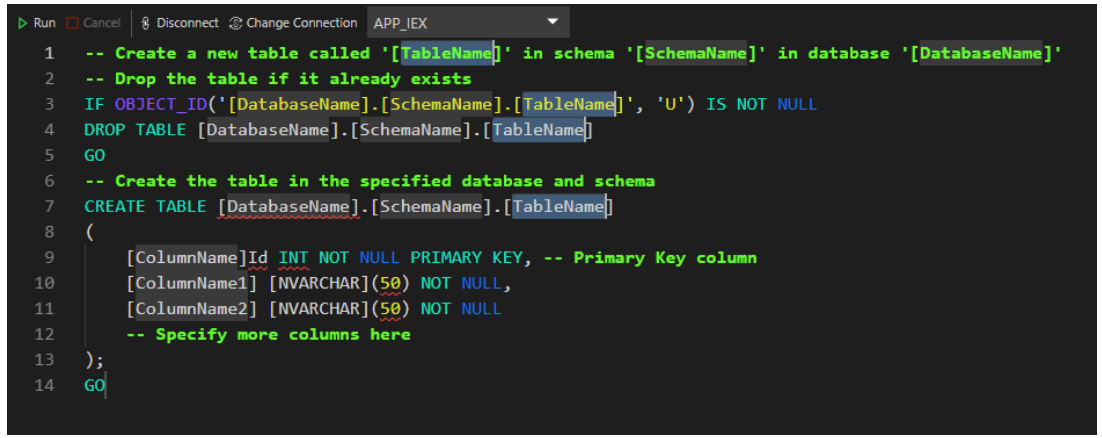
Color Theme

Editor color customization



Snippets

Packaged TSQL with tabstops



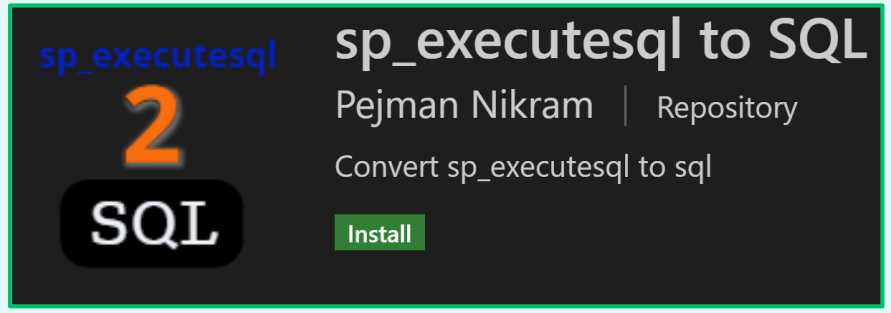
Type Examples

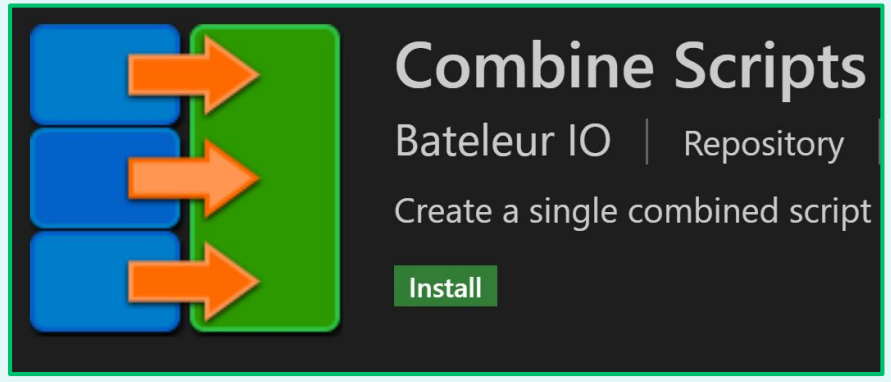
Extension Pack

Admin Pack for SQL Server is a c
you will download the following

- SQL Server Agent
  - List SQL Server Ag
  - View Job History w
  - Basic Job Control t
- SQL Server Profiler
  - Browse through ex
  - View and manage
  - Filter search of eve
- SQL Server Import
  - Use the Import Fla
- SQL Server dacpac
  - Use the Data-Tier

TypeScript/JavaScript





Extension Development Framework

Your Prerequisites

Workstation OS

Windows, Mac, or Linux OS

**ChromeOS*

Knowledge Requirements

Beginner Git

Beginner TypeScript

A Quick Poll:

Do You Want To Play Along?

