### [Course Overview](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc)

[Hi, my name's Mark Heath, and welcome to my course, Dapr 1 Fundamentals.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=4.34) [I'm a software architect working at NICE.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=11.29) [One of the things I really like about Dapr is that it has](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=14.54) [the potential to greatly accelerate and simplify the](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=18.4) [process of developing microservices.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=22.11) [In this course,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=25.61) [we're going to learn about several of the key](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=26.42) [building blocks that Dapr provides.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=29.01) [We'll see how Dapr helps us with state management, service invocation,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=31.94) [pub sub messaging,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=36.94) [managing secrets, and dealing with the challenge of](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=38.54) [observability in a microservices application.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=41.96) [By the end of this course,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=46.44) [you'll know how to add the Dapr building blocks to your own microservices.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=47.65) [As part of this course,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=52.84) [I'll be demonstrating how to install a Dapr application onto](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=54.05) [Kubernetes, and so if you have some prior experience with](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=58.3) [Kubernetes, that will be useful,](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=61.81) [but I'll also be showing you how to use Dapr without Kubernetes. I](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=64.54) [hope you'll join me in this journey to learn about Dapr with the Dapr 1 Fundamentals course, here at Pluralsight.](https://app.pluralsight.com/course-player?clipId=53c9ecda-722d-4cad-8a4f-5e9c0a9db0fc&startTime=69.57)

## [Getting Started with Dapr](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07)

### [Course Introduction](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07)

[Hi, my name's Mark Heath, and welcome to the Dapr 1 Fundamentals course.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=1.14) [In this course,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=6.39) [we'll learn about how Dapr can help us to build distributed applications.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=7.7) [Over recent years,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=13.32) [microservice architectures have established themselves as one of](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=15.25) [the primary approaches to building modern,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=19.63) [scalable, cloud‑native applications.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=22.85) [They provide an elegant solution to the problems of scale and maintainability](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=26.24) [that a more traditional monolithic architecture can run into.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=30.78) [But adopting microservices comes with some considerable challenges,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=35.94) [as there's a lot of capabilities and infrastructure requirements](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=40.24) [that you're going to need to implement in order to successfully](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=44.79) [build and run a microservices application.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=48.14) [And although there are many excellent existing runtimes and](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=52.24) [frameworks specifically designed to help simplify the](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=55.94) [process of microservice development,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=59.62) [there's still plenty of room for improvement.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=62.64) [And that's where Dapr comes in.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=65.43) [Dapr is an abbreviation of the phrase distributed application runtime,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=68.54) [and it takes an innovative approach to provide many of the building blocks](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=73.71) [that you need to create a microservices application.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=78.84) [On its official website, which you can find at dapr.io,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=82.59) [Dapr describes itself as "APIs for building portable and reliable](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=87.94) [microservices." Dapr isn't intended as a replacement for other](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=94.02) [frameworks and runtimes that you might already be using.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=99.02) [In fact, it can work well alongside them.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=102.78) [And Dapr is not language specific, so whatever programming language you use,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=105.78) [you can make use of Dapr.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=111.69) [In this course, I will be using C#, which is the language I'm most familiar with,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=113.78) [but we're going to be mostly focusing on the capabilities of](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=119.84) [Dapr that you can apply to any language.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=122.81) [Now,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=126.54) [I'm not going to spend any time in this course](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=126.92) [arguing for why you should use Dapr,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=129.25) [and that's because there's also a Dapr 1 big picture course here](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=132.44) [on Pluralsight that gives a high‑level introduction to what Dapr](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=136.59) [is and why you want to use it.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=140.86) [And I can highly recommend that you watch that course in order to](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=143.94) [get the background that you need to fully understand everything](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=147.04) [that we'll be doing in this course.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=150.3) [But I also think that the demos you're going to see in](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=153.24) [this course will speak for themselves, and as you see Dapr in action,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=155.77) [you'll begin to understand the value that it could add to](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=160.3) [your own microservice applications.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=163.49) [In this module, you'll learn how to get started with Dapr.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=165.8) [At its heart,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=170.43) [Dapr is a collection of building blocks that help you](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=171.87) [to build distributed applications,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=175.5) [and Dapr makes use of what it calls sidecars to implement those building blocks.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=177.79) [So I'll explain what Dapr sidecars are,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=184.24) [and we'll also learn about the two modes in which Dapr can run in,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=186.88) [self‑hosted and Kubernetes, both of which we'll be using in this course.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=191.24) [I'll also show you how to install the Dapr CLI and the Dapr](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=196.63) [runtime, and you'll want to do this if you're planning to](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=201.62) [follow along with the demos. And finally,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=204.44) [I'll introduce you to the GloboTicket demo application,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=207.58) [which we're going to be using to illustrate the benefits of](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=211.25) [using Dapr in a microservices application.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=214.5) [Later on in the rest of this course,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=218.44) [we're going to be making use of several of the Dapr building](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=220.56) [blocks within our GloboTicket demo application.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=223.7) [We'll see how to store state, how to perform service‑to‑service invocation,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=227.84) [how to communicate asynchronously with pub‑sub messaging,](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=233.64) [how to access secrets securely, how bindings can simplify integration with](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=237.84) [external services, and how the observability capabilities that are built](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=243.23) [into Dapr can give us powerful insights into what's going on in our](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=248.35) [microservices application. And I'll also show you how to run a Dapr‑enabled application on Kubernetes.](https://app.pluralsight.com/course-player?clipId=a59cd054-bfc0-4792-87dc-48a95bf48c07&startTime=253.13)

### [Version Check](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57)

[Let's quickly take a look at the versions of software](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=1.24) [that this course is applicable.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=4.6) [To this course was made using this version of Dapr, version 1.5,](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=5.96) [which was the latest version at the time of recording, and the](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=11.99) [demo application is using .NET version 6.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=15.73) [So if you'd like to build and run the demo code, then you will](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=19.64) [need to have version 6 of the .NET SDK installed.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=23.15) [Of course, Dapr itself doesn't require .NET 6.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=27.44) [You can use any version of .NET with Dapr. And in fact,](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=31.17) [Dapr isn't a .NET‑specific technology. It's actually written in Go,](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=34.92) [and you can use it from any programming language.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=39.58) [New versions of Dapr are released fairly regularly, and here you](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=43.54) [can see the versions of Dapr that the material in this course](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=47.87) [applies to. And if at the time that you're watching this there's a](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=51.23) [newer version of Dapr available,](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=55.53) [then it's very likely that the demo application will be](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=57.73) [able to work on that version as well.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=60.63) [And the reason I say that is because the Dapr development team take](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=63.44) [great care to avoid making breaking API changes when they release new](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=67.25) [versions, and this means that all of the techniques that I'm showing](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=71.95) [in this course should continue to work even when new capabilities are added to Dapr.](https://app.pluralsight.com/course-player?clipId=a38ff2d9-249b-4b47-acdc-9e61a7585e57&startTime=75.51)

### [Dapr Building Blocks](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642)

[In essence, Dapr is a collection of what it calls building blocks,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=1.04) [and each of these building blocks implements a capability that is commonly](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=5.88) [needed when you're building a distributed application.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=11.14) [And you can see the current list of Dapr building](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=14.76) [blocks here at the docs.dapr.io website.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=17.95) [And this is the official Dapr documentation site,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=22.84) [and this site will be extremely valuable to you as you build Dapr applications,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=26.12) [so make sure you bookmark this and take some time exploring it.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=31.32) [The building blocks that we're going to be using in this course are](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=36.38) [the service‑to‑service invocation, state management,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=39.95) [publish and subscribe, resource bindings,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=43.74) [observability, and secrets building blocks. And these building](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=47.14) [blocks greatly simplify several of the most common tasks involved](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=51.97) [with creating a microservices application.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=56.9) [And you can also see here that there are other building blocks,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=60.74) [such as one that lets you use a virtual actor programming model, and](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=63.57) [there's a configuration building block, which, at the time I'm](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=67.94) [recording this, is still in a preview state.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=70.92) [Now,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=74.34) [the great thing about Dapr building blocks is that](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=74.58) [they're all completely optional.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=76.84) [If you've already got something that implements state management or](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=79.06) [pub‑sub messaging and you prefer to use that,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=82.79) [then that's absolutely fine.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=85.75) [With Dapr,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=87.87) [you can simply choose the building blocks that are going](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=88.68) [to be most useful for your application.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=91.59) [And this means that Dapr can be very easily introduced into an](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=94.64) [existing microservices application, as well as being a great way to](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=98.01) [significantly reduce the amount of work involved in creating a](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=103.17) [brand‑new microservices application.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=107.12) [Many of the Dapr building blocks take advantage of](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=110.94) [third‑party services, and by that,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=113.72) [I mean that the Dapr building blocks are exposing capabilities of very](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=116.83) [popular industry‑standard services such as Redis,](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=122.12) [for example. And Dapr actually gives you the freedom to pick between](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=125.51) [different underlying implementations of each building block, and you](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=130.11) [can even switch between them easily.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=134.42) [We're going to be taking advantage of this capability in our demo](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=136.74) [application to use different building block implementations, depending on whether we're running locally or in production.](https://app.pluralsight.com/course-player?clipId=78813c42-cc21-4907-938f-6cfac1de5642&startTime=139.98)

### [Dapr Sidecars](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b)

[I've said that Dapr consists of a number of building blocks,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=0.94) [but let's talk a bit now about how those building blocks are implemented,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=5.05) [because Dapr takes quite an innovative approach,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=9.64) [compared to many of the other frameworks that you might be used to.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=12.61) [Dapr isn't actually implemented as a component or a library](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=17.04) [that you load into your microservice process.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=21.1) [Instead,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=24.57) [Dapr actually runs alongside your microservice as what's called a sidecar,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=25.43) [and this sidecar is a separate process.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=31.64) [There's one Dapr sidecar for each one of your microservices.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=34.54) [And when your application wants to make use of any of the Dapr building blocks,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=38.38) [it does so by making a straightforward network request to its sidecar.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=44.14) [Let me explain this with a couple of simple examples.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=49.84) [In this diagram, I've got a simple microservice,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=52.98) [and it wants to make use of the Dapr state store](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=56.4) [building block to store some state.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=60.21) [And the way our microservice does that is it makes a](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=62.84) [network call to the Dapr sidecar process,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=66.6) [and then the Dapr sidecar uses a component definition file](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=70.24) [which tells it where it's actually going to go in order to](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=73.99) [store and retrieve the state.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=77.53) [And in this example,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=79.94) [the Dapr building block is configured to store the state in Redis.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=81.21) [Or let's consider a slightly more complicated example](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=86.54) [of service‑to‑service invocation.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=89.57) [Imagine that we have two microservices,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=92.64) [and Service A wants to talk to Service B. How Dapr allows us to do](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=94.74) [this is that Service A simply talks to its sidecar and tells it that](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=100.84) [it wants to invoke a method on Service B.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=106.01) [And then the sidecar from Service A talks to the sidecar for Service B,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=109.64) [which then forwards the request onto Service B. And then the response from](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=114.84) [Service B flows all the way back through the sidecars to Service A. And there's](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=120.49) [actually several benefits to this approach, because the Dapr sidecars are able](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=125.97) [to implement additional capabilities for us.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=130.6) [First of all,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=134.84) [the Dapr building block can automatically retry if it](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=135.68) [encounters any transient network issues.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=139.32) [Second,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=142.74) [it can also automatically encrypt the traffic between](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=143.41) [the microservices using mutual TLS, including automatic certificate rollover,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=146.81) [which is something that can be very challenging if](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=153.07) [you try to implement it yourself.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=155.54) [Another security benefit is that Dapr can enforce access control](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=158.34) [policies that ensure that only the services that are allowed to](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=162.78) [talk to each other can actually do so.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=166.39) [And this approach also allows the Dapr sidecar to capture traces and metrics,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=169.34) [which can provide us additional insights and diagnostics](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=174.88) [about what's going on with our microservices.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=178.59) [And because all of these benefits are implemented inside the sidecar,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=182.54) [it keeps your microservice code very simple and straightforward.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=186.73) [And one of the key benefits of this sidecar approach is that it](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=191.34) [means that Dapr can be language agnostic.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=195.04) [It really doesn't matter at all what language your](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=198.02) [microservice is written in; as long as you can call the](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=200.56) [Dapr sidecar with an HTTP request,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=204.04) [you're able to take advantage of all of the Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=207.14) [Now, you might be a little bit concerned about the performance](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=211.44) [implications of inserting sidecars into the communication between](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=214.24) [services, and the good news is that Dapr has been extensively optimized](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=218.32) [to keep any performance impact to a minimum.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=223.25) [And it's also worth noting that Dapr makes use of gRPC,](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=226.44) [which is a protocol that's been optimized for performance. And the Dapr sidecars use gRPC for their communications between each other.](https://app.pluralsight.com/course-player?clipId=6f5f384f-d262-4b2d-ad9d-e33ad068192b&startTime=230.28)

### [Self Hosted and Kubernetes Modes](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d)

[There are two main ways in which you can run Dapr distributed applications.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=1.24) [The first is known as self‑hosted.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=6.84) [In self‑hosted mode,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=10.04) [you take responsibility for running a Dapr sidecar process](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=11.36) [alongside each of your microservices,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=15.75) [and the sidecar runs on the same machine as the microservice.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=18.32) [And the Dapr command‑line tool includes the dapr run command that you can use](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=22.6) [to easily launch your application along with a sidecar.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=28.71) [Now, obviously, any dependencies for the building blocks, such as Redis,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=33.54) [for example,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=37.9) [to support the state store or pub‑sub building blocks do need to be running](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=38.96) [somewhere and available for the Dapr sidecar to access.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=43.5) [And for local development,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=47.2) [it's quite common for those dependencies to be run as Docker containers,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=49.15) [and I'll be showing a demo of how to set that up later on.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=53.23) [And self‑hosted mode is very often used for local development,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=57.74) [although it's perfectly valid to use it in production,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=62.04) [for example, if you've chosen not to containerize your microservices.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=64.87) [The second way to run your Dapr application is on a Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=69.19) [and this would be a very common choice for production scenarios,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=75.34) [but again,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=79.24) [there's no reason why you can't also use this as an option for](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=79.73) [local development and testing if your development team are](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=83.24) [comfortable with working with Kubernetes.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=86.43) [When you deploy a Dapr application to Kubernetes,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=89.12) [you first install Dapr onto the cluster.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=93.09) [Once you've done this,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=96.64) [any microservices that you install need to be annotated in a special](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=97.79) [way that causes the Dapr runtime to automatically create a Dapr sidecar](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=102.27) [alongside each of your microservice containers, and this means that](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=108.64) [your Kubernetes Pod will contain both your microservice container and a](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=113.46) [Dapr sidecar container.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=118.85) [And if you've used service meshes in Kubernetes before, you'll recognize this](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=121.14) [kind of approach of using sidecars. And in this course,](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=125.91) [I'm going to be showing you how we can run the demo application in both modes, self‑hosted and on Kubernetes.](https://app.pluralsight.com/course-player?clipId=31bea28d-29e9-4d59-8b81-d76a27125d4d&startTime=130.4)

