

# ANGULAR WORKSPACE

## The Ultimate Guide

---

Learn how to leverage the capabilities of the Angular Workspace to elevate your developer workflow.

---

*Techniques and strategies to minimize effort and maximize results.*



AngularArchitecture.com

build something amazing

Vers. 6, 7, 8 & 9



# MATT VAUGHN

# Angular Workspace - The Ultimate Guide

Matt Vaughn

This book is for sale at <http://leanpub.com/angular-workspace-the-ultimate-guide>

This version was published on 2020-08-28



This is a [Leanpub](#) book. Leanpub empowers authors and publishers with the Lean Publishing process. [Lean Publishing](#) is the act of publishing an in-progress ebook using lightweight tools and many iterations to get reader feedback, pivot until you have the right book and build traction once you do.

© 2020 Matt Vaughn

# Contents

<b>What is an Angular Workspace?</b> . . . . .	<b>1</b>
Workspace: Where You Work Matters . . . . .	1
May the 4th Be With You . . . . .	2
What Is So Special About Angular version 6? . . . . .	2
Monorepo with Angular Workspace . . . . .	3
Angular Workspace Improves Developer Efficiency . . . . .	3
<b>Workspace: A Container for Projects</b> . . . . .	<b>5</b>
Angular.json . . . . .	5
Applications . . . . .	5
Libraries . . . . .	7
<b>Anatomy of an Angular Application</b> . . . . .	<b>8</b>
<b>Share and Reuse Code</b> . . . . .	<b>9</b>
Why Share Code? . . . . .	9
How Can We Do It? . . . . .	9
<b>JavaScript Packages</b> . . . . .	<b>10</b>
Not Effective? . . . . .	10
<b>Nx Dev Tools</b> . . . . .	<b>11</b>
Upgrade Existing Angular Workspace . . . . .	11
Getting Started - New Workspace . . . . .	11
Create a New Workspace . . . . .	11
<b>Code Organization</b> . . . . .	<b>12</b>
Architectural Patterns . . . . .	12
<b>Library Project Types</b> . . . . .	<b>13</b>
<b>Benefits of a Monorepo Workspace</b> . . . . .	<b>14</b>
<b>Resources</b> . . . . .	<b>15</b>

# What is an Angular Workspace?

- multiple projects in a single development environment
  - application
  - library
- other monorepo examples
  - .NET and Java -> solutions of different projects
  - JavaScript: Lerna
  - Nrwl.io Nx Dev Tool

## Workspace: Where You Work Matters

Most everyone agrees that a nice workspace is not only enjoyable but essential to being effective. Angular gave us an impressive gift with version 6. This chapter provides an overview of the game-changing work environment for developing web applications. For years, other development environments (i.e., .NET, Java, etc.) have had rich IDEs that provide environments to create solutions with multiple projects of different types.

Programmers require the ability to organize code and projects related to each other. Today, the requirements for web applications require the same capabilities.

Web developers typically have a folder and the entire code base for the project is in that folder. There is no concept of sharing and reusing libraries in a solution (or workspace). If you want to create a JavaScript library, it is typically developed as a separate project, published to a package manager, and then consumed by other JavaScript applications. This is a workflow that has many steps - not a very efficient or effective workflow for rapid development. I know, because my team did this for Angular 2 applications up through version 5 for about 11 or 12 projects. Some of our packages had dependencies on other custom libraries. A small change in one library would require us to build and deploy several other libraries due to our dependencies. This was very time consuming.

So, what if there was a work environment that enabled you to manage multiple web application projects, create libraries, and share/reuse those libraries with any of the web applications and/or the other libraries without the overhead of publishing and consuming packages via npm? It was quite a surprise when Angular 6 was released with support for a Workspace environment that does all of these things.

The Angular Workspace at first appearance may not seem like much. The ability to add multiple projects is significant. However, the new supported project type for library projects may provide the biggest impact. It allows us to build modern web applications with new architectural patterns that simplify and make our code more powerful.

This book will focus on the changes the new Angular Workspace provides to implement strong enterprise-level architectures.

The Angular Workspace simplifies the management of multi-project environments. All projects now share the same package dependency configuration, the same Angular version, a single configuration for building, serving, linting, and testing projects. We got all of this with Version 6 of Angular!

We can have great web application architecture that is clean without the Angular Workspace. However, the additional CLI tooling along with the ability to share and reuse code via libraries is the biggest game-changer to web development workflows and work environments. Individuals and teams that leverage these capabilities will certainly be more efficient and effective - this means a lot in a competitive software industry.

## May the 4th Be With You

On, May 4th, 2018, Angular 6 was released. The [Angular.io](https://angular.io)<sup>1</sup> site actually says May 3rd. However, it appears that the press releases were pushed the following day. And I like the sound and reference to using the force with the date that most recognize as the public release date.