1. Yes
2. No
3. Undecided

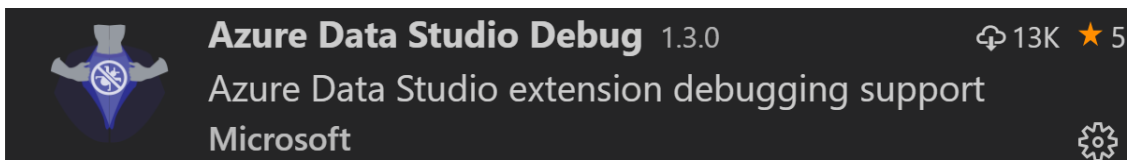
System Prerequisites

Applications

VS Code

Azure Data Studio Debug
(extension for VS Code)

Azure Data Studio



Development Tools

Git

NodeJS

Checking Installs

Powershell

```
PS C:\Users\drewk> git --version
git version 2.22.0.windows.1
```

```
PS C:\Users\drewk> node -v
v10.16.0
```

Bash

```
drewsk@drewsk-2018:~$ git --version
git version 2.17.1
```

```
drewsk@drewsk-2018:~$ nodejs -v
v8.10.0
drewsk@drewsk-2018:~$
```

You don’t need command line experience to get started with extension development.

System Prerequisites

```
npm install -g typescript

npm install -g yo azuredatastudio

npm install -g vsce
```