### [Demo: Installing the Dapr CLI](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2)

[In this demo, we'll see how to install the Dapr CLI.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=1.24) [Here we are on the official Dapr website at dapr.io,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=6.33) [and if I follow the Get Started link, it takes us to the documentation,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=10.54) [which includes up‑to‑date installation instructions for a variety of platforms.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=15.2) [If I look at the instructions for installing the Dapr CLI,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=22.14) [it shows me the commands necessary for Linux,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=25.58) [Windows, or macOS.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=28.92) [I'm on Windows,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=31.54) [so I'm going to copy the Windows install command to my clipboard, and I'm](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=32.56) [going to run it inside an administrative command prompt.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=37.6) [When I run this,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=41.94) [it downloads the Dapr CLI tool and updates my path to make that tool available.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=42.77) [And so if I open a new command prompt and enter the dapr ‑v command, here I](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=49.14) [can see the version that I've got currently installed, and it's showing that](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=55.65) [the runtime version is not applicable because we haven't yet installed the](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=60.34) [Dapr runtime. We're going to be doing that in a moment. And if I just enter](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=64.6) [the dapr command on its own,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=69.49) [this is going to give me a summary of the](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=71.53) [subcommands that we have at our disposal.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=73.59) [Now, one quick note for Windows users,](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=76.74) [if you're trying to install the Dapr CLI, you may need to run](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=78.97) [Set‑ExecutionPolicy RemoteSigned with the scope of CurrentUser in order to](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=82.82) [allow PowerShell to actually run the installation script. And another quick](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=88.79) [tip for you, if you've already installed an earlier version of the Dapr CLI](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=93.87) [and you'd like to upgrade, then the approach I tend to take to do that is](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=98.34) [just to run the install script again, and that upgrades you to the latest version.](https://app.pluralsight.com/course-player?clipId=ab55564b-5735-462d-b35d-092fabda97a2&startTime=102.35)

### [Installing the Dapr Runtime in Self-hosted Mode](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8)

[In this demo, we'll install the Dapr runtime in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=1.44) [Again,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=6.84) [the most up‑to‑date instructions for this can be](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=7.48) [found on the official Dapr website, which I'm showing here.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=10.24) [And one important thing to note is that by default, installing](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=14.64) [the Dapr runtime locally does expect that you've got Docker](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=18.75) [already installed on your machine.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=22.62) [If you don't have Docker installed,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=25.24) [then I recommend downloading Docker Desktop for Windows or Mac,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=27.02) [although, please note that you may need to purchase a license for Docker Desktop,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=31.06) [depending on the size of your company.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=35.86) [Now, on my machine, I've got Docker Desktop for Windows installed,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=38.44) [and I've set it to the Linux container mode,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=42.07) [so I'm ready to install the Dapr runtime.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=45.19) [Here I'm in an elevated command prompt, and I'll enter the dapr init command,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=48.26) [and this might take a couple of minutes,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=55.04) [as it needs to download some container images.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=56.86) [But once it's completed,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=60.12) [our machine is ready to run Dapr applications in](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=62.02) [self‑hosted mode. There are three Docker containers that](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=66.24) [will have been started by this command,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=70.15) [and we can see them if we run the docker ps command.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=72.15) [There's a Redis container, which is used for the state store and pub‑sub](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=76.74) [messaging Dapr building blocks; there's a Zipkin container, which is used for](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=81.14) [observability; and there's also a dapr\_placement container,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=86.3) [which is there in case you want to use the actor building block. By the way,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=90.86) [it's not essential that you have these containers running to use](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=95.76) [Dapr in self‑hosted mode. You're completely free to define your](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=99.24) [own Dapr components that point to services that aren't running](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=103.14) [locally as Docker containers.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=106.7) [Another thing that happened when we ran dapr init is that](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=109.34) [it created a default components folder,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=112.76) [which can be found in the .dapr folder in my user profile.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=115.73) [So if I look inside that folder on my computer,](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=120.84) [you can see that there are two YAML files, and these define](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=123.8) [the state and pub‑sub Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=128.19) [And these default components make it even easier for us](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=132.14) [to get started with local development.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=135.5) [And both of these components make use of that Dapr Redis container that got created for us with the dapr init command.](https://app.pluralsight.com/course-player?clipId=55c8ec96-101f-463a-a6e9-a0ca2ba17fb8&startTime=138.84)

### [Demo: Installing the Dapr Runtime on Kubernetes](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0)

[Let me also quickly show you how we can install the Dapr](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=1.34) [runtime onto a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=4.82) [Now, before you do this,](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=7.94) [you do need to have set your kubectl context to point to the](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=9.18) [Kubernetes cluster that you want to install Dapr onto.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=13.63) [And assuming that you're familiar with Kubernetes](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=17.18) [and the kubectl command‑line tool, then you will know how to do that.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=20.13) [But once you've done that,](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=24.36) [you can install the Dapr runtime with the dapr init ‑k command.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=26.19) [So here I am at a command prompt, and I've currently got a](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=32.34) [Kubernetes cluster that's running in Azure that my kubectl context](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=35.99) [is pointing at, and so if I run dapr init ‑k,](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=40.68) [then we can see that it deploys the Dapr runtime into my](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=45.84) [Kubernetes cluster. And once that's completed, it prompts me to](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=49.84) [verify it with the dapr status ‑k command.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=54.42) [And so here we can see that it lists all the containers that have](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=58.94) [been created as part of the Dapr control plane installation on](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=62.3) [Kubernetes. Now, they've not all started up yet,](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=67.18) [but if I wait a while and issue this command again,](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=71.05) [we can see that all of these containers are up and running on my](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=74.59) [Kubernetes cluster. By the way, if you've already initialized a previous](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=77.82) [version of Dapr and you want to upgrade to the latest version, then, in](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=83.23) [self‑hosted mode you run dapr uninstall ‑‑all and then follow that with](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=88.46) [another call to dapr init.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=95.16) [And in Kubernetes, you can use the dapr upgrade ‑k command, passing in the](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=98.24) [desired runtime version with the ‑‑runtime‑version flag, as you can see with this example here.](https://app.pluralsight.com/course-player?clipId=f2f2ad59-369b-4932-940a-10f7e6dda6d0&startTime=103.95)

### [GloboTicket Demo Application](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8)

[Let's take a quick look at the demo application we'll](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=0.94) [be working with in this course.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=4.36) [The code is available both in the Pluralsight course download materials,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=6.46) [but you can also find it on GitHub at the markheath/globoticket‑dapr repository.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=11.52) [GloboTicket is a company that sells tickets for music and theater events.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=19.54) [They have an online website which allows customers to browse](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=25.04) [the available tickets and to place orders.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=28.43) [And they've adopted a microservices architecture,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=31.3) [enabling several software development teams to work](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=35.13) [independently on different areas of the application.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=38.29) [In this course,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=42.44) [we're going to see how Dapr building blocks can be used to](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=43.28) [simplify many of the microservices development tasks that the](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=47.03) [GloboTicket development team are going to need.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=51.14) [We'll see how Dapr can be added to the existing GloboTicket microservices,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=54.94) [allowing them to take advantage of Dapr's capabilities incrementally.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=60.41) [And to keep things as simple as possible,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=64.86) [our demo scenario is just going to have three microservices.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=67.65) [By the way, although I'll talk a lot about microservices in this course,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=71.85) [Dapr doesn't assume that you're necessarily using a microservices architecture,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=77.72) [so it calls the services applications,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=83.04) [with each application having its own sidecar and having a](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=86.35) [unique name, which is known as the app id.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=90.73) [Let's look at the GloboTicket microservices.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=94.54) [First is the front‑end shop website.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=97.64) [This is an ASP.NET Core website,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=101.64) [and this is the website that GloboTicket customers visit to](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=104.46) [browse the available tickets and place orders.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=108.79) [The second is the event catalog microservice.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=113.14) [The front‑end microservice calls the catalog microservice whenever it](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=115.92) [needs to retrieve the list of available tickets.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=121.03) [Third is the ordering microservice.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=124.75) [This service has the responsibility of performing the tasks](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=127.48) [that need to be done after an order is placed,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=131.73) [such as emailing the customer with an order confirmation message.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=135.04) [Let's see how we'll be introducing the Dapr building blocks into](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=140.44) [the GloboTicket architecture in this course.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=144.6) [For the communications between the front‑end website](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=147.64) [and the event catalog microservice,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=151.37) [we'll be using the service‑to‑service invocation building block,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=153.95) [which gives us service discovery, retries,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=158.14) [and encrypted communication.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=161.37) [The event catalog microservice needs a connection](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=164.14) [string in order to access a database,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=167.21) [and so we'll see how the secret building block enables](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=169.71) [us to fetch that connection string.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=173.76) [The front‑end website allows customers to put items into a shopping basket.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=176.64) [In the future,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=182.06) [GloboTicket would like to create a dedicated shopping basket microservice,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=183.54) [but for this first version,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=188.15) [we're simply going to use the Dapr state store building block directly,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=190.2) [which gives us a really easy way to save and load the](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=194.86) [contents of users' shopping baskets.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=197.92) [When an order is placed, we're going to use the pub‑sub messaging building block,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=200.97) [which will allow the front‑end website to publish a message and the](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=206.86) [ordering microservice to subscribe to it. And this keeps the](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=211.03) [ordering microservice decoupled, making for a more resilient](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=215.56) [architecture, and it also keeps the customers' online shopping](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=219.27) [experience as responsive as possible.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=223.17) [We also have a scheduled task that we want to run periodically](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=227.04) [against the catalog microservice, and we're going to use the cron](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=230.95) [input binding to call an endpoint on a schedule. And the ordering](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=235.26) [microservice needs to send an email,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=240.76) [so we'll use a Dapr output binding to send an email to an SMTP server.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=243.34) [Finally,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=249.54) [we'll see how the Dapr observability features enable us to](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=250.14) [understand the flow of calls between our microservices.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=254.12) [We'll be making use of the C# Dapr SDK to implement these features.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=259.04) [Although you don't need to use the Dapr SDK, since you can](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=264.13) [communicate with the Dapr sidecars by just using regular HTTP](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=268.31) [requests, the Dapr SDKs simplify this task and ensure that we get the](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=272.72) [URLs and the request payloads right.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=278.19) [If you download the demo code,](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=281.94) [you'll be able to build and run the GloboTicket demo application, either in](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=283.62) [self‑hosted mode or on Kubernetes, so feel free to do that.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=288.59) [Or you might actually prefer instead to take a microservices application](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=294.1) [of your own, and as you follow along with this course, try to add each of the Dapr building blocks into that application.](https://app.pluralsight.com/course-player?clipId=fe947745-624d-439a-84ea-d9b520e90ca8&startTime=298.86)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6)

[In this module,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=0.94) [we've learned about some of the key concepts in Dapr and seen how we can](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=1.76) [set up a development machine in order to start using it, by installing the](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=6.26) [Dapr CLI and initializing the runtime. We've learned about many of the key](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=10.99) [building blocks that Dapr provides, including service‑to‑service invocation,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=17.07) [state management, and pub‑sub messaging.](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=22.1) [And each of the upcoming modules in this course is going to](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=25.64) [demonstrate the use of one of those building blocks.](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=28.84) [We saw that the Dapr architecture makes use of sidecars,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=32.94) [which are processes that sit alongside each microservice. And we also](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=37.03) [learned about the two modes that Dapr can run in,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=42.03) [known as self‑hosted,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=44.93) [which is quite commonly used for local development, and also running on](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=46.67) [Kubernetes. And I introduced the GloboTicket demo application that we'll be](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=50.83) [using throughout this course to put into practice the Dapr concepts we're](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=56.65) [learning about. In the next module,](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=61.27) [we'll actually get started using Dapr by using the state management building block to implement the shopping basket feature.](https://app.pluralsight.com/course-player?clipId=14bb4309-12ea-484d-a741-80fd84f7d6b6&startTime=63.84)

## [Implementing State Management](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8)

[Hi, my name's Mark Heath, and in this module,](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=1.14) [we're going to use our first Dapr building block and see how Dapr can help us](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=4.24) [implement state management for our microservices in a simple and elegant way.](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=9.64) [We're going to see what capabilities the Dapr state management building block](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=15.33) [has to offer and why we might want to use it, instead of some of the other](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=20.35) [options available to us.](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=24.82) [As this is the first Dapr building block that we're going to cover,](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=26.94) [we'll also look at the concept of Dapr components. We'll see how to](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=30.3) [configure a state store and how the Dapr components model allows us to](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=35.66) [seamlessly switch out the underlying back‑end store, depending on what](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=40.58) [environment we're running in.](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=45.81) [We'll look at the URLs that we can call on the Dapr sidecar to](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=48.14) [access the state management building block easily,](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=52.33) [no matter what programming language we're using. I](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=55.7) [also want to introduce the Dapr SDKs,](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=59.3) [which make it even easier to access the Dapr building](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=62.44) [blocks from your language of choice.](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=65.63) [And in this course, we're going to be using the .NET Dapr SDK. And we'll see](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=68.44) [the Dapr state store building block in action as we implement a shopping basket feature in our GloboTicket demo application.](https://app.pluralsight.com/course-player?clipId=d53ff7c7-44b0-476f-a430-e35949bbebe8&startTime=74.1)

### [The Dapr State Management Building Block](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621)

[The Dapr state management building block is designed for those situations where](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=1.24) [your microservices need to store state as key‑value pairs.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=6.45) [It's extremely common for services in a distributed application](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=11.84) [to have some state that they need to store,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=15.69) [and often,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=18.11) [you need to share that state between multiple instances of the](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=19.31) [same service running on different machines.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=23.28) [And this means that you need a shared store that](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=26.84) [all of the instances can access.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=29.78) [Now, of course, many microservices will already have a database,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=32.74) [perhaps a relational or a document database,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=37.26) [that they can use to store state, and if that's the case,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=39.79) [then, of course, you can use that.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=43.85) [Dapr doesn't force any of its building blocks on you,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=46.2) [and if you've already got a solution that you're happy with,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=49.77) [then that's fine.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=52.6) [But the particular type of data that the Dapr state management](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=54.74) [building block helps you to manage is key‑value pairs,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=58.06) [and this might include things like temporary data,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=61.78) [user preferences, configuration settings,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=65.05) [or the status of currently logged‑in users.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=68.36) [And this sort of data often doesn't warrant the additional cost and](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=72.44) [complexity of being stored in a traditional database.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=76.19) [And a better choice might be to store it in a cache like Redis.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=80.05) [And what we want for this sort of data store is for it to be as simple,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=84.72) [cheap, and fast as possible.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=89.33) [The example we're going to be using in this module is to store the contents](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=92.44) [of the users' shopping basket for our GloboTicket store.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=96.75) [Shopping baskets do need to be persisted short term,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=100.94) [but they don't necessarily have the same high business value as the](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=104.54) [actual orders that are placed in the GloboTicket store,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=108.6) [and so having a nice,](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=111.8) [easy to use key‑value state store to keep those shopping baskets in is quite an attractive option.](https://app.pluralsight.com/course-player?clipId=a1f0e297-5e13-46ec-a98a-23ccbd0f9621&startTime=113.37)

### [Benefits of Dapr State Management](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537)

[You might be thinking, well,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=1.14) [I understand why I might want to use a key‑value state store like Redis,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=2.79) [but why don't I just use it directly?](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=7.73) [And of course you're free to do that,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=10.24) [but let me mention a few of the benefits of using the Dapr](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=12.03) [state management building block in order to take advantage](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=15.87) [of a key‑value store like Redis.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=19.61) [First,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=22.33) [the Dapr state management building block supports swappable back‑end stores.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=22.98) [The actual state can be stored in not just Redis,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=29.44) [but MongoDB or Azure Blob Storage or AWS DynamoDB.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=32.76) [There's a whole host of supported state stores. And](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=38.74) [here on the Dapr docs website, there's a list of all of the available options,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=42.02) [and as you can see there's a really good amount of choice.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=46.65) [The ability to switch between back‑end stores is particularly](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=50.88) [useful if you want to use one back‑end store for local development](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=54.45) [but a different one when you're in production. You can also](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=58.42) [configure multiple state stores.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=62.54) [This allows different types of state to be stored in the most](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=64.9) [appropriate back‑end store but doesn't require the application](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=68.32) [developer to have to learn how to use the SDKs of each one of](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=71.98) [those different back‑end stores.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=76.56) [It's also very easy to use, thanks to a simple API. You just issue HTTP](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=79.14) [requests to the Dapr sidecar in order to get and set state. By default,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=85.12) [the state is isolated to the microservice, or what Dapr calls the](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=90.69) [application, that sets the state. But you can also configure a Dapr](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=96.01) [state store that can share state between multiple applications. And in](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=101.04) [addition to basic getting and setting of state by key, the Dapr state](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=107.45) [management building block offers several useful added value capabilities,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=112.72) [such as control over the time to live, concurrency control using](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=117.94) [ETags, support for querying and bulk operations,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=122.6) [and automatic state encryption with key rotation.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=127.16) [So as you can see,](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=131.94) [Dapr offers a very powerful and flexible state management capability that's easy to use.](https://app.pluralsight.com/course-player?clipId=93a488d9-de0b-41ae-861a-1ec575e52537&startTime=132.99)