## What Is So Special About Angular version 6?

The Angular CLI command to create a new project was dramatically upgraded. This command no longer creates a project. It creates a Workspace.

```
1 ng new <my-workspace-name-here>
```

Most developers didn't notice much. The command above still worked - there were no breaking changes. I think there should have been, because the change was so significant - I think the new capability flew under the radar for awhile. This command created a new Workspace with a default application project where both had the same name. In fact, the default application project was in the same location as before. There was no indication that anything different was going on. Yes, there was a new angular.json file.

If you inspected this file, you noticed that there was a projects node with the default project containing a project type of application. Interesting.

```
1 "projectType": "application"
```

Note: Angular version 7, allows for an empty workspace with the `ng new` command. You would use the `ng generate application <name>` to add a new application to the workspace.

---

<sup>1</sup><https://angular.io/guide/releases>

## Monorepo with Angular Workspace

We just got a monorepo. A single environment where we can now develop multiple projects. Not just single-page web applications, but also library projects (more on this later).

Is this a good thing? It is for several reasons. Think about all of the time that could be saved without the overhead of configuring development environments. A monorepo allows a team to continually improve the workspace environment - and everyone benefits. It is like a community garden that continually gets better and always produces fine results.

One of the main goals of Angular is to enable companies, teams, and individual developers to be as effective as possible. Effectiveness also includes efficiency and building the right things at the right time. Therefore, to enable and provide the necessary environment to develop effective Angular solutions, we require the implementation of a single Workspace for all things Angular. This includes:

- all Angular applications
- all Angular and Typescript libraries (cross-cutting concerns, domain libraries, core and foundational libraries)
- all tooling (Schematics)
- documentation
- linting
- testing
- build

This environment allows a team to establish, monitor and improve patterns and practices as a collective. This does not mean the removal of specialization and/or creativity within the team or individual developers - however, it provides increased transparency, knowledge sharing, workflow and collaboration for the many in a single work environment.

A good example for my current team is code formatting. This may seem like a trivial thing. However, HTML formatting became very important in regard to internationalization (i18n). We were able to use tools like Prettier - with a single configuration that applies to all applications and libraries in our Workspace. We also added a pre-commit hook to format our Typescript files before commits to our repository. This level of consistency and maintainability is very impressive.

## Angular Workspace Improves Developer Efficiency

When you are working on any project whether it is software, home, automobile, or other hobbies; the setup and breakdown of your work environment takes a lot of time.

- getting the tools together in a single location
- clearing out a space for work

- retrieving and putting things away
- cleaning up after the work

Although important, the setup and breakdown of a project is considered waste or non-productive time, because you are not building features. To solve this problem, a monorepo allows you to focus on a specific project - not setting up an environment for each new project.

There is a lot of overhead and maintenance of a project beyond the actual domain-specific implementation. Before Angular 6, if you had 3, 5, or 10 Angular applications, you had as many Angular projects. Depending on when they were created, they may vary with different versions of Angular, 3rd-party packages, or Typescript. Keeping all of your projects in-sync with version updates is/was a time consuming process. I have seen older projects just left alone without any upgrades.

The new Angular Workspace solves this problem. The workspace is a single work environment where all projects share the same runtime and development dependencies (packages). They all use the same version of dependencies and they all share the same package.json version. When you upgrade the workspace, all projects are updated at the same time. If there are any problems or issues, you have early notification during the build and serve of your applications.

No project left behind.

Instead of setting up and managing each project independently, you have a single-source of truth and place to manage all of your Angular projects.

If you develop custom Angular libraries and publish them as packages, you doing a very noble thing. Sharing and reusing code is very effective. However, the process to build, publish, and consume library updates is not a very efficient process, right? There are many steps. There are even more steps if you have libraries that have dependencies on other custom libraries. The Angular Workspace solves this problem with the new project type: **library**.

For more information on this topic, please read my blog post: [Monorepo + Angular Packaged Libs :: You Can Have Your Cake and Eat it Too!!<sup>2</sup>](#).

---

<sup>2</sup><https://medium.com/@angularlicious/monorepo-angular-packaged-libraries-you-can-have-your-cake-and-eat-it-too-8c5687c4ffe9>

# Workspace: A Container for Projects

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Angular.json

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Applications

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

### Application Configuration (angular.json)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

### Application Module (AppModule)

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

### Application Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

### Shared Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.



## Core Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Site Module

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Feature Modules

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Routes

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Components

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Services

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Models

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Application Routes

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Cross-Cutting Concerns

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Custom Framework Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Business Actions

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Rule Engine

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Feature Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Component Libraries

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Common

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## API Response Model

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Anatomy of an Angular Application

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Share and Reuse Code

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Why Share Code?

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## How Can We Do It?

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# JavaScript Packages

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Not Effective?

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Nx Dev Tools

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Upgrade Existing Angular Workspace

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Getting Started - New Workspace

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Create a New Workspace

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Code Organization

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

## Architectural Patterns

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Library Project Types

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.



# Benefits of a Monorepo Workspace

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.

# Resources

This content is not available in the sample book. The book can be purchased on Leanpub at <http://leanpub.com/angular-workspace-the-ultimate-guide>.