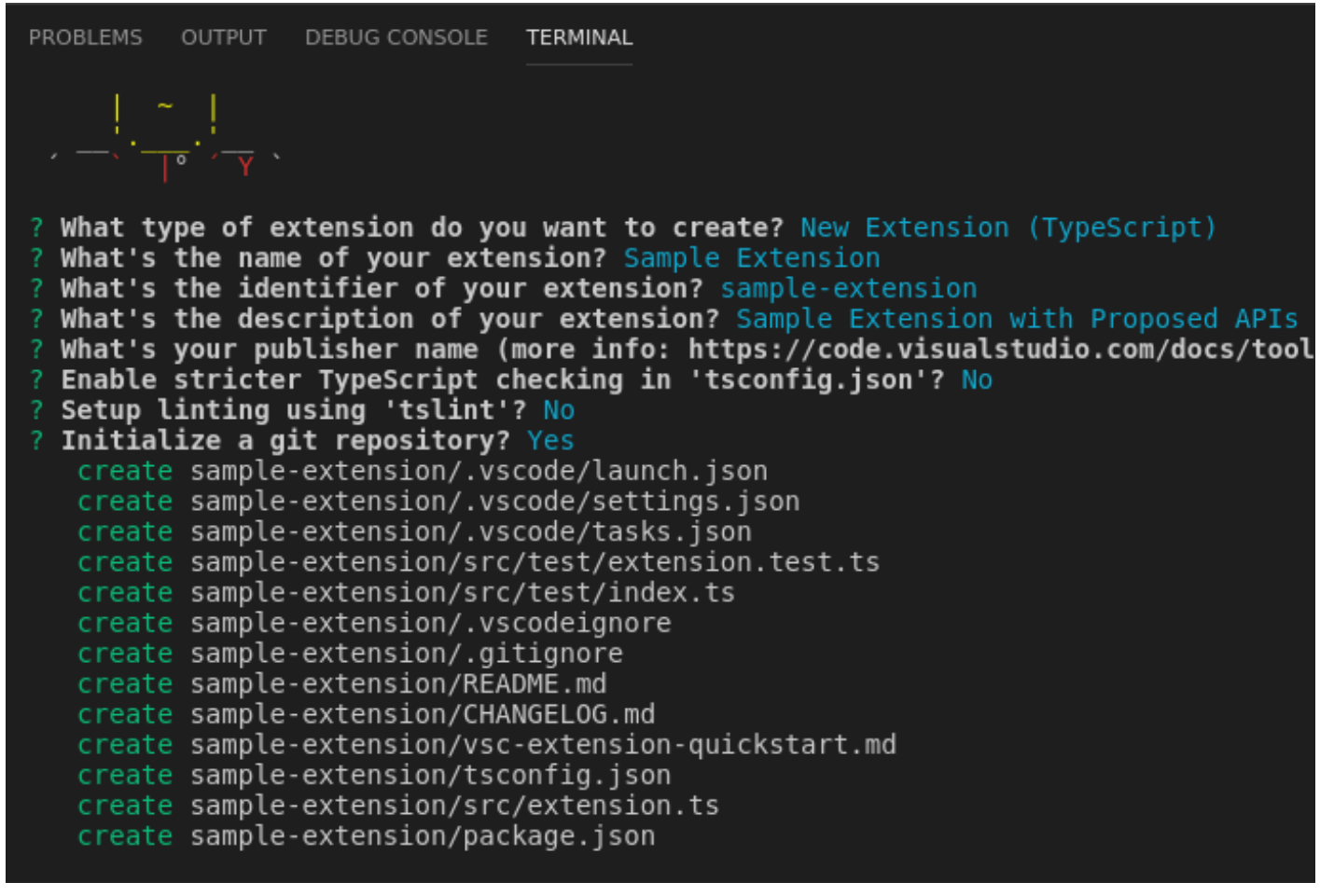
Install Through *npm*

- TypeScript
  - JavaScript with type checking + more
- Yeoman Extension Generator
  - *yo*
  - Open source extension template builder
- VS Code Extension Manager
  - *vsce*
  - Packages extension into .vsix for installing into Azure Data Studio

Yo

yo azuredatastudio

- For this sample, leave the *Typescript Extension* selection
- Enter a name, identifier, description, and your publishing name
- Decline stricter checking and linting
- Select the initialization of a Git repository



Demo Outline

We're Going to Explore:

- Activation events: when the extension is "started", code entry point
- Contribution points: additions to the application interface
- VS Code APIs + Azure Data Studio APIs
- *Alternative NodeJS packages*
- *Extended extension architecture*

Initial Extension Files

```
.
├── .vscode
│   ├── launch.json
│   └── tasks.json
├── .gitignore
├── README.md
├── src
│   └── extension.ts
├── package.json
└── tsconfig.json
```

.vscode folder controls how VS Code interacts with the project

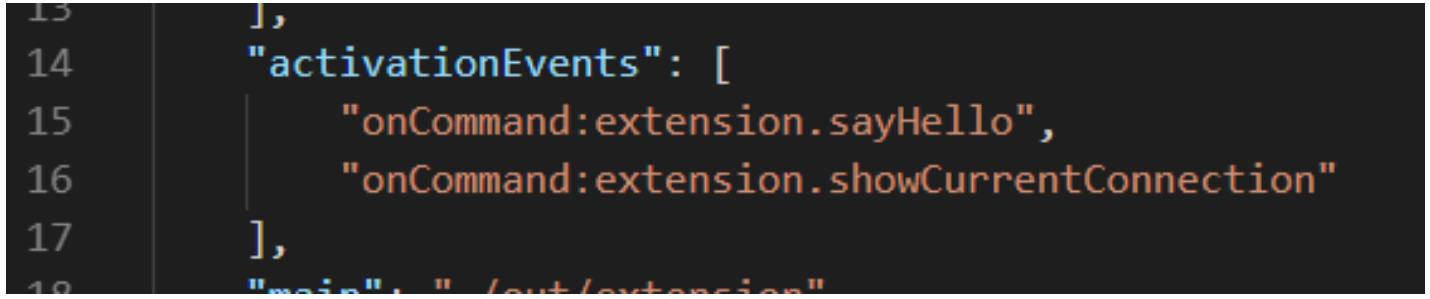
README.md stores documentation

Extension.ts contains code entry point(s)