### [Demo: Using the State Management Building Block](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a)

[Let's actually see the Dapr state management building block in](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=1.24) [action by accessing it directly with HTTP requests.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=5.11) [Here I am in a command prompt,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=9.9) [and because I've installed the Dapr CLI and I've initialized](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=12.7) [Dapr in self‑hosted mode on this computer,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=17.17) [I can start a Dapr sidecar with the dapr run command,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=20.74) [and I'm using the dapr‑http‑port switch to say that we want](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=24.91) [the Dapr sidecar to listen on port 3500.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=30.42) [Normally, we'd pass several additional arguments to dapr run,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=35.34) [and we're going to be seeing those later,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=38.87) [but for now, we're simply running a Dapr sidecar on its own.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=40.79) [You can see here that there's a log message saying that](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=46.04) [the Dapr sidecar is up and running.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=48.53) [Let's make some HTTP requests to this Dapr sidecar.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=51.94) [And to make it easy for me to show you issuing HTTP requests,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=56.54) [I've decided to use this Visual Studio Code extension called REST client,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=61.19) [and this just allows me to show you in text format the URLs we're calling](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=66.65) [and any payloads of the requests and see the response.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=72.04) [So the first one is a POST to localhost port 3500,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=76.74) [which, if you remember,](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=81.95) [is the port that the sidecar is listening on. And we're going to](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=82.91) [call the v1.0/state/statestore endpoint. I'll be explaining more](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=86.93) [about how this URL works in a little bit.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=93.54) [The body is a JSON array of objects with keys and values.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=97.74) [We're only setting one state value here with a key of color and a value of red.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=103.64) [I'll send this request, and we can see that it's been accepted with a 204](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=109.64) [response code. And now, I'll issue a GET request to the same URL but with](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=114.27) [the key name of color appended to the end.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=120.79) [As you can see, we get a 200 OK response, and the body](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=124.64) [contains the value of the state, which was red.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=128.59) [Now, at this point, you might be wondering where this state is actually](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=132.54) [being stored. How did the Dapr sidecar know what back‑end store it](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=135.92) [should use? And to answer that question, we need to learn about components, so let's talk about those next.](https://app.pluralsight.com/course-player?clipId=398b5ccf-54ca-46de-83fd-dbc586be671a&startTime=140.77)

### [Dapr Component Definitions](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19)

[How did the Dapr sidecar know that we wanted our state to be stored in](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=1.34) [the Redis Docker container that's running locally?](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=6.01) [Well, when we installed Dapr in self‑hosted mode,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=9.94) [some global components were set up for us.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=13.48) [I can find those global component definitions on Windows in my user](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=16.84) [profile directory under a folder called .dapr.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=21.43) [Here we can see inside that folder there's a config.yaml file,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=26.14) [which includes the Dapr configuration for tracing.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=30.66) [But the thing of interest to us is this components folder.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=34.34) [If I look inside, I can see pubsub.yaml and statestore.yaml.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=38.64) [These are the two global components that automatically got set up for us.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=42.67) [Let's look inside statestore.yaml since that's the one we're using.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=50.14) [This file says that this is a Dapr component, and it has a name of statestore.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=54.84) [You can actually have multiple state stores and give each one its own name.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=61.84) [The type says that for this state store,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=66.44) [we want to use Redis as the backing store.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=69.07) [And when we installed Dapr in self‑hosted mode,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=72.54) [a Redis container was automatically created for us,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=75.37) [and this component is set up to use that Redis container,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=78.87) [which is exposed on localhost port 6379.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=82.96) [And if I run docker ps, I can see that Redis container is running on my](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=87.94) [PC and has the name dapr\_redis. Back in the configuration file, we can](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=92.44) [see that a configuration file can contain additional settings, such as](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=99.34) [the password needed to connect, which is blank for this local Redis instance.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=104.15) [Now, you might be wondering how you would know what the](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=110.04) [correct syntax of this component configuration file is and](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=112.65) [what the required values are.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=116.75) [The answer is that we can get this information from the official Dapr](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=118.74) [documentation website. So, here we are looking at the list of supported](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=122.84) [state stores again. Let's navigate into the entry for Redis. Here we can](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=128.35) [see a fuller example of the settings that we can configure for a Redis](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=133.97) [state store. You can see that we can enable TLS or configure the maximum](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=139) [number of retries. And if I scroll down, we can see additional help for](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=144.22) [each of the fields, including whether or not they're required.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=149.32) [So you could use this information to point your component configuration](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=153.74) [to a completely different Redis cluster if you wanted.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=157.72) [Or let's suppose that we wanted to use MongoDB instead. Let's navigate into](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=161.74) [the Component format page for MongoDB, and we can see the type value that](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=167.13) [we'd need to set, which is state.mongodb, and we can see the settings that](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=172.68) [are relevant for this backing store.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=178.2) [When we started Dapr with dapr run earlier,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=181.04) [we didn't specify a path to a components folder, and so it just](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=183.99) [used the globally defined Dapr components.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=188.46) [This is great because it gives you a nice convenient quick start to](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=192.24) [using Dapr. But, for real‑world applications, you'd typically](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=195.8) [create dedicated YAML component definition files that were just for](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=199.97) [that application. For example,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=204.67) [in the GloboTicket demo application that we're going to be using in this course,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=207.58) [here's the state store component definition that we use](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=211.85) [when we're running locally in self‑hosted mode. It's](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=215.49) [almost identical to the global one, but notice that the store name is different.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=219.24) [I've decided to call this state store,](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=224.44) [shopstate. And, let me also show you the one that I use when I deploy](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=226.66) [GloboTicket into Azure running on Kubernetes. Here, I'm defining that the](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=231.47) [state store called shopstate is actually using Azure Blob Storage as the](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=236.96) [backing store, and there's different metadata here that allows the Dapr](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=241.91) [sidecar to know how to connect to my Azure Blob Storage account. And we'll](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=246.07) [be seeing more of these component definition files in future demos in this course.](https://app.pluralsight.com/course-player?clipId=8672e463-5dd4-47ad-9914-aadc012f6c19&startTime=251.5)

### [Dapr State Store URLs](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505)

[Let's quickly review the URLs that we saw in the last demo,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=1.24) [remember that when we use Dapr building blocks were talking to the](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=5.84) [sidecar so it's always available on localhost,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=9.38) [whether we're in self‑hosted mode or whether we're running on Kubernetes.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=13.28) [The default port number used by a Dapr sidecar is 3500.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=18.54) [But when you're running in self‑hosted mode,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=23.84) [there are potentially many sidecars running on the same computer,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=26.54) [and so each one will need its own unique report number.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=30.94) [And you can discover the port number that your Dapr sidecar is](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=35.44) [using by inspecting the Dapr HTTP port environment variable that](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=38.66) [the Dapr runtime sets up for you.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=44.34) [The state part of this URL means that we want to use](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=47.04) [the state management building block.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=51.02) [The other Dapr building blocks use a similar URL structure,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=53.74) [but will have something different here.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=57.36) [And the v1.0 simply means that we're using version one of the state](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=60.14) [management building block API. And having a version in the URL](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=65.18) [enables Dapr to offer backwards compatibility if they ever needed to](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=69.73) [introduce a new version of the state building block API that wasn't](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=74.34) [backwards compatible.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=79) [Finally, the statestore part of this is the name of the statestore as it's](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=80.84) [defined in the component definition YAML file. For the GloboTicket examples](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=86.49) [that we just looked at, this would be set to shopstate.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=92.41) [Now, the reason I took the time to show you how we can call the Dapr](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=97.24) [Sidecar directly by constructing these URLs is that you're free to](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=101.36) [interact with Dapr simply by using whatever mechanisms you already have](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=106.12) [available to you for making HTTP requests in your programming language](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=110.79) [or framework of choice.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=115.84) [For example,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=117.84) [in ASP.NET, I could simply use the HttpClient class and call these endpoints](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=118.53) [myself and serializing my state in and out of JSON. And this is one of the great](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=124.99) [advantages of Dapr, it really does have a minimal barrier to entry and really](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=131.15) [doesn't matter what programming language you use. As long as that programming](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=136.18) [language can make HTTP requests, you can use the Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=140.03) [However,](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=146.14) [it would be nice if we didn't have to remember how to construct these](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=146.7) [URLs ourselves and do the work of JSON serialization and deserialization](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=150.52) [each time we use the Dapr building blocks. And that's where the Dapr SDKs](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=156.45) [for different languages come in. So let's talk about those next.](https://app.pluralsight.com/course-player?clipId=db80daea-07e5-4efa-945e-dc9d95ea3505&startTime=161.74)

### [Dapr Language SDKs](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87)

[Dapr provides client SDKs for several of the most popular programming languages,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=1.04) [and these simplify accessing the main Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=7.34) [This page on the Dapr Documentation site is a good place to visit to](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=12.44) [check on the current availability of Dapr SDKs.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=17.2) [At the time of recording, there are stable Dapr client SDKs available for .NET,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=21.94) [Python, Java, Go, PHP, and JavaScript, and there are others in development.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=27.82) [You'll also notice that some of the languages have server extensions available,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=35.11) [such as an ASP.NET Core extension that we'll be using later on in this course.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=40.74) [In our next demo, we're going to be using the state store in GloboTicket.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=46.59) [And, even though we could do that without using the .NET Dapr SDK,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=52.02) [we're increasingly going to see the value of using the SDK as we move](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=56.84) [on to some of the more complex building blocks.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=61.45) [So,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=64.44) [I'm going to be showing you how we can use the Dapr .NET SDK to](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=64.7) [implement the GloboTicket shopping basket.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=69.63) [But before we're ready to integrate the Dapr state store,](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=73.14) [let me show you first how we can run the GloboTicket demo application in self‑hosted mode with the dapr run command.](https://app.pluralsight.com/course-player?clipId=7b6fbcf8-d888-4eb4-a119-88857a9add87&startTime=76.76)

### [Demo: GloboTicket Dapr Run Startup Scripts](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184)

[In this demo,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=1.04) [I'll show you the scripts that I've created to start](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=1.98) [GloboTicket in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=5.43) [These scripts use the dapr run command,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=8.84) [and we'll see several of the arguments that can be](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=11.75) [used to configure the Dapr sidecar.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=14.09) [The code for the Dapr GloboTicket demo is available from GitHub](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=17.84) [at the Mark Heath GloboTicket Dapr repository,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=22.64) [and if you clone this, you'll have the full solution,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=26.54) [including all the Dapr integration code,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=29.79) [and so you can explore that in more detail and experiment with it.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=32.74) [There are also branches in that Git repository that include a](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=37.24) [version of GloboTicket before Dapr has been added,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=41.14) [so you can use that as a starting point if you want to](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=44.71) [follow along and add Dapr to it.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=47.6) [The README for the repository describes a number of ways](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=50.74) [that you can run GloboTicket locally, including with Docker Compose.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=53.96) [Docker Compose is quite an attractive option for running microservices locally,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=58.94) [but in this course, we're not going to be using Docker Compose,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=64.14) [and I'm just going to show you how we can start each of our](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=67.52) [microservices in self‑hosted mode from the command line.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=70.26) [Here we are in Visual Studio Code, and I've loaded up the demo project,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=74.94) [and you can see that there are folders for each of the three microservices,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=80.14) [frontend, catalog, and ordering.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=83.99) [For this demo,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=87.94) [we're only going to need to have the frontend and catalog services running.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=88.68) [If we look inside the frontend folder,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=93.74) [you'll see a start‑self‑hosted PowerShell script that has a dapr run](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=95.87) [command that starts up the Dapr sidecar for the frontend microservice](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=101.29) [and launches the frontend web server.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=106.31) [Let's look at the additional arguments we're passing to dapr run.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=109.64) [First, we're specifying an app‑id.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=114.34) [Each application needs a unique ID,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=117.39) [and here you would typically just use the name of the microservice.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=120.64) [This one is called frontend.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=125.14) [Second,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=127.74) [we need to tell the Dapr sidecar what port our microservice is listening on.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=128.23) [For this application it's 5266,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=134.04) [which you can find in the launchSettings.json file in the Properties folder,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=137.44) [if you're wondering where I got that value from.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=142.23) [Third, we need to specify the port that the Dapr sidecar should listen on.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=145.74) [For this application, I'm using the default Dapr port of 3500.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=151.44) [Fourth,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=157.04) [we're telling the Dapr sidecar where our component definitions can be found.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=157.71) [They're in the Dapr components folder.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=163.34) [If we look at this folder,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=166.44) [you'll see a few YAML files for various building blocks that](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=167.72) [we're going to be using throughout this course.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=171.62) [But the only one that matters for now is stateStore.yml,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=174.34) [which we've seen earlier.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=177.5) [It's using Redis as the backing store and giving it the name, shopstate.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=180.14) [Back in the dapr run command script, the final parameters are dotnet run,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=186.04) [and here, you'd put whatever command is needed to start your microservice,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=191.54) [which for .NET Core is simply the dotnet run command.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=196.64) [We also need to start the event catalog microservice.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=201.54) [So, if we look in that folder, we can see another start‑self‑hosted.ps1 file.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=205.08) [The command in here is almost exactly the same,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=211.14) [except we've got a different app id, a different app port,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=214) [the catalog service is listening on port 5016,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=218.34) [and I do need a unique port number for the Dapr sidecar to use,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=222.67) [so I've chosen 3501.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=226.89) [So, now I've shown you these dapr run startup scripts,](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=230.34) [in the next demo we'll actually run these scripts and explore the GloboTicket website.](https://app.pluralsight.com/course-player?clipId=a3b5d5bf-66a5-45df-adb3-c0b7514c2184&startTime=233.56)

### [Demo: Running GloboTicket in Self-hosted Mode](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a)

[In this demo,](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=1.04) [we're going to use the dapr run commands that we looked at previously to start](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=1.92) [up the frontend and catalog microservices in self‑hosted mode and see the](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=6.89) [problems with the current in‑memory basket implementation. In Visual Studio](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=13.14) [Code, I've opened up two terminal windows,](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=18.47) [which I've renamed to catalog and frontend, just for convenience.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=21.06) [And, in the frontend window, I've navigated into the frontend folder.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=26.54) [Let's run start‑self‑hosted.ps1 to run that dapr run command.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=31.44) [You'll see there's a fair bit of logging output from the Dapr sidecar itself,](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=37.94) [but we can also see some log messages that start with APP, and these are the log](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=42.94) [messages that are coming from our frontend microservice. And, I can see here](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=48.22) [that our microservice has started up successfully.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=53.34) [So, let's jump to our other terminal window, in which I've navigated](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=56.74) [into the catalog folder, and we'll start the event catalog service](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=60.49) [in exactly the same way with our start‑self‑hosted.ps1 script that](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=64.84) [runs the dapr run command.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=70.18) [So now these two microservices are running, let's navigate in a web browser to](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=72.54) [localhost port 5266, and we can see the frontend website. Now, notice, we're not](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=77.2) [going through the Dapr sidecar here. We are talking directly to our frontend](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=85.66) [website microservice. And the ticket details that you can see here have been](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=90.68) [retrieved from the event catalog microservice, although that communication isn't](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=95.55) [actually using Dapr yet.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=101.09) [Let's put something into the shopping basket.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=103.64) [I'll add a ticket for this concert. And, back on the home page,](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=106.64) [you can see that the basket is showing that I've](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=111.14) [currently got tickets for one concert.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=113.76) [Now, let's imagine that our frontend web server restarts at this point, and](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=117.24) [I'll simulate that by quickly stopping and restarting it.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=122.13) [And now, let's go back and we'll reload the home page. As you can](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=126.74) [see now, I've got 0 tickets in my shopping basket.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=131.18) [Well, why did that happen?](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=135.54) [Well, currently, the frontend is just using in‑memory shopping baskets,](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=137.34) [which is not really a very good practice.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=142.31) [It can't survive the restart of the service, and it also won't let us scale](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=145.14) [out because each instance of the frontend service won't know about the](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=149.78) [shopping basket contents on the other instances.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=154.3) [So let's see how the Dapr state store building block enables us to fix this in our next demo.](https://app.pluralsight.com/course-player?clipId=f4363f3e-3607-4651-8ff2-c78fe86bc95a&startTime=158.24)

### [Demo: Using the Dapr State Store Building Block](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc)

[In this demo,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=0.94) [we're going to change the shopping basket from using a simple in‑memory](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=1.91) [cache to use the Dapr state store building block. Here we are in Visual](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=6.59) [Studio Code, and the shopping basket functionality currently resides](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=11.83) [inside the frontend website. Now, maybe in the future,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=16.65) [GloboTicket would want to break this out into its own shopping basket](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=20.72) [microservice, but for now, we're just keeping things very simple.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=24.72) [The first thing we need to do is to reference the Dapr.AspNetCore NuGet package,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=29.44) [and this package references the Dapr.Client NuGet package,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=35.94) [which is the main thing we need for this demo. But, Dapr.AspNetCore adds in](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=39.98) [some additional capabilities relevant to ASP.NET Core that we're going to be](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=45.61) [using throughout this course, so it makes sense for us to reference this](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=50.24) [package. And I'm referencing the latest version of this package at the time](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=54.63) [of recording. The next thing to show you is in the Program.cs file for the](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=59.6) [frontend project.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=65.49) [And this is where all of the startup code for the web server belongs.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=67.04) [And there's only really one thing we need to do here, and that's to](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=71.54) [ensure that we call builder.Services.AddDaprClient, and this registers](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=74.49) [that Dapr client into the inversion of control container so it can be](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=80.67) [easily used in any of our services and controllers. Here you can see](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=85.15) [I've defined a simple interface that specifies the capabilities of the](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=90.75) [shopping basket.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=95.06) [You can see here that we've got methods that let us add,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=96.74) [update, or remove items from the basket.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=99.38) [Now,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=103.04) [the current implementation that the frontend microservice is using is in](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=103.18) [this InMemoryShoppingBasketService that you can see here. Here we've got](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=107.48) [some in‑memory dictionaries that are storing the shopping basket](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=112.68) [contents and they're also caching details of the events that the tickets](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=116.52) [are for. And we need that because whenever we view the shopping basket](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=120.86) [in the website,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=124.96) [we also want to see the name and date of the events that our tickets relate to.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=126.22) [So the shopping basket actually has got two types of data](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=131.14) [that it needs to temporarily store.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=134.15) [It needs to store the contents of the basket and also some](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=136.56) [details of the events that are in the basket. Here,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=139.82) [you're looking at an implementation of the IShoppingBasketService that](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=143.72) [makes use of the Dapr state store building block.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=148.21) [There's a couple of important things to note here.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=152.14) [First of all, you can see that in the constructor,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=154.8) [we're injecting an instance of DaprClient, and this is the](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=157.61) [class that we're going to use in order to access the Dapr state](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=162.11) [store building block functionality.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=165.95) [Also, I've defined a constant called stateStoreName,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=168.74) [which holds the name of the Dapr state store that we want to use,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=172.42) [which if you remember from our YAML files is shopstate.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=176.76) [Now, most of the implementation of this class is exactly the same as for](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=181.74) [the in‑memory implementation. The only difference is how it stores the](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=185.56) [basket contents and event details cache. Let's take a look at how we](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=190.22) [save a basket into the state store in the SaveBasketToStateStore method.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=195.56) [It's really very easy.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=201.94) [I just call SaveStateAsync on the daprClient, passing in the stateStoreName,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=203.74) [the key that I want to use, which I'm building from the BasketId, and the](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=210.14) [thing that I want to store as the value of that state.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=215.25) [In this example, it's a data transfer object called](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=218.84) [StateStoreBasket, and the Dapr client is automatically](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=221.98) [going to serialize this to JSON for me.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=226.18) [And there's a similar method here called SaveEventToStateStore,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=229.94) [which does the same thing, but with event details.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=233.97) [Now let's see what happens when we want to get a basket from the state](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=238.74) [store. Here in the GetBasketFromStateStore method, we simply call](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=242.54) [daprClient.GetStateAsync, specifying the stateStoreName and the key we](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=247.71) [want to retrieve, as well as the type to deserialize the state into. Most](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=253.8) [of the code here is actually dealing with what happens if the basket](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=259.8) [isn't in the state store, which prompts us to create a new basket.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=263.31) [There's a similar method called GetEventFromStatStore, and that](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=268.14) [again uses the daprClient.GetStateAsync method to fetch details](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=272.04) [of an event from the state store.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=277.39) [If it doesn't find it,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=279.84) [it makes a call to the catalog microservice to fetch the data](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=281.38) [so that it can cache it in the state store.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=285.65) [Okay, so that's the Dapr implementation of the state store,](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=289.14) [and I can quickly switch over my code to use this by changing my startup](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=293.14) [code in Program.cs to register the Dapr client state store shopping basket](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=297.94) [instead of the InMemoryShoppingBasket service.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=304.37) [And, I was also able to change this registration from AddSingleton to](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=308.24) [AddScoped because the state is no longer stored in memory, so it doesn't](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=312.65) [matter if this class is discarded and recreated.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=317.11) [So now we've seen the code changes needed to implement the Dapr state store building block, let's try this out.](https://app.pluralsight.com/course-player?clipId=b82a6ea8-f987-4b6d-a3a1-915b58f158fc&startTime=321.24)

### [Demo: Testing the Dapr State Store Building Block](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b)

[In this demo,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=1.04) [we'll test the changes that we just made to make use of the Dapr](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=1.87) [state store building block in GloboTicket.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=6.26) [First of all,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=9.94) [I'm going to do a dotnet build to ensure that we've built](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=10.64) [the changes for our frontend microservice and that](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=13.8) [everything is compiling correctly.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=16.74) [And,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=19.24) [now I'll run my PowerShell script again to start](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=19.53) [this service in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=23.13) [And, the catalog microservice is actually still running on my machine,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=26.04) [so we can try this out immediately in the browser.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=30.15) [The first time in, as expected, my shopping basket is empty.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=33.84) [So, let's put something into the shopping basket,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=37.94) [and because we're using the Dapr state store building block,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=40.76) [this now should be persisted in the Redis state store.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=44.74) [And to prove that that's the case,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=48.74) [I'm actually going to go and restart my frontend service again.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=50.25) [I'll stop it, and start it up with the PowerShell script.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=53.97) [And now, if we go back to the web page and reload,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=59.64) [we can see that our ticket is still in the basket and we can](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=63.24) [check that all of the contents are still there and correct](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=66.84) [by looking inside the basket.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=70.05) [Now, if you're interested in what's actually going on behind the scenes,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=72.74) [you can run a command inside the Redis container.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=76.38) [I've used Docker Desktop to open a bash shell into my Redis container,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=80.24) [and if I run the redis‑cli KEYS '\*' command, which lists all of the keys](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=84.69) [that are stored in this instance of Redis,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=91.39) [we can see that the keys include the state key name that we generated in code,](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=94.24) [but they're also all prepended with the application name.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=99.84) [And that's because a single state store might be](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=103.84) [holding state from multiple applications, and this protects them of accidentally interfering with each other's state.](https://app.pluralsight.com/course-player?clipId=caec07ab-ab1f-4690-904b-8ea30664c07b&startTime=106.53)

### [Demo: Using a Different State Store Component](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941)

[For our final brief demo in this module,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=1.04) [I want to quickly show you how we can use a completely different](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=3.88) [backend store for the state. And, in this example, we'll be using](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=7.61) [Azure Blob Storage. In the deploy folder, you can find the Dapr](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=12.39) [component definitions that are used when I run GloboTicket in Azure](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=17.1) [on the Azure Kubernetes Service.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=21.87) [We'll be talking more about this later, but for now, we're just](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=24.84) [going to focus on the state store part of this setup. In](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=28.04) [azure‑statestore.yaml, we can see that we're defining a state](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=32.31) [management component, which has the same name,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=36.8) [shopstate, but its type is state.azure.blobstorage. And, I've filled in the](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=39.65) [name of my Azure Blob Storage account, as well as the Azure Blob Storage](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=46.48) [container name that I want the state to be stored in, and I do also need to](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=50.73) [provide an account key, which is a secret.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=55.18) [So I've used this syntax to say that the account key that's](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=58.44) [needed to connect to this storage account can be found in a](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=61.92) [Kubernetes secret. Again,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=65.19) [that's something we'll look at in more detail later on in this](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=67.59) [course. And there's a file here called aks‑deploy.ps1, which shows](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=70.68) [all the steps that you need to go through to get GloboTicket set up](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=76.53) [and running in Azure. Again,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=80.88) [we're going to be looking at this in a lot more detail later on. But for now,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=83.27) [I just want to focus on this line where we use kubectl apply to deploy this](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=87.53) [state store component definition file onto our AKS cluster. And this cluster](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=92.99) [has already been initialized with Dapr,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=99.04) [as you can see here, using the dapr init ‑k command, so it](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=101.69) [understands what these Dapr component definitions are.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=105.87) [And I've also set up that Kubernetes secret that we're](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=109.94) [using here with the call to kubectl create secret, which](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=112.62) [contains the storage account key.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=117.23) [So, if I visit my cloud instance of GloboTicket, which is running in Azure](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=120.14) [on AKS and it happens to be at this IP address, you can see basically the](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=125.21) [same website and everything is working just fine.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=130.94) [I can put items into my basket and everything seems](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=134.74) [to be persisted, as expected.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=138.2) [If I want to check what's happening behind the scenes,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=141.24) [I can actually go into the Azure Portal and look at that blob](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=144.46) [storage account that I've set up as the backing store for my state](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=148.28) [store, and I can look at the blobs inside of the container that we](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=151.94) [set up to use for the state store.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=156.75) [And, if we look at those blobs,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=159.54) [we can see that actually the way that this Dapr component](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=161.29) [has been implemented is that for each of the keys that we](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=164.48) [store in the state store,](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=167.53) [it's created a file in blob storage. And obviously, every implementation](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=169.61) [of the Dapr state store building block is going to have its own way of](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=174.79) [implementing the state store behavior, depending on the capabilities of the service that you're using.](https://app.pluralsight.com/course-player?clipId=6ab0b763-987b-45d7-9896-837329ac7941&startTime=178.89)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a)

[In this module,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=1.04) [we learned about how the Dapr state management building block can be used to](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=1.89) [greatly simplify the task of adding state storage to a microservice. We](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=6.98) [learned about the concept of components in Dapr and saw the default global](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=12.62) [components that are configured when we install Dapr in self‑hosted mode, as](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=17.49) [well as how we can pass in a path to a components folder when we call dapr run](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=22.63) [to allow us to specify our own component definitions. We saw that the state](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=28.07) [store building block supports pluggable backends and demonstrated this in](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=33.64) [action with both Redis and Azure Blob Storage state store component](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=38.51) [definitions. We learned that the way this works is that our application makes](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=43.42) [a call to the Dapr sidecar, and then the Dapr sidecar manages the details of](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=48.67) [how to actually talk to the underlying backend provider,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=53.9) [whether that's Redis, or Azure Blob Storage, or something else.](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=57.49) [We used the .NET Dapr SDK to make it really easy to get and set state values](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=62.04) [from inside an ASP.NET Core application. And, of course, if you're not using](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=68.02) [C#, you can use the Dapr SDK for your programming language of choice. And, in](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=72.97) [this module, I focused on how to get started using the state management](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=79.03) [building block, and it may well be that the capabilities I've demonstrated](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=83.2) [are perfectly sufficient for you if all you want to do is just get and set](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=87.34) [state by key.](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=91.83) [However,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=93.74) [if you'd like to dive deeper into some of the more advanced](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=94.2) [features of the state management building block,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=97.21) [such as how to query the state, or how to set time to live on your state,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=99.85) [then a great place to start exploring that is this page](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=105.39) [in the official Dapr documentation.](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=108.96) [Next up,](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=111.94) [we're going to learn about the service invocation building block in Dapr and see how we can use it to call the event catalog service.](https://app.pluralsight.com/course-player?clipId=9e4ad497-fdb2-4001-9706-71a73841a44a&startTime=112.7)

## [Invoking Services](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797)

[Hi, my name's Mark Heath, and in this module,](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=1.04) [we'll see how Dapr can help us with invoking other services.](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=4.14) [In the last module, we looked at our first Dapr building block,](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=9.24) [the state management building block,](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=13.2) [which we used within a single microservice. And in this module, we're](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=15.44) [going to be using the service invocation Dapr building block to allow us](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=19.77) [to communicate between our microservices.](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=24.54) [We'll start by learning about some of the key reasons why you](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=28.04) [might want to use the service invocation building block. And](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=31.4) [these benefits include service discovery,](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=35.41) [encrypted communication, built‑in tracing, and automatic retries.](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=38.54) [I'll explain how the service invocation building block](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=44.64) [actually works behind the scenes,](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=47.45) [and we'll see how to construct the URLs that allow Dapr to](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=50.04) [direct your requests to the correct service.](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=54.33) [We'll also update our GloboTicket demo application to use the](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=58.04) [service invocation building block to support the frontend](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=62.29) [microservice calling the event catalog. microservice.](https://app.pluralsight.com/course-player?clipId=904001d9-616e-4e15-bc94-caee49cae797&startTime=66.1)

### [Service Invocation Challenges](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb)

[When you build a distributed application,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=1.14) [you're likely to want to make a call from one microservice to another.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=4.1) [On the surface of things,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=8.28) [this seems like it ought to be relatively straightforward,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=9.88) [but there are some common concerns that need to be addressed.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=13.44) [First of all is the challenge of service discovery.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=17.74) [How can I find out the IP address of the server that's hosting my other service?](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=21.44) [Ideally, I'd just like to address the microservice by name,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=26.52) [for example, the catalog service,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=30.34) [and for my request to get automatically routed to the correct place.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=33.04) [Obviously,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=38.24) [there are other technologies apart from Dapr that can solve this problem,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=38.8) [but one of the nice things about using Dapr is that it's](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=43.24) [going to work in exactly the same way,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=46.22) [whether we're running on Kubernetes in production or](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=48.47) [running locally in Dapr's self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=51.81) [Second, whenever we make calls between microservices,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=55.54) [we want that communication to be secure.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=59.3) [Often in a microservices architecture,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=62.14) [your microservices may all be hosted on a virtual network, which](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=64.65) [protects you from malicious callers outside the network perimeter](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=68.87) [being able to call your microservices or listen in to traffic](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=72.51) [between the microservices.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=76.66) [But nevertheless, it's considered best practice in microservices to](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=78.94) [use a defense‑in‑depth approach to security where we make use of](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=83.1) [multiple levels of security. We'd ideally like our communication](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=88.02) [between microservices to be encrypted and to restrict communications](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=92.78) [between microservices to only allow the ones that are supposed to be](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=97.75) [talking to each other.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=101.83) [Another thing we'd like in a microservices environment is](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=103.84) [observability of calls between microservices.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=107.39) [It can be really hard to debug problems when there's](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=111.74) [communication between microservices involved.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=114.68) [And one advantage of sending all of your service‑to‑service traffic](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=118.34) [through the Dapr sidecar is that it can capture tracing information](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=121.88) [using the open telemetry standard, allowing you to view that](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=126.34) [telemetry in tools like Zipkin.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=130.12) [The final benefit I'll mention is the need to retry when there's a](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=134.24) [transient error. In distributed applications,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=137.8) [you can be sure that from time to time there are going to be](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=141.27) [network connectivity issues, or maybe the service that you're](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=144.31) [trying to reach is temporarily unavailable.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=148.06) [Of course,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=151.44) [you can implement your own service invocation retry logic yourself; however,](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=151.97) [you'd have to remember to do that everywhere where you](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=157.58) [were making a service‑to‑service call. With the Dapr service invocation building block, you can get retries automatically.](https://app.pluralsight.com/course-player?clipId=ba623f95-238c-4ab7-be72-d1a88cd30abb&startTime=160.26)