Package.json = Extension Manifest

Package.json

Activation Events



When the extension is loaded – consumes CPU/memory

- onCommand
- workspaceContains
- onFileSystem
- onView
- * *(startup)*

Package.json

Contribution Points

Commands
Configuration
Menu
Keybindings
Themes
Snippets
Dashboard
Container

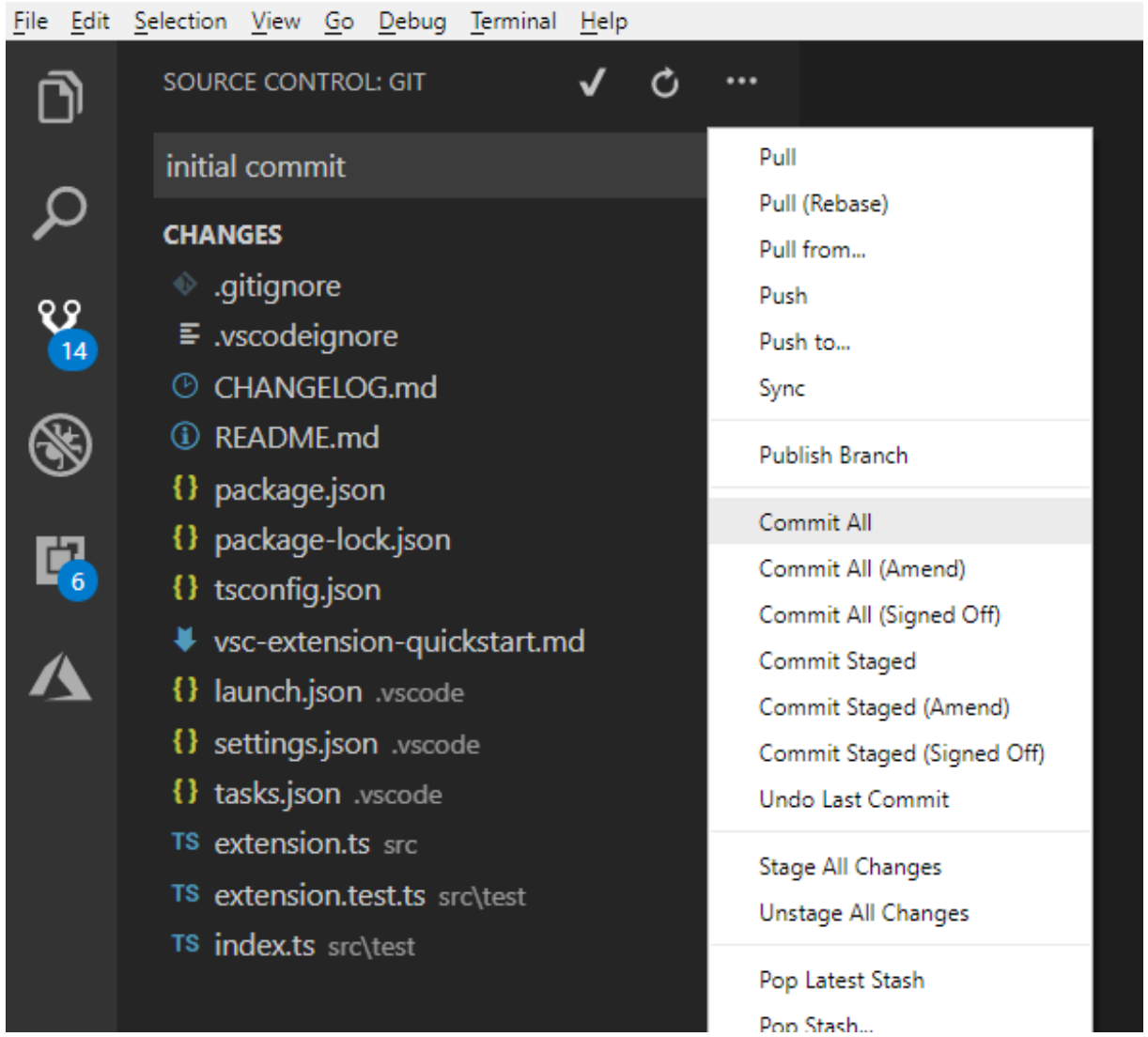
```
"contributes": {
    "commands": [···
    ],
    "keybindings": [···
    ],
    "configuration": [
        {
            "title": "NewQueryTemplate",
            "properties": {
                "newquerytemplate.DefaultQueryTemplate": {
                    "type": "array",
                    "default": [
                        "--set a default new query template wit
                        "",
                        ""
                    ],
                    "description": "Query text to insert into n
                },
                "newquerytemplate.DefaultQueryLine": {
                    "type": "number",
                    "default": -1
                },
                "newquerytemplate.DefaultQueryCharacter": {
                    "type": "number",
                    "default": -1
                }
            }
        }
    ]
```