### [How it Works](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938)

[Let's see how service invocation works behind the scenes.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=1.14) [And a great place for us to learn about service invocation is, of course,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=5.74) [the official documentation here on docs.dapr.io. Here,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=10.12) [we're looking at an overview of the service invocation building](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=15.72) [block. In particular, let's focus on this diagram.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=19.01) [This diagram is showing two microservices,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=23.54) [Service A and Service B, and Service A wants to make a request to](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=26.4) [Service B. Without Dapr, Service A would need to find out the IP](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=31.18) [address of Service B and then make a direct call and handle any](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=35.89) [concerns like encryption and retries itself.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=40.21) [But when we're using the Dapr service invocation building block,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=43.64) [instead of Service A talking directly to Service B, it simply talks to](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=47.54) [its sidecar, which is available on localhost.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=52.37) [The Dapr sidecar then uses Dapr's name resolution component to find](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=56.44) [out the network address of the target service.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=61.84) [Now, if you're running in Kubernetes,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=65.54) [the name resolution component is just the standard Kubernetes DNS](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=67.67) [service. And if you're running in self‑hosted mode, Dapr uses](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=71.83) [something called mDNS, or multicast DNS. And Dapr actually supports](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=76.44) [pluggable name resolution components,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=82.42) [so you can swap this out for something else if you have the need to. However,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=84.83) [for the vast majority of cases,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=89.7) [these default name resolution components will work just fine.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=91.97) [Now we're up to step three in this diagram,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=96.94) [which is where the Dapr sidecar for Service A passes the invocation](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=99.62) [request onto the Dapr sidecar for Service B. You'll notice on this diagram](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=104.19) [that it shows that this call is encrypted using mutual TLS, and you might](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=110.52) [also notice here that it tells us that the communication between Dapr](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=115.9) [sidecars always uses gRPC, which offers improved performance compared to a](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=120.08) [regular HTTP request.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=126.77) [Step four is where the Dapr sidecar for Service B passes the request](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=130.14) [on to Service B. Service B then responds to its own Dapr sidecar,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=135.19) [and that gets proxied back, again, encrypted and using gRPC for](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=140.57) [performance over to the sidecar for Service A. Finally, in step](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=145.56) [seven here on the diagram,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=150.83) [the response from Service B is passed back to Service a. And so](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=152.53) [essentially, what Dapr is doing is acting as a proxy.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=157.33) [We just need to communicate with our sidecar, and we](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=161.44) [allow Dapr to take control of the rest.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=164.49) [By the way,](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=168.24) [if you're concerned that this proxying might introduce a performance](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=168.83) [degradation, then you can take a look at this helpful performance summary](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=172.3) [for Dapr provided in the official documentation.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=176.48) [Now there's a lot of useful information here, but the short summary is that you](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=180.14) [can expect the overhead to be in the region of one to two additional](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=184) [milliseconds on each request, which actually compares quite favorably with many](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=188.47) [commonly used service meshes on Kubernetes. And, of course, you need to](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=193.85) [remember that you're getting quite a lot of added value for that small addition to the time of each request.](https://app.pluralsight.com/course-player?clipId=4041b336-dfb5-484b-8d67-4c0bb9cb5938&startTime=198.17)

### [Service Invocation URLs](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430)

[We've just seen how the service invocation building block works under the hood,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=0.84) [but how do the URLs that we call when we want to talk to another](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=6.24) [microservice need to change if we want to use that building block?](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=9.72) [Well, currently in the GloboTicket demo application,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=14.64) [the frontend service makes a call to the event catalog](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=17.73) [microservice to ask for details of an event,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=21.54) [and the URL that it calls might look something like this.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=25.34) [We're requesting details of event number 12345,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=29.04) [and in this example,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=33.64) [I'm assuming that we've got some kind of DNS resolution already set up](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=34.82) [that knows where the event catalog service is located.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=39.54) [So how does that change with Dapr?](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=43.64) [Well, instead of talking directly to the event catalog service,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=45.64) [we instead talk to the sidecar of the service that's making the request,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=49.96) [in this case the frontend service, and here you can see the URL that we called.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=54.99) [We're talking to our Dapr sidecar, which is available on localhost,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=60.15) [and in this example, I'm assuming it's using the default Dapr port of 3500,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=65.11) [and we're telling Dapr that we want to invoke another service, and the name](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=71.24) [of the service that we're invoking is the catalog service,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=76.7) [or in Dapr terminology, the catalog application.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=80.32) [We also need to specify what method we want to call.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=84.24) [The method is simply the endpoint on the service we're calling,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=88.04) [and in this example, it's event/12345,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=91.94) [the same endpoint that I showed you in the previous example.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=96.64) [Now,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=101.04) [although this URL is a bit longer and more complex than the one that you'd](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=101.25) [call to make a direct request to another microservice,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=105.62) [in practice,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=109.84) [it doesn't really matter because this whole section on the](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=110.52) [left up to /method remains constant, and so in most programming frameworks,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=113.65) [you can simply set this up as the base address, and then your code just](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=120.21) [needs to add on the actual endpoint that you want to call.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=124.89) [However, if you don't like this long prefix,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=128.74) [Dapr actually offers an alternative approach where we just make a call](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=131.52) [to the sidecar as though it were the target service.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=135.74) [So with this approach, the URL we'd call looks like this.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=139.64) [Of course,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=143.64) [now the URL doesn't contain any information about](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=144.14) [which microservice we want to talk to,](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=147.44) [and so you have to supply a Dapr ID header with the name of the](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=150.14) [target microservice for this technique to work. Let's see this in action now, and I'll show you both types of URL.](https://app.pluralsight.com/course-player?clipId=0349fbbb-5d7b-44e7-9f3b-a8cbacc8e430&startTime=154.43)

### [Demo: Invoking a Service Through the Dapr Sidecar](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da)

[In this demo, we'll see the service invocation capabilities of Dapr](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=1.24) [by directly making a HTTP request to the Dapr sidecar. First, let's](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=5.74) [start the event catalog microservice in self‑hosted mode just like we](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=12.36) [did before with the start‑self‑hosted PowerShell script file inside](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=17.33) [the catalog folder.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=22.46) [This starts both the event catalog microservice and the Dapr sidecar for](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=24.34) [that microservice, which is listening on port 3501.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=28.97) [Now, I'm going to start by just calling the event catalog](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=34.24) [microservice directly, and this is so that you can see the](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=37.42) [URLs that it's expecting. Once again,](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=41.07) [I'm using the REST client extension in Visual Studio Code, which gives](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=43.91) [us a nice, easy way to issue HTTP requests that I've defined in a](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=48.44) [simple text file that we're looking at now. The event catalog](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=53.28) [microservice is listening on port 5016.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=57.61) [So if I call the event endpoint, you can see that I get a JSON](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=61.74) [list of events back, and I can also pass one of the event IDs in](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=65.76) [to get details of a specific event.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=71.4) [So now, instead of calling the event catalog microservice directly,](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=75.54) [I'm going to talk to the Dapr sidecar, which is listening on port 3501.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=80.34) [I'm saying that we want to invoke the event catalog service, and the method is](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=85.64) [just /event, which is to get the full list of events. As you can see, the Dapr](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=90.21) [sidecar simply passed that method on to the actual event catalog microservice](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=96.72) [and proxied the response back to us.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=102.08) [So we've got exactly the same response as we did when we called](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=104.94) [the event catalog /event endpoint directly.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=108.74) [Let's try the same thing, but with an event ID as well. And you can](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=113.04) [see that once again we get the same result, as if we'd gone directly](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=117.13) [to the event catalog microservice. Now, in these examples,](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=121.21) [I'm talking to the sidecar of the event catalog microservice because that's](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=125.52) [the only sidecar that I've got running at the moment.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=129.71) [But I could actually send this request to any Dapr sidecar, and it](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=133.14) [would work out where to route the request to.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=137.25) [Finally, before we move on to using this in code,](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=140.64) [let's look at the alternative syntax I mentioned that gets rid of the](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=143.88) [v1.0/invoke/catalog/method part of the URL. Here,](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=148.5) [I'm still talking to the sidecar on port 3501, but you can see that](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=154.52) [the rest of the URL is just asking for specific event details, and](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=159.06) [then I'm passing in a header called dapr‑app‑id with the value of](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=163.62) [catalog, which tells the Dapr sidecar that I want to invoke this](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=167.7) [method on the application with the name catalog, and you can see that](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=171.88) [this works as well.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=177.28) [Now all the examples that I've just shown you happen to be using the](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=179.34) [GET http method, but the Dapr sidecar can use any HTTP method, so we](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=182.89) [could use POSTs, and PUTs, and DELETE requests, and they would all](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=188.95) [work in exactly the same way.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=192.95) [So now we've hopefully got an idea of how the service invocation building](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=195.54) [block works and what the URLs that we need to call look like.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=199.33) [Let's update the GloboTicket demo application to use the service invocation building block.](https://app.pluralsight.com/course-player?clipId=ecb46989-9913-420e-bd72-75efc64672da&startTime=203.73)

### [Demo: Update GloboTicket to Use Service Invocation](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153)

[In this demo, we'll update the GloboTicket application to use the Dapr](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=1.04) [service invocation building block to allow our frontend microservice](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=5.81) [to call the EventCatalog microservice.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=10.83) [First,](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=13.45) [let me show you how the frontend application currently makes](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=14.57) [requests to the EventCatalog microservice.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=17.47) [Here in the EventCatalogService class in the frontend application,](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=20.18) [you can see that it's very simple.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=25.52) [We get a HttpClient passed into our constructor, and this](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=27.57) [has already been set up to point to the correct base](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=32.08) [address for the EventCatalogService.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=35.09) [The method to get all events is called GetAll, and we simply make an HTTP get](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=38.24) [request by calling GetAsync on the event endpoint and then deserializing the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=43.56) [response. And to get details of a specific event, we call GetAsync on the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=49.9) [event/ event id endpoint. And, of course, there's nothing in here at all about](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=56.09) [where the EventCatalog/Service is located, and that's because it's all been](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=62.67) [configured in the startup.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=67.22) [So here in Program.cs, which is where the code that runs when our service](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=69.54) [starts up is, you can see that we're using AddHttpClient to set up the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=74.9) [HttpClient that's being used by the EventCatalogService, and we're setting](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=80.79) [the BaseAddress property of that HttpClient to a value that we're reading](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=86.52) [out of Config. And if I jump into my appsettings.Development.json file for](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=91.82) [the frontend project,](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=97.19) [you'll see that we've got the EventCatalog's base address set up here for when](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=98.65) [we're running locally in a development environment, which is localhost port](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=102.9) [5016. And this is quite typical for what you'll see in many real‑world](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=107.6) [microservice applications. They need to be told through configuration where to](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=112.41) [find the other services and often that's different depending on what](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=117.58) [environment you're running in, and this can be a pain to set up correctly and a](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=121.68) [source of frustrating errors.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=126.49) [Let's switch this over to use Dapr.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=129.44) [It's really quite straightforward to do.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=131.84) [All we need to do is change the BaseAddress of this](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=134.54) [HttpClient to the address of the Dapr sidecar.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=137.79) [The Dapr sidecar port is available to us using the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=142.44) [DAPR\_HTTP\_PORT environment variable.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=145.78) [So I am fetching that with a call to GetEnvironmentVariable, and](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=149.54) [we can use that port number to construct the service invocation](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=153.96) [BaseAddress, which is following the pattern that we looked at](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=157.32) [earlier in this module.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=160.96) [The BaseAddress is now http://localhost on the daprPort number, and then](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=163.04) [v1.0/invoke/catalog, and catalog obviously is the name of the service that we](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=170.46) [want to talk to, and then method. And this means that because our](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=176.16) [EventCatalogService class was just using relative endpoints, those are just](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=180.91) [going to get appended onto our BaseAddress, so this call to event actually](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=186.26) [results into a call to this full URL, which includes the BaseAddress of the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=191.67) [Dapr sidecar. And because we're now using Dapr for service discovery, we](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=197.06) [don't need these settings anymore, so I can delete them, and I'll save my](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=202.49) [changes.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=207.51) [Let's run this and check that it's all working.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=209.64) [I'll start the EventCatalog microservice first using the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=212.64) [start‑self‑hosted script in the catalog folder, and I'll start](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=216.39) [the frontend microservice, again, using the start‑self‑hosted](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=221.05) [script in the frontend folder.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=225.17) [Now if we visit the website, which is on localhost port 5266,](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=227.94) [everything looks exactly the same and we can see a list of events,](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=233.23) [but this time, rather than directly knowing where to find the](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=237.84) [EventCatalog microservice, we've retrieved this data through the Dapr](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=241.52) [sidecar, and that means we're already benefiting from the additional](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=246.15) [security, retries, and observability that the Dapr service invocation](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=250.1) [building block has to offer, and we only needed to make a very minimal code change.](https://app.pluralsight.com/course-player?clipId=eeb2fc72-33d5-4efa-a5a2-0a07629ac153&startTime=255.76)

### [Demo: Using the Dapr SDK for Service Invocation](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2)

[In our last demo,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=1.44) [we used Dapr service invocation without using the Dapr SDK at all,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=2.57) [and that's because service invocation in Dapr is so straightforward to use that](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=8.24) [you might not feel that the Dapr SDK offers any benefits.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=12.86) [However,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=16.94) [the SDK does provide us a feature that helps us to do service invocation,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=17.35) [and you may prefer that.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=21.9) [So in this short demo, let's switch over to use the Dapr SDK instead.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=23.43) [What the Dapr SDK does is that,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=28.54) [instead of us having to know how to construct this special URL and](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=30.8) [having to get hold of the Dapr HTTP port ourselves,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=35.25) [instead, we're going to register our EventCatalogService as a singleton,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=39.64) [and into the constructor,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=44.54) [we'll pass a HTTP client that's been created with the](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=46.09) [DaprClient.CreateInvokeHttpClient method,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=50.62) [and we need to pass in the name of the target microservice or application,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=55.54) [which is catalog,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=59.81) [and this is going to create an HttpClient that's already set](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=61.34) [up for us with the correct base address,](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=65.48) [and this simplifies our code even further.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=68.4) [Let's quickly test and make sure it's still working as expected.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=71.64) [I'm going to rebuild the frontend service and start it up again.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=76.04) [And now when I load the website in my browser at localhost port 5266, we can see that our event catalog is still loading correctly.](https://app.pluralsight.com/course-player?clipId=0e52c709-9837-4912-8a8b-6b552fec92b2&startTime=81.24)

### [gRPC and Access Control Lists](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d)

[Before we wrap up this module,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=1.04) [I did want to mention a couple of additional capabilities of Dapr's service](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=3.08) [invocation building block that you might be interested in.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=7.53) [And the first of those is that,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=11.44) [as well as accessing regular HTTP APIs that we've seen so far,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=12.86) [you can also use Dapr to call services exposed via gRPC.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=17.99) [Now we've already said that Dapr uses gRPC internally](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=23.14) [for sidecar‑to‑sidecar communications, which offers improved performance,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=27.46) [but this expands on that to allow Dapr service invocation to be](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=32.74) [used to call APIs that are exposed using gRPC.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=36.95) [Now, this feature is still in preview at the time I'm recording this.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=41.74) [And, of course, gRPC is still a relatively new technology,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=45.29) [but it is growing in popularity due to the performance benefits it offers.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=49.94) [So if you're considering Dapr and you'd like to use](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=55.04) [gRPC for your own microservice APIs, then you may want to explore this option.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=57.7) [The second feature I want to mention is the ability to](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=63.94) [configure access control for service invocations.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=66.69) [By default, Dapr will allow you to make service invocations to any other service.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=70.22) [But Dapr offers the concept of access control lists,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=77.24) [which allows you to choose which services are allowed to make requests,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=81.03) [what endpoints they can call,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=85.64) [and you can even go to the level of specifying what HTTP](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=87.69) [verbs such as GET or POST are allowed.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=91.11) [And this page in the official documentation gives some](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=94.64) [examples of how to configure common scenarios.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=97.8) [So you can use these as starting points for](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=101.64) [configuring your own access control rules.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=104.41) [Access control lists give you the ability to adhere to](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=107.49) [the principle of least privilege,](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=111.19) [which is a security best practice where you only allow services to perform the](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=113.74) [specific actions that they ought to be allowed to perform.](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=118.84) [And that's another potential reason why you might want to adopt Dapr if you'd like to have this level of control..](https://app.pluralsight.com/course-player?clipId=db15e069-112f-4142-8034-dc2175a0720d&startTime=122.94)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58)

[In this module,](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=1.04) [we've seen how Dapr gives us a simple way to make requests to other services.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=1.91) [We saw that the benefits include service discovery,](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=8.04) [built‑in retries, traceability, and encryption with mutual TLS.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=11.35) [We saw the structure of the URL that you need to use in order](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=17.34) [to use the service invocation building block and how the Dapr](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=21.24) [SDK can construct that for us.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=25.08) [We also learned a bit about how service invocation works behind the scenes,](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=28.64) [proxying the requests between the Dapr sidecars of both services.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=33.94) [Of course, in a microservices architecture,](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=38.61) [sometimes rather than making a direct service invocation to another service,](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=42.24) [we'd like to use asynchronous messaging instead.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=47.48) [And that's what the Dapr pub/sub messaging building block can offer and we'll be looking at in the next module.](https://app.pluralsight.com/course-player?clipId=c8389c52-3098-417c-814f-f98ba2529e58&startTime=51.34)

## [Communicating using Pub Sub Messaging](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c)

[Hi.](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=0.94) [My name is Mark Heath, and in this module,](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=1.18) [we'll see how the Dapr pub/sub messaging building block](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=4.29) [enables microservices to communicate with one another](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=8.53) [asynchronously using messaging.](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=12.14) [We'll start out with a quick look at how pub/sub messaging works and why we](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=15.14) [might want to use it in a microservices architecture.](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=20.06) [Then we'll see some of the advantages of choosing the Dapr publish and](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=24.04) [subscribe building block and learn a bit about how it works behind the](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=28.27) [scenes. In particular, we'll see that there's a few different choices](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=32.53) [for how we can tell Dapr that we want to subscribe to messages on a](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=37.1) [particular topic,](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=41.57) [and we'll actually use the pub/sub building block in our GloboTicket demo](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=43.34) [application to send a message to our ordering microservice whenever a new order is placed in the front‑end website.](https://app.pluralsight.com/course-player?clipId=9aeadd97-9bec-4ac3-9a10-df92feb9b62c&startTime=48.18)

### [Pub Sub Messaging](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078)

[In our last module,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=1.24) [we saw how one microservice can directly call another microservice](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=2.47) [using the Dapr service invocation building block,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=7.83) [and the example we looked at where the front‑end website needed to](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=11.6) [get a list of events from the event catalog microservice was a good](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=15) [example of where this makes sense.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=19.64) [We need that information immediately,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=22.54) [and so we make a direct synchronous call using HTTP,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=24.41) [which is, of course,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=30.24) [proxied through the Dapr sidecars. But there are many other scenarios in](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=31.11) [microservices where it might be better to use messaging instead.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=36.34) [For example, when the customer places an order in the website,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=41.64) [we need to pass details of the order onto the ordering microservice,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=46.14) [which might have a whole host of responsibilities,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=51.04) [such as ensuring that payment is taken and emailing](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=54.01) [a confirmation to the purchaser.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=57.51) [However, from a user interface perspective,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=60.14) [all that we need to do on the website is say to the customer,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=63.02) [thank you for your order.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=66.68) [They don't need to sit and wait for all the back‑end tasks](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=68.54) [associated with that order to complete. Those can be done](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=71.76) [asynchronously in the background, which we can accomplish by sending a message.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=75.88) [There are several reasons why using messaging for communication](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=81.54) [between microservices is beneficial. With messaging, rather than](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=85.36) [communicating directly to the target service,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=90.42) [you publish a message to a message broker. Then that message broker can](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=93.95) [either push that message to the recipient or store it temporarily until the](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=99.65) [recipient asks to receive it. And this means that if the recipient of your](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=105.05) [message isn't currently available,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=110) [you can still send the message knowing that it will get picked up later.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=112.64) [Another advantage of this pattern is that it can be](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=118.04) [expanded to support multiple subscribers.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=120.75) [In other words,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=124.74) [you can publish one message to what's known as a topic, and then](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=125.45) [multiple services can register their interest in receiving that](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=129.93) [message by creating what's known as a subscription.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=133.85) [In the example we're looking at, you could imagine that when an order is placed,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=138.64) [the two activities of taking your payment and emailing you your](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=143.1) [tickets might be handled by different microservices, and so a](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=147.42) [single message indicating that a new order has been placed could be](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=151.95) [published to a topic called orders, and then the orders topic could](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=156.24) [have two subscriptions,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=160.74) [one by a payments microservice and one by an email microservice. Notice](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=162.75) [that even though we only send one message to the topic,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=169.42) [both subscribers get a copy of that message. And](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=173.02) [this is a very extensible model.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=177.11) [If in the future there's something else we want to do](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=179.74) [whenever a new order is placed,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=182.61) [we can simply create a new subscription on that topic.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=184.82) [So having a message broker that supports pub/sub messaging allows us to write](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=189.04) [our software in an event‑driven manner where microservices simply publish](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=194.25) [what are sometimes called domain events, messages that indicate that](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=199.72) [something of interest has happened in the system and other microservices](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=204.2) [subscribe to those domain events and carry out different activities. This](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=208.8) [greatly reduces the coupling between services since the publisher of a](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=214.28) [message doesn't need to know anything about who the subscribers are. Using](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=218.96) [messaging also has scalability benefits because if we build up a backlog of](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=224.12) [messages,](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=229.13) [we can scale out additional instances of the subscriber](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=230.14) [microservice in order to catch up. And it has resilience](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=233.96) [benefits as subscribers can catch up on messages they've missed](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=239) [if they have any temporary downtime.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=243.82) [Now that we've understood the basics of pub/sub messaging and](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=246.84) [the reasons why we might want to use it, let's see how Dapr makes this capability available to us.](https://app.pluralsight.com/course-player?clipId=fb3049e3-c577-4176-b920-b58afab8b078&startTime=250.65)

### [Dapr Pub Sub Building Block](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0)

[Let's take a look at how we can publish messages and subscribe to](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=1.14) [them using the Dapr pub/sub messaging building block. A](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=5.39) [microservice that wants to publish a message makes an HTTP post to](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=9.99) [the Dapr sidecar using this URL format,](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=15.48) [which should be looking quite familiar to you by now.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=18.98) [This is saying that we're accessing the publish capability of](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=22.44) [Dapr, and we also need to specify the name of the pub/sub](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=25.72) [component that we want to interact with.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=29.8) [In this example, it's just called pubsub. And in most systems,](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=32.54) [you'll probably just have a single Dapr pub/sub component, but](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=37.31) [you can have more than one if you want.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=41.71) [The final portion of this URL is the topic name that we want to publish](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=44.84) [to. In this example, we're publishing to a topic called order, and you](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=49.42) [can define as many topics as makes sense for your application. The body](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=54.57) [of the request should contain the message payload that is going to be](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=59.72) [received by the subscribers.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=63.79) [In this example, I'm just showing some of the details of the order.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=66.64) [What happens on the subscriber end? Let's imagine that we've published a](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=71.34) [message to the orders topic on the message broker, and let's imagine that](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=75.43) [we've got two microservices, the payments and email microservices that both](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=80.19) [want to subscribe to the orders topic.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=85.6) [What happens is that the Dapr sidecar for both of our](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=88.44) [subscriber microservices connects to the message broker and](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=91.93) [registers to receive messages on that topic.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=96.49) [Then, whenever a message is posted to the topic,](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=100.34) [the Dapr sidecar receives that message and then passes it on to the](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=103.74) [microservice. And the way it does that is it calls an endpoint that you've](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=108.29) [exposed on your microservice. The name of the endpoint is configurable, but](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=113.18) [let's say for this example that it's called order.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=118.26) [The Dapr sidecar can then make an HTTP request to your microservice, and](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=121.84) [the body of the request contains the message payload.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=127.4) [A nice thing about this approach is that from your microservices perspective,](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=131.74) [it's a push model.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=136) [Any polling of the message broker that might be required](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=137.54) [can be handled by the Dapr sidecar.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=140.9) [Now you might be wondering at this point,](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=144.04) [how does Dapr know which of your microservices want to](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=146.21) [subscribe to a particular topic, and how does it know what](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=150.12) [endpoint to call them on with those messages?](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=154.61) [Dapr actually offers two ways of setting up subscriptions, so let's talk about those next.](https://app.pluralsight.com/course-player?clipId=1ddf7bf4-4d2b-42ff-a724-cf05ba5bbda0&startTime=157.87)

### [Declarative and Programmatic Subscriptions](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331)

[We need a way for our subscribers to register that they're](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=1.14) [interested in receiving a message posted to a topic,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=4.84) [and Dapr allows us to do that in two ways known as declarative and programmatic.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=8.84) [Declarative subscriptions are where we create a YAML file similar to the](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=15.94) [component definition YAML files we've already seen,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=20.76) [and programmatic subscriptions are where we define our subscriptions in code.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=24.44) [Let's start by looking at declarative subscriptions.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=30.04) [Here,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=34.34) [we're looking in the official Dapr documentation at an](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=34.68) [example of a declarative subscription file.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=38.12) [We can see that the kind is set to subscription,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=42.04) [and we can give a name to this subscription,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=45.94) [which here is just called order\_pub\_sub.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=48.25) [Then in the spec, we're going to say what topic we're subscribing to,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=52.24) [what route we want to be called with each message,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=57.74) [which is checkout in this example,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=61.44) [and the name of the pubsub component that the topic exists on.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=64.34) [Notice here is the Scope section which lists two microservices.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=69.34) [Obviously, not every microservice wants to subscribe to all messages,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=75.14) [and so this limits the services that will attempt to subscribe.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=80.24) [Only the microservices with the Dapr application names,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=84.44) [orderprocessing and checkout,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=87.91) [are going to create subscriptions and get called on the](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=90.48) [/checkout endpoint with new messages.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=94.17) [The second method of subscribing is called programmatic.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=98.24) [With this approach,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=102.54) [your microservice is able to report which topics it wants to subscribe to.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=103.45) [The way that this works is that the Dapr sidecar attempts](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=109.54) [to call the /dapr/subscribe endpoint,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=112.71) [and if your microservice is listening on that endpoint,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=117.34) [you can respond with details of which topics you want to subscribe to,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=120.44) [as well as which endpoint that you want to be called](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=125.21) [whenever a new message is received.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=128.11) [Now,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=130.94) [if that sounds like it's a bit more complicated to](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=131.22) [set up than the declarative mode, you're right.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=133.65) [However, the Dapr SDKs provide quite a lot of help to simplify the process,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=137.34) [and in our demo,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=142.94) [I'll show you how easy it is to configure the programmatic technique in .NET.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=144.04) [While we're talking about subscriptions,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=149.64) [it's important that you understand the concept of competing](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=152.03) [consumers when you're working with pubsub messaging.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=155.64) [Let's imagine we have a situation where there are two](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=159.44) [instances of the payments microservice.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=162.43) [The payments microservice is configured to receive messages on the orders topic,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=166.24) [but what happens if we have two instances of the payments](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=172.24) [microservice running for resilience reasons or maybe we've](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=175.78) [scaled out to handle additional load.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=179.72) [Will both instances receive a copy of the message?](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=183.04) [The answer is no because,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=186.54) [although there are two instances of the payments microservice,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=188.44) [there is only one subscription,](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=192.22) [and so the two instances are what's known as competing consumers.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=194.64) [That means that the message will only be delivered to one of those instances.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=199.54) [Let's look next at how we can configure the underlying service that acts as a message broker.](https://app.pluralsight.com/course-player?clipId=553294f3-1b88-492d-be48-9104b023f331&startTime=205.54)

### [Pub Sub Component Definitions](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1)

[How does the Dapr sidecar know which message broker](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=1.14) [we're using and where to find it?](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=5.06) [Well, it works just the same as we saw for the state management building block.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=7.64) [Dapr supports many different pub/sub message brokers that can be used](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=12.69) [to implement the pub/sub messaging building block.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=17.32) [You create a component configuration file that](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=20.74) [defines which one you want to use,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=23.57) [along with the details of how to connect to it.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=26.12) [Here,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=29.74) [we're looking at the pubsub component definition file that I'm](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=29.9) [using for running locally in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=34.02) [We can see that I'm giving this component a name of pubsub,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=37.94) [which is the name that you put after /publish in the request](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=41.84) [to the Dapr sidecar to publish a message,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=46.14) [and the component type is pubsub.redis.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=49.24) [And this is in fact the default pub/sub implementation that gets set up](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=51.58) [when you install Dapr in self‑hosted mode because,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=57.64) [as we've already seen,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=61.24) [it creates a Redis container that can be used not just for the state store,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=62.41) [but also for the pub/sub building blocks.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=67.04) [So in the metadata,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=70.84) [we simply need to provide the network address of our local Redis server,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=72.18) [which is running as a Docker container, and the password,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=77.34) [which is blank.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=80.96) [If we take a look here at the official documentation for Dapr,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=83.04) [we can see the current set of supported pub/sub brokers,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=86.9) [along with their status of whether they're stable or](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=91.52) [still considered to be in alpha or beta.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=94.57) [As you can see, many of the most popular message brokers are supported,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=97.18) [including Apache Kafka, NATS, and RabbitMQ.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=102.79) [And if I scroll down,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=107.44) [we can see that there's also support for the messaging services](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=108.64) [that are available in various cloud providers,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=112.16) [such AWS, Google Cloud Platform, and Azure.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=114.95) [I'm deploying GloboTicket on Azure Kubernetes Service,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=118.99) [so it makes sense for me to pick an Azure message broker for](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=123.65) [my pub/sub building block in production.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=127.54) [I've chosen Azure Service Bus, so let's look at the configuration for that.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=130.74) [Here we can see that I've given my component definition the same name,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=135.27) [pubsub, but it's of type pubsub.azure.servicebus.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=140.42) [For the connection string,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=144.29) [I've chosen not to hard code my connection string in](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=147.53) [the component definition file, but instead reference a Kubernetes secret.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=150.49) [And this is considered a best practice from a security perspective,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=155.16) [and we'll be looking in more detail at the Dapr support](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=160.26) [for secrets later on in this course.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=163) [Now we've learned about how to configure pub/sub messaging,](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=165.22) [we're ready to update the GloboTicket application to take advantage of the Dapr pub/sub building block.](https://app.pluralsight.com/course-player?clipId=8469e6fa-915d-42be-ba49-1e9fbd52b3f1&startTime=169.66)