Initial Commit

- Open the folder for your extension in VS Code
- Stage and Commit all changes

Optional:

- Create a repository on GitHub without initializing
- Run command to add Git remote in VS Code

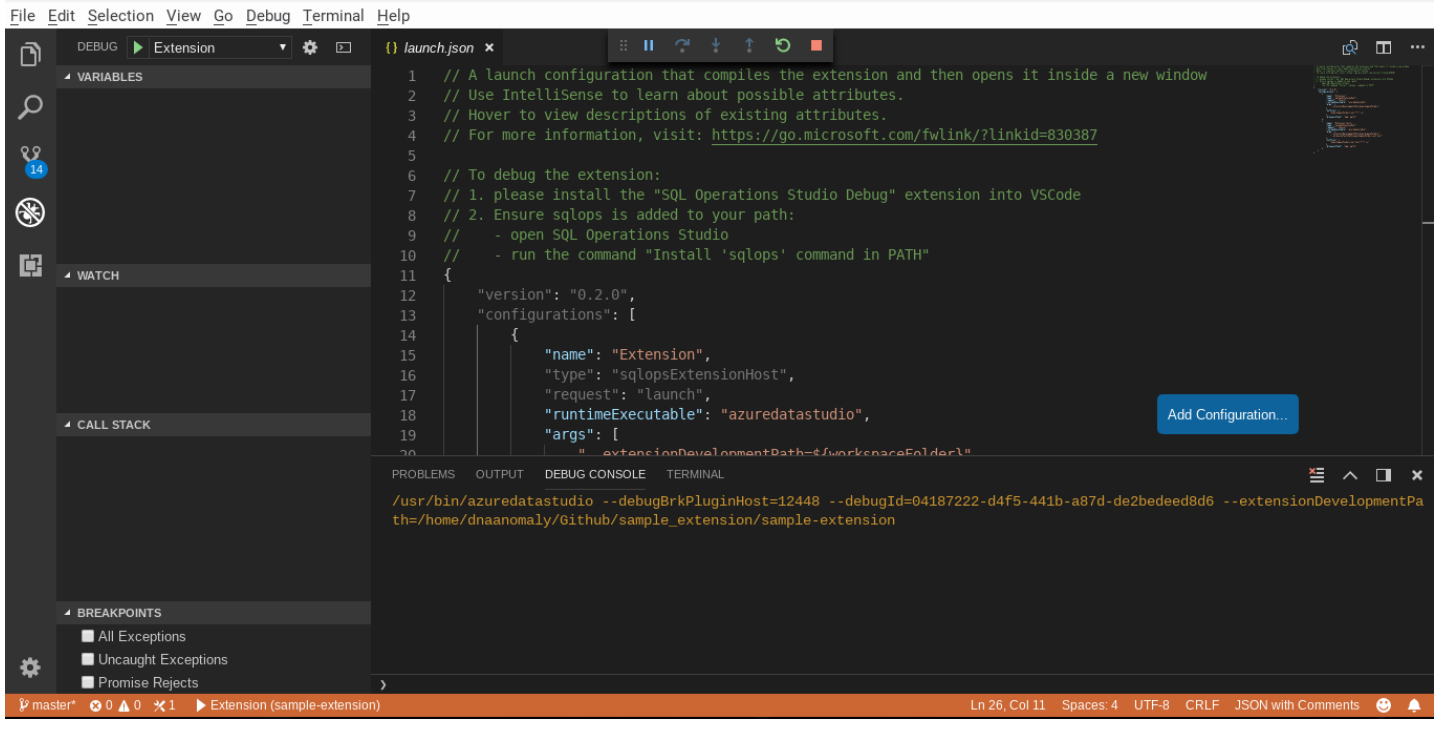


Test Extension

Select Start Debugging from the Debug menu

A special *Extension Development* instance of ADS opens

Test the extension from the command palette (ctrl+shift+P)