### [Demo: Publishing a Message](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4)

[In this demo,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=1.04) [we're going to update the GloboTicket front‑end web application to publish](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=1.95) [a message to a topic whenever a new order is placed.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=7.11) [Here,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=10.91) [we're in the components folder that we use when](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=11.41) [we're running in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=14.69) [We already looked earlier in this course at the stateStore.yml file,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=17.04) [which set up the shop state stateStore using our Redis](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=21.1) [Docker container as the backing store.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=24.91) [And here, in the pubsub.yaml file, we're doing something very similar.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=27.22) [This component is called pubsub,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=32.04) [and it's also using Redis as the backing](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=34.08) [implementation for the pubsub messaging block.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=37.15) [And the config file you're seeing here is exactly the same](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=40.3) [as the default global component that gets set up when you](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=44.01) [install Dapr in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=47.87) [All of our startup scripts for running in self‑hosted](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=50.53) [mode points to this components folder,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=54.11) [which means that all of our services will be able to publish](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=56.68) [messages to a topic on the pubsub component.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=59.72) [Now let's look at the code that actually publishes a message.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=63.64) [Here, we're looking at the CheckoutController at the purchase endpoint,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=67.31) [and this is called when the customer actually purchases something.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=72.7) [You can see here that we call the SubmitOrder method on the](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=76.94) [orderSubmissionService and pass it this checkout object,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=81) [which contains details of the order that's just been placed.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=85.3) [The IOrderSubmissionService interface has just got a single SubmitOrder method.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=90.14) [And in the demo code base,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=95.33) [I've provided an implementation that doesn't use Dapr](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=97.26) [called HttpOrderSubmissionService,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=100.57) [but we're going to look at the implementation that uses Dapr pub/sub messaging,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=104.44) [which is here in DaprOrderSubmissionService.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=109.44) [You'll notice that in the constructor,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=111.76) [we're asking for an instance of daprClient,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=115.26) [which is part of the Dapr .NET SDK.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=118.15) [Then, in the SubmitOrder method,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=121.44) [the vast majority of the code that you can see here is simply](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=124.27) [constructing the message that we're going to publish.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=127.56) [The message object we're constructing is of type OrderForCreation,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=130.94) [and we're setting up details of the customer and the items in the order,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=134.67) [which we fetched from the shoppingBasketService.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=139.54) [And you may remember that earlier in this course,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=142.64) [we saw that the shoppingBasketService is making use of](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=145.22) [the Dapr stateStore building block.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=148.52) [Then, to actually publish the message,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=151.01) [we're calling PublishEventAsync on the daprClient,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=153.95) [passing in the name of the pubsub messaging component,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=158.35) [which, in our case is pubsub, and the name of the topic that we're publishing to,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=161.77) [which is orders.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=167.27) [Note that we don't have to do anything to create this topic ourselves.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=169.14) [It will get automatically generated for us.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=172.73) [Finally, we pass in the message itself, which is an instance of OrderForCreation,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=176.64) [and this is going to get serialized to JSON behind the](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=183.04) [scenes to be passed to the message broker.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=185.81) [One important thing to note that when you're using messaging is](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=189.14) [that the format of the messages that you send between](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=192.52) [microservices is essentially a contract.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=195.5) [Both the sender and receiver have to understand the message format,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=199.24) [and so you should take care not to make breaking](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=203.32) [changes to the format of a message.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=206.16) [Okay, that's all we need to do to publish a message,](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=209.44) [but we haven't written any code yet to subscribe to the message, so let's do that in our next demo.](https://app.pluralsight.com/course-player?clipId=12e9e475-7e4f-4fd4-aac4-15268619a5c4&startTime=212.8)

### [Demo: Receiving a Message](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32)

[In this demo,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=1.04) [we'll update the ordering microservice to subscribe to the message on the orders](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=1.93) [topic that we published a message to in our previous demo.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=7.13) [We're going to use the programmatic subscription method](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=11.44) [making use of the Dapr SDK for ASP.NET.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=14.73) [Here, we're looking at the project file for the ordering microservice,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=19.24) [and you can see here that we've referenced the Dapr.AspNetCore NuGet package.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=23.54) [This not only contains the Dapr client, which we've used before,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=29.74) [but it has some additional helpful support specifically for](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=33.84) [ASP.NET Core that we're going to be making use of.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=37.66) [Let's take a look at the startup code in Program.cs.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=41.54) [You can see here the AddDapr method that registers the Dapr client in](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=46.14) [our microservices inversion of control container.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=51.32) [Again, we've done that before in this course,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=54.84) [but there are two new things that we've needed to add.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=57.84) [First is UseCloudEvents.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=61.04) [This is needed because Dapr takes advantage of the cloud](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=64.64) [events specification as a message format,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=68.23) [and this means that the messages we publish get](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=71.84) [wrapped into a cloud events message, which adds additional metadata,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=74.48) [and what this UseCloudEvents method does is it allows us to](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=79.64) [easily unwrap those cloud events and get back to the underlying](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=83.4) [data type that our code is expecting.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=87.66) [Another thing we've added here is in the call to app.UseEndpoints.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=91.14) [This call to endpoints.MapSubscribeHandler is what sets up the](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=96.54) [endpoint that the Dapr sidecar is going to call to find out what](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=101.3) [topics our application wants to subscribe to,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=105.39) [and this means that the Dapr sidecar can call our application on](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=109.24) [the /dapr/subscribe endpoint, and it will respond telling the](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=113.27) [sidecar what topics we want to subscribe to.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=117.79) [So those are a few things we needed to set up in](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=121.44) [the startup of our microservice, but now we've done that,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=124.09) [we can create the endpoint that's actually going to receive the messages.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=127.77) [The method that we want to be called whenever a new message is published](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=132.84) [on the orders topic is here in the order controller.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=136.6) [We have an endpoint called Submit, and it accepts HttpPost requests,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=140.3) [and that's important because the Dapr sidecar will post to this](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=146.77) [endpoint whenever a new message is available,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=151.14) [and this HttpPost attribute is simply saying that this endpoint is](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=154.94) [listening on the base route for this controller,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=159.62) [which will be /order and accepting post requests.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=162.32) [The way that we ensure that Dapr knows that we want to create a new](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=167.74) [subscription is by adding this topic attribute to the method.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=171.08) [We're saying that we want to subscribe to messages posted to](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=175.74) [the orders topic on the pubsub component.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=179.4) [The Dapr.AspNetCore SDK looks for methods that have this topic attribute and](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=183.34) [then uses that information to construct the response to the method that asks](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=189.27) [for the details of the subscriptions we'd like.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=194.36) [The other thing to notice for now about this method is that we're expecting](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=197.74) [an OrderForCreation object to be in the message body,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=202.25) [and this is, of course,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=207.24) [the same type that we sent from the frontend microservice.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=208.31) [However,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=213.24) [remember that we said that this gets wrapped using](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=213.84) [the cloud events message format,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=217.11) [and that was why we needed the UseCloudEvents method in our startup.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=219.74) [With that present,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=224.24) [the Dapr SDK is able to unwrap the payload of the cloud](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=225.2) [event into the type that we're expecting.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=229.76) [Inside this method, for now,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=233.64) [we're just going to log a message to say that we have received the order,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=235.36) [but we'll be updating this later in the course to actually send an email.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=239.34) [Let's test this out.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=243.94) [Here you can see the start self‑hosted script for the ordering microservice,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=246.14) [and we've seen this before with the other two microservices,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=251.78) [so the contents won't be a surprise to you.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=255.24) [It's exactly the same,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=257.84) [except that we've picked the next free port number for the Dapr sidecar 3502,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=259.39) [and we've told Dapr which ports our ordering](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=265.34) [microservice is listening on port 5293.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=268.45) [It's pointing at the Dapr components folder,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=273.34) [which as you remember,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=276.44) [includes that pubsub.yml file which configures the pubsub messaging](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=277.5) [component and enables our sidecar to create a subscription and receive](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=282.71) [messages to pass on to our microservice.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=288.06) [And so here in a Visual Studio Code terminal window,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=290.96) [I'm going to run that start self‑hosted script.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=295.37) [And I need to do exactly the same thing for the other two services starting](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=299.44) [the event catalog microservice and the frontend microservice because all three](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=303.54) [are going to be involved in the purchase process.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=308.99) [Now in a browser, I'm going to visit the GloboTicket application,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=312.34) [we'll pick an event, add a ticket to our basket,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=316.54) [and I'll go to the checkout, I'll fill in some customer details about this order,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=320.44) [and make the purchase.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=326.33) [And if everything has worked correctly,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=328.54) [the frontend microservice should have published a message to the orders topic.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=330.68) [Let's jump back and take a look at the log output for our microservices.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=336.44) [Here,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=341.54) [we can see that the frontend has posted an order message](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=341.78) [using the Dapr pubsub building block.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=345.47) [And if I look at the log output for the ordering microservice,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=348.84) [I can see that we've received that message,](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=352.44) [and we know that we've been able to deserialize it correctly because we can see the customer name.](https://app.pluralsight.com/course-player?clipId=769dd716-687b-44e8-9336-7cc73e8d9b32&startTime=355.24)

### [Additional Pub Sub Building Block Characteristics](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0)

[Before we wrap up this module,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=1.14) [I did want to briefly discuss a few additional characteristics of the Dapr](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=3.12) [pub/sub building block that are worth being aware of.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=8.18) [First, Dapr offers an at least once delivery guarantee for messages.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=12.54) [In other words,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=18.64) [Dapr will ensure that every published message is](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=19.31) [received at least once by every subscriber.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=22.53) [Of course, in normal operation, messages should be delivered exactly once,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=26.24) [but what this means is that you should try to write idempotent message](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=31.04) [handlers that are able to cope with the possibility that on occasions the](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=34.92) [same message might be delivered more than once.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=39.44) [And by the way, this isn't a Dapr‑specific thing.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=43.04) [Lots of message brokers operate in exactly this way.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=46.04) [Another capability that you might be aware of in existing message](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=50.74) [brokers is the ability to set a time to live,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=54.28) [or TTL, for each message.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=57.7) [This allows you to say that if a message is not read](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=60.54) [after a certain period of time, it can be discarded,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=63.11) [and you can configure this on a per message or a per topic basis.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=66.41) [Finally, there's also an event routing feature currently in preview.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=72.44) [This allows different message types to be sent to different endpoints This](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=76.92) [page in the documentation explains more about this feature.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=82.6) [It includes information about how you can set it up](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=87.14) [declaratively in a YAML file, as you can see here with these match rules,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=89.96) [and also how you can set it up programmatically,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=95.64) [which you can see here in this C# example,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=98.33) [which uses the topic attribute that we saw in our earlier demo,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=101.28) [but with an additional parameter that sets up the routing rule.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=105.82) [This example shows how we could have one endpoint for messages of type](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=110.11) [widgets and another endpoint for messages of type gadgets.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=114.63) [And this makes it much easier in a language like C# to use](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=119.94) [strongly‑typed objects to deserialize those messages into,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=123.5) [rather than having to try to write one generic](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=127.58) [handler for both types of message.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=130.23) [Further down on this page,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=133.34) [you can see some examples of the types of expressions that](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=134.78) [you can configure to select the messages that you want to](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=137.94) [route to different endpoints.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=141.18) [One particularly helpful use case is if you need to](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=143.07) [support multiple versions of a message,](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=146.7) [allowing you to receive old versions on one endpoint and the current versions on a different endpoint.](https://app.pluralsight.com/course-player?clipId=12652703-d016-45d6-a74b-c528939646d0&startTime=149.1)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446)

[In this module,](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=1.14) [we've learned that the Dapr pub/sub building block makes it](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=1.96) [really simple and easy to add asynchronous messaging to](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=5.91) [your microservice application.](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=9.96) [We saw how a simple YAML configuration file allows us to point at](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=12.64) [any one of a number of well‑known message brokers and that we can](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=17.24) [switch easily between them using one for local development and a](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=21.43) [different one in production.](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=25.74) [We saw that Dapr offers two different ways to subscribe to messages on a topic,](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=28.44) [declarative or programmatic, and that if you choose programmatic,](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=33.73) [the Dapr SDK for your language of choice is able to simplify the](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=38.84) [work required to configure your subscriptions.](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=43.34) [In the next module,](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=47.64) [we're going to look in a bit more detail at how Dapr can](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=48.69) [help us to manage secrets securely,](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=51.7) [and we're going to focus in a bit more detail on how we can run Dapr on Kubernetes.](https://app.pluralsight.com/course-player?clipId=d98edfc3-eb6d-4685-84db-44d3e51f3446&startTime=54.44)

## [Accessing Secrets Securely on Kubernetes](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193)

[Hi.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=1.04) [Mark Heath here, and in this module,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=1.5) [we're going to learn about how Dapr can help us access secrets securely,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=3.91) [and we'll be particularly focusing on running Dapr on Kubernetes in this module.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=9.44) [Often in distributed applications,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=15.34) [we need to deal with secrets such as connection strings to](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=17.7) [databases or API keys to access third‑party services.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=21.38) [And I'm sure you're already aware that it's considered bad practice to](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=25.68) [hard code secrets or to check them into source control.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=31.13) [And of course,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=35.64) [there are plenty of existing tools that are designed to](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=36.53) [help us to store secrets securely.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=39.67) [Kubernetes has secret support,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=43.04) [and cloud providers like Azure offer services like](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=45.44) [Key Vault that can store secrets,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=48.56) [and programming frameworks like ASP.NET also have tooling to](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=51.54) [support user secrets in local development.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=55.66) [And you might wonder whether we really need anything additional from Dapr.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=58.03) [However, Dapr's secret management support is worth considering for a few reasons.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=63.74) [First, it's not necessarily a replacement for those services I just mentioned.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=68.47) [Dapr builds on top of existing secret stores behind the scenes and](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=74.74) [provides us with a simplified API to access those secrets.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=79.54) [Second, as we've already seen in this course,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=84.14) [very often we need to make use of secrets inside our Dapr component definition](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=86.84) [YAML files, and the Dapr secrets building block makes it very easy to reference](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=92.62) [a secret from those component files. And third,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=98.59) [one of the key benefits that the Dapr building blocks offer is swappable](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=102.66) [component implementations. By using the Dapr API to access secrets, your](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=107.39) [code for running locally can work in exactly the same way that it runs in](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=113.58) [production, even though they are storing their secrets in completely](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=118.16) [different places. In this module, I'll show you how we can configure](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=121.68) [secret stores both for local development, and we'll see how secrets work](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=126.22) [in Kubernetes. We'll see how to configure a local secret store and use it](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=131.58) [to directly fetch a secret from within our GloboTicket application.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=137.13) [We'll also look in a bit more detail at how a secret can be](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=142.34) [referenced in a component configuration file.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=145.97) [Finally,](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=149.74) [we'll see GloboTicket running in Kubernetes to demonstrate both ways of](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=150.36) [accessing secrets in production. As well as showing you how to access](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=155.06) [secrets with Dapr on Kubernetes, I'll be walking you through all the](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=160.06) [steps that you need to follow if you'd like to run the demo app on a Kubernetes cluster yourself.](https://app.pluralsight.com/course-player?clipId=0f3ea5cd-ba6f-4820-ad89-6203454d9193&startTime=164.08)