Building Extension Features

Azure Data Studio APIs

A few examples…

connection

ConnectionProfile

getCurrentConnection()

ServerInfo

objectexplorer

NodeInfo

Traverse object tree

findNodes()

https://github.com/microsoft/azuredatastudio/blob/master/src/sql/azdata.d.ts

Azure Data Studio APIs

DataProvider extends to:

ConnectionProvider – server connections

MetadataProvider – object info

ScriptingProvider – “script as”

QueryProvider – execute queries

ProfilerProvider – extended events

AgentServicesProvider – SQL agent

BackupProvider – backup and restore

https://github.com/microsoft/azuredatastudio/blob/master/src/sql/azdata.d.ts

VS Code APIs

window

Text editors

Terminals

User inputs

workspace

Currently open folder

FileSystem

tasks

Background scripts/processes

```
const editor = vscode.window.activeTextEditor;
let cursorPosition = editor.selection.start;
let lineCount = editor.document.lineCount;
```

More Info: https://code.visualstudio.com/api/references/vscode-api

Proposed APIs

azdata.proposed.d.ts

- APIs in preview
- Subject to breaking changes

Using Proposed APIs

- To install from command line after Yeoman generator:
  npm run proposedapi
- To install manually:
  Copy file to *src/typings* folder

https://github.com/microsoft/azuredatastudio/blob/master/src/sql/azdata.proposed.d.ts

NodeJS Packages

Additional Functionality

Parsing SQL, JSON, etc

HTTP requests

Filesystem access

Operating system info

npm/yarn

Package managers for NodeJS:

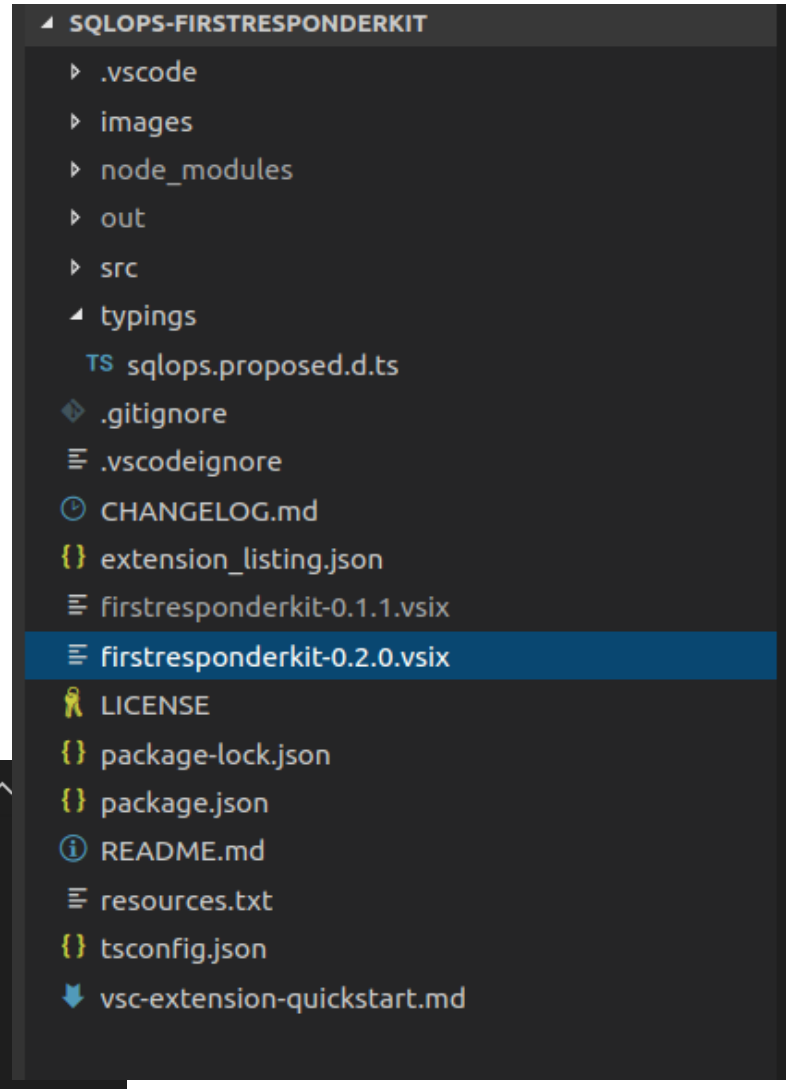
- Yarn
- npm

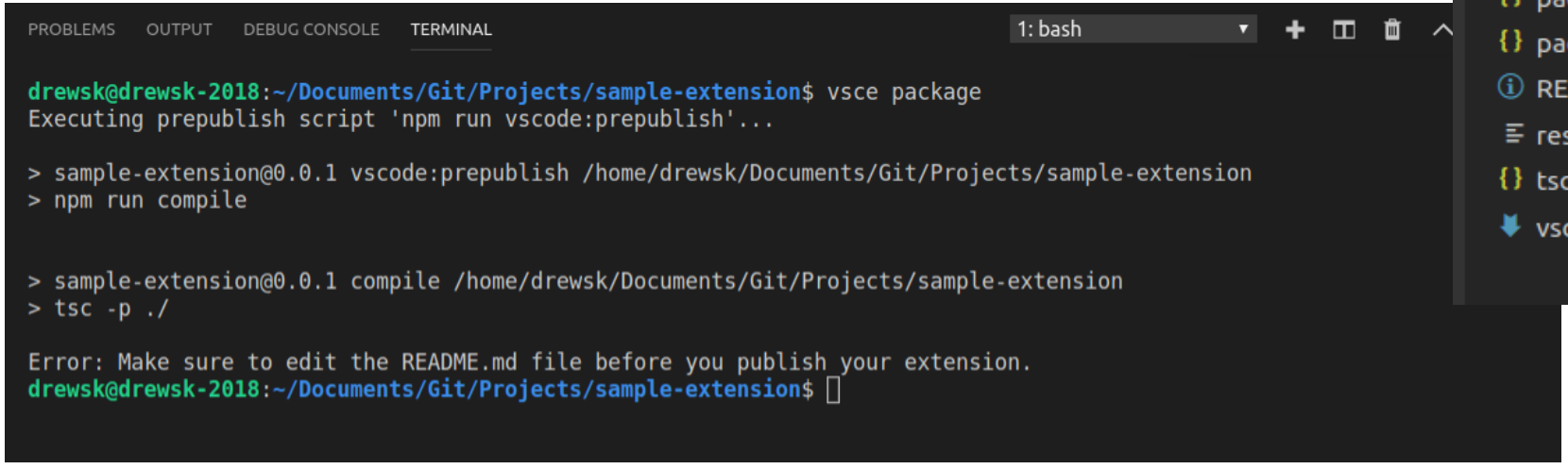
Start here: https://www.npmjs.com/

Package and Publish

Package to .vsix

- (update README.md)
- Run **vsce package** from terminal
- Load .vsix into ADS via command palette





Options for Publishing

A philosophical question:

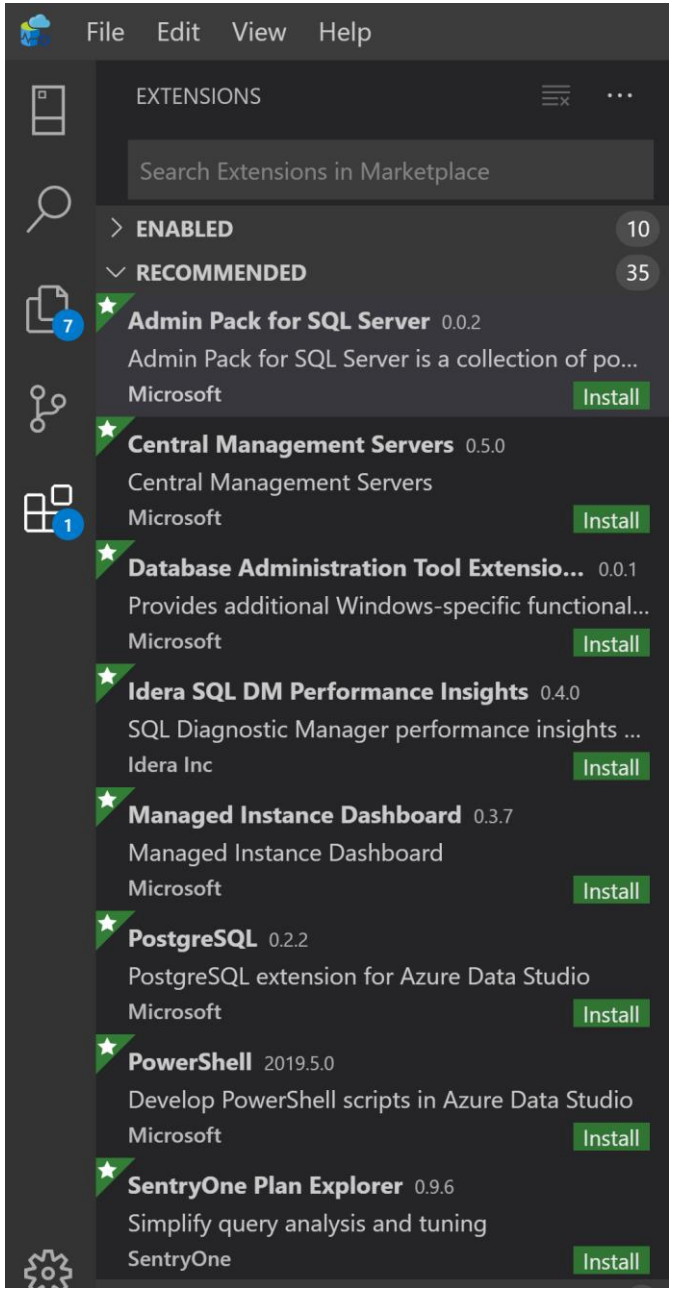
Open Source

Others can view your code, contribute improvements or fixes

Closed Source

Protect your intellectual property

Others can download and install the .vsix



Azure Data Studio Marketplace

Pull Request to release/extensions

Update [extensionsGallery.json]

extensionsGallery.json

files

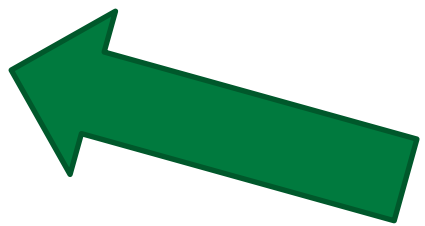
- SQLOps.DownloadPage
- VisualStudio.Services.Icons.Default, *optional*
- VisualStudio.Services.Content.Details
- VisualStudio.Code.Manifest
- VisualStudio.Services.Content.License

Values don't have to be links to GitHub

extensionsGallery.json

properties

- VisualStudio.Code.ExtensionDependencies
- VisualStudio.Services.Links.Source, *optional*
- AzDataEngine
- VisualStudio.Code.Engine



Set minimum versions

Wrap Up

Key Takeaways

Building an Extension in Azure Data Studio

- You can do it.
- VS Code extension + Azure Data Studio APIs
- Extension functionality can derive from both NodeJS and extended capabilities
- Extensions can be monetized and/or proprietary
- Open source contributions elevate the data platform community's capabilities

Additional Resources

- **https://github.com/dzsquared/AzureDataStudio_ExtensionDevelopment**
- https://code.visualstudio.com/docs/extensions/overview
- https://medium.com/@kevcunnane/extending-sql-operations-studio-hello-connected-world-part-1-of-n-e868542c6157
- https://www.drewsk.tech/2018/10/sql-saturday-796-minnesota-2018/
- https://medium.com/ingeniouslysimple/how-we-built-an-extension-for-sql-operations-studio-f93532ce4456
- https://cultivatehq.com/posts/how-we-built-a-visual-studio-code-extension-for-iot-prototyping/

Drew - @sysadmindrew drew.koballa@gmail.com