### [Dapr Secrets Management Building Block](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b)

[The Dapr secrets management building block works in a very similar](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=1.14) [way to the building blocks we've already seen.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=5.19) [It exposes a simple API that we can call to access a secret](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=7.39) [and supports several backing secret stores.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=13.24) [The URL that we'd call to ask the Dapr sidecar to fetch a secret looks](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=15.66) [like this. Here, we're asking to retrieve a secret called mysecret from a](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=21.57) [secret store called vault. Here in the Dapr documentation, we can see the](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=27.69) [list of supported secret stores.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=33.46) [The two that we'll be using in this course are local file for local](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=36.64) [development and Kubernetes for running on Kubernetes in production.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=40.7) [But you can see that there's also support for HashiCorp Vault, which is a](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=45.64) [very popular secret store, as well as specific cloud stores for different](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=49.37) [cloud providers, such as Azure Key Vault.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=54.23) [As usual,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=58.24) [we can click on any one of these supported secret stores to get some help](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=59) [on the format of the configuration definition file.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=63.81) [Here's an example of the local file SecretStore configuration file,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=67.44) [which you might use for development purposes.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=71.6) [This Dapr component has the name secretstore, and its](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=75.34) [type is secretstores.local.file.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=79.07) [The metadata indicates the location of the secrets file,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=83.24) [but be careful with the path here.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=87.04) [The working directory is not necessarily going to be the](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=89.34) [folder that this configuration file is in.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=92.36) [Instead, it's the folder that we launch Dapr from.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=95.44) [I'm pointing at a local file in the same folder called](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=99.24) [secrets.json. And if we look inside,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=102.48) [we can see that I've simply provided an example secret. Obviously,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=105.57) [if you do use this approach for local development and your](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=110.46) [secrets.json file contains high‑value secrets,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=113.57) [then do be careful not to check that file into source control.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=117.04) [And of course, although this is convenient for local development,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=121.24) [you are free to use any of the Dapr supported secret stores.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=124.79) [For example,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=129.04) [maybe you've got an API key that you want all of your](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=129.89) [developers to share when they're working locally,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=132.83) [but you don't want to store that in a local JSON file.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=135.64) [In that case,](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=139.44) [you might decide to define an additional Dapr secret store that points](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=140.08) [at a secure location, such as an Azure Key Vault, and store that](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=144.63) [high‑value secret in there while still using the convenience of a](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=148.82) [local JSON file for using things like the connection strings to your locally hosted databases.](https://app.pluralsight.com/course-player?clipId=433e212d-06a2-459d-bb48-386db4f2593b&startTime=152.48)

### [Demo: Accessing Secrets Programmatically](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7)

[In this demo,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=1.04) [we're going to update the catalog microservice to request a secret](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=2) [programmatically by calling the Dapr secrets API.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=6.28) [Let's start by reminding ourselves of the configuration for the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=10.64) [localSecretStore that we looked at a moment ago.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=13.95) [It's pointing to a local secrets file called secrets.json,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=17.44) [which contains a single secret called eventcatalogdb.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=20.71) [And the value I've put in here is just an example string that we can use](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=26.54) [to check that we've successfully retrieved a secret.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=30.02) [We're not actually going to be connecting to a real database for this demo.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=33.74) [Let's look at the code that we'd used to access a secret programmatically.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=38.44) [Here in the event catalog microservice,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=42.04) [we're looking at the EventRepository class,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=45.32) [and this is responsible for fetching the list of events.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=49.14) [In a real‑world scenario, this would likely connect to a database,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=52.09) [but in our demo it's just returning an in‑memory list.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=57.24) [However, for the purposes of demonstrating the secrets building block,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=61.24) [let's imagine that we do need to get hold of a connection](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=65.26) [string to allow us to access a database.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=67.8) [And you can see here I'm calling a method called GetConnectionString,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=70.11) [which we'd like to implement using Dapr to fetch](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=74.94) [the secret from the secret store.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=78.02) [So let's see what we need to do to implement this functionality.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=80.64) [To do this, we need to make use of the Dapr .NET SDK again.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=83.55) [Even though we could just construct the URL and talk to the Dapr sidecar](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=88.84) [directly, the Dapr SDK simplifies our code by constructing the URL for us](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=93.57) [and handling the process of correctly parsing the JSON response. Here in](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=100.24) [the catalog microservice csproj file, you can see that I'm referencing the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=105.94) [Dapr.AspNetCore NuGet package. And in the startup code in Program.cs, were](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=111.1) [calling AddDapr to register the Dapr client in the inversion of control](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=118.31) [container.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=123.19) [And this means that back in my EventRepository class I can inject an instance](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=124.54) [of DaprClient into the constructor. And in the GetEvents method, before](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=129.26) [returning the events, I'm asking for that connection string, which we'd need](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=135) [if we were talking to a real database, and I'm writing it to the logs just so](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=139.83) [that we can prove this works.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=144.68) [Let's look at my GetConnectionString method.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=147.44) [Now I've decided to make the secret store name configurable with an](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=151.14) [environment variable, and the reason for that is simply that when we](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=154.94) [run on Kubernetes there's a default secret store called Kubernetes, and](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=158.85) [we're going to be using that later.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=163.82) [But when I'm running locally, it would be a bit strange to use a secret](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=165.84) [store called Kubernetes, so we're going to use the local file secret store](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=169.21) [that we've configured, which is simply called secretstore. As well as a](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=173.66) [secret store name, there's a secret name.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=177.97) [Our secret is just called eventcatalogdb. Now, here's](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=181.44) [where things can get a little confusing.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=186.1) [An individual Dapr secret can contain multiple key‑value pairs,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=188.64) [and this is because some of the secret stores support multiple](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=193.34) [keys in a secret. Kubernetes is actually an example of a secret](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=197.81) [store that can do this.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=202.46) [And this means that to get an actual secret value, such as a connection](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=204.44) [string, we need the secret store name, the secret name, and the key of the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=208.07) [actual secret that we want. To actually access the secret, we call the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=214.32) [GetSecretAsync method, passing in the secretStoreName and the secretName.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=219.87) [Behind the scenes, this is going to make a call to the Dapr sidecar with a](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=224.63) [URL that looks like this.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=229.86) [The GetSecretAsync method returns a dictionary so that secret stores that](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=232.84) [support multiple keys for a single secret are supported. However,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=237.55) [our dictionary will only have one key in it, which is the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=242.1) [secret name again, eventcatalogdb. Okay,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=245.48) [we're ready to try this out.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=249.42) [I'll need to start both the catalog and front‑end microservices, and I'll](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=251.84) [do that in the usual way with my script that uses dapr run to start each](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=256.9) [service. Once both of those are up and running, I'm going to visit the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=261.93) [GloboTicket home page, and that will make a call to the catalog service,](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=266.68) [which, behind the scenes, will fetch the secret that it needs for the](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=271.74) [database. And we can check that it works by looking at the log output for](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=274.94) [the catalog microservice. And sure enough, here we can see the log message which includes the secret value that came from the local file.](https://app.pluralsight.com/course-player?clipId=a095b900-d7ba-49a6-8713-644c09c91eb7&startTime=279.73)

### [Demo: Running On Kubernetes Part 1 - Creating Azure Resources](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a)

[For our next demo,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=1.14) [we're going to see how to use secrets with Kubernetes, but I've not](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=2.38) [yet shown you in detail how I've got GloboTicket running on](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=7.17) [Kubernetes. And so in this first part of the demo,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=10.95) [I'm going to show you how I created the Azure resources that I'm using,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=14.61) [which are an Azure Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=18.94) [an Azure storage account for the state store to use, and](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=21.54) [Azure Service Bus for the pub/sub messaging.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=25.27) [Now,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=28.54) [none of these are requirements for working with Dapr, and so if](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=28.68) [you're not interested in Azure or maybe you've already got your own](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=32.21) [Kubernetes cluster that you want to use,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=35.65) [feel free to skip on to the next part of the demo where I'm going to install the](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=37.97) [GloboTicket application onto Kubernetes. The main tool that I'm using to create](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=42.13) [my resources is the Azure CLI. This provides a really simple command line](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=47.86) [interface for creating and interacting with resources in Azure. You can download](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=53.19) [it here, and it's cross platform,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=58.98) [so you can use these commands on any operating system.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=61.09) [We're looking here at the aks‑deploy PowerShell script that's in](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=65.54) [the root of the demo project. With the Azure CLI,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=70.25) [you start off by calling az login,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=73.45) [which will prompt you to log into Azure if you haven't already,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=76.72) [and if you have multiple Azure subscriptions,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=80.94) [be sure to call az account set to select the one you want to use, as you can](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=83.7) [see with this example. I've already done this, so let's move on. Next, I'm](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=89.07) [declaring a few variables that we'll need. My Azure RESOURCEGROUP, which](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=94.98) [will contain all of the Azure resources I'm creating, will be called](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=100.02) [globoticket‑dapr; and I've chosen the Azure westeurope region, as that's](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=104.08) [near to where I live; and I'm going to use globoticketdapr as the name for](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=109.68) [my AKS cluster.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=115.39) [Now I'm going to run these in my PowerShell session, as we're going to be using](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=117.44) [them for some later commands. And I'm making use of a nice feature in Visual](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=121.77) [Studio Code where I can just press F8, and the lines that I've got selected](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=126.55) [will get sent into my PowerShell terminal.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=131.68) [Now let's look at the commands that create the Azure resources. I've](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=134.84) [actually already created these in Azure with the names that I want, so I](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=139.08) [don't need to run these commands again, but I will show you what each one](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=143.8) [does. First I create the RESOURCEGROUP with az group create with the name](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=147.19) [and location I've chosen.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=153.22) [I can run this one again, and it will do nothing](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=155.44) [because the group already exists.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=158.04) [Then I create the AKS cluster using the az aks create command, and I'm](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=160.74) [putting it inside my RESOURCEGROUP with the name that I've chosen. To](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=167.31) [keep costs down, I've specified an initial node‑count of just 1 node,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=172.1) [and I've also chosen to enable the http\_application\_routing add‑on and](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=177.54) [to generate ssh keys.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=182.73) [The next command, az aks get‑credentials, will update my local](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=185.54) [kubectl configuration file to allow me to use the kubectl CLI tool to](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=190.72) [manage this cluster. I've said that I want to overwrite any existing](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=196.73) [entries for this AKS cluster name.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=201.26) [Let's run this command to ensure that my kubectl context is set up correctly.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=204.64) [If I now call kubectl config current‑context,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=210.04) [we can see that the current context is now pointing at my](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=214.07) [GloboTicket cluster. By the way, kubectl is of course the](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=217.61) [standard command line tool for managing a Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=222) [and if you've used Kubernetes before,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=226.34) [you'll be very familiar with it. Conveniently, if you've installed](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=228.49) [Docker Desktop, then kubectl will be automatically available to](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=232.73) [you, but you can install it separately.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=236.97) [There are two other Azure resources that GloboTicket depends on.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=239.84) [The first is a storage account that's going to be used for the](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=243.69) [Dapr state store building block.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=247.53) [It needs a name, and this has to be unique across](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=250.64) [Azure, so if you're following along, you'll have to pick your own name here.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=253.72) [I've called mine globoticketstate. I then create the storage account with](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=258.24) [az storage account create, specifying the account name,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=263.53) [the resource group and location to put it in, and what pricing tier I want,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=268.14) [and I've chosen the Standard\_LRS. Again, I don't need to run this now](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=273.24) [because I've already created my storage account.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=278.17) [The next two commands you see here are how I can get the storage account](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=281.94) [connection string and key. We're going to need the account key for one of](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=285.98) [the secrets later on, and in order to call the next command, we need to set](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=290.31) [up an environment variable called AZURE\_STORAGE\_CONNECTION\_STRING, and I'm](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=295.54) [setting that up to use the STORAGE\_CONNECTION\_STRING that we just retrieved,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=300.16) [and this allows me to create an Azure Blob storage container inside the](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=304.49) [storage account, and this container is going to be used by the Dapr state](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=309.6) [store.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=313.82) [I've already done this, but here you can see the az](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=314.84) [storage container create command that I used to create a](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=317.8) [container with the name statestore.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=321.41) [There's one final Azure resource we need to create, our Dapr pub/sub](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=324.64) [component is going to use an Azure Service Bus namespace, so I've picked](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=329.7) [a name for that, globoticketpubsub, and again,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=334.93) [if you're following along, you'll need to provide your own unique name here.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=338.87) [The command to create the namespace is az servicebus](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=343.24) [namespace create, and I pass in the name,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=347.08) [the resource group and the location to put it in, as well as](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=350.64) [the pricing tier I want, which is Standard.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=354.37) [Next, I need to retrieve the service bus connection string,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=357.94) [which is a bit trickier, but this az servicebus namespace](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=361.42) [authorization‑rule keys list command can find it for us.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=366.1) [I'm saying that I want the RootManageSharedAccessKey, and I'm filtering](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=371.34) [to just get the primaryConnectionString, and again,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=376.22) [this is because we're going to use this in a secret later on. Now I](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=379.77) [appreciate that if you've not done a lot with Azure,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=384.31) [then maybe some of that went a bit over your head, but don't worry, as](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=386.95) [none of this is necessary to use Dapr. You can work with the cloud](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=390.49) [providers and tools that you're most familiar with and create your own](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=394.95) [Kubernetes cluster using whatever backing services you need for the Dapr](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=399.09) [components that you're using.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=403.73) [But now that I've shown you how I created my Kubernetes cluster in Azure,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=405.84) [along with the backing services for the Dapr components,](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=410.25) [let me show you how we can install GloboTicket onto this Kubernetes cluster next.](https://app.pluralsight.com/course-player?clipId=3b0fa9a2-665b-4248-ac44-d8504226040a&startTime=413.72)

### [Demo: Running on Kubernetes Part 2 - Installing GloboTicket](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee)

[In the previous demo,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=1.14) [I showed you how I created my AKS cluster and updated](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=2.52) [my kubectl context to point at it.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=6.75) [In this demo,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=9.8) [we're going to begin installing GloboTicket onto our Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=11.27) [starting by initializing Dapr on Kubernetes with dapr init ‑k,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=15.97) [and we'll be building our container images and looking at the Kubernetes](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=21.65) [deployment YAML files for the GloboTicket services,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=26.19) [and we'll see how they're annotated to ensure that they](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=30.24) [have the Dapr sidecar injected correctly.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=33.11) [The first thing we need to do, if we haven't already, is to](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=36.94) [install Dapr onto our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=39.8) [Because we've connected kubectl to the cluster,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=42.76) [we can just use the dapr init ‑k command,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=45.94) [which we saw in the first module of this course.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=49.53) [I've actually already installed Dapr onto my Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=52.54) [so it will error if I call this again.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=56.16) [But I can verify that Dapr is installed correctly by calling dapr status ‑k,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=58.45) [and this shows me a summary of the Dapr containers](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=65.19) [that are running on my cluster.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=68.15) [This includes the dapr‑sidecar‑injector,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=70.24) [which is the component that ensures that all of my microservices](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=73.32) [have a Dapr sidecar running alongside them.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=77.02) [Now we do need to containerize our GloboTicket services.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=80.94) [The way that you containerize your application will vary depending on](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=84.39) [the programming language and framework you're using.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=88.93) [But here you can see the Dockerfile for the front‑end](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=91.61) [microservice. This was auto generated by Visual Studio for](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=95.29) [me, and it's completely standard.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=99.69) [There are no special customizations for Dapr necessary. I've got some](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=102.1) [docker build commands here that will build each of the three microservice](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=106.73) [containers using their Dockerfiles and tag them.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=111.33) [I'm prefixing the tags with my Docker Hub username, which is markheath.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=114.94) [So now, the front‑end service has been tagged](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=119.84) [markheath/globoticket‑dapr‑frontend. And I'll do the same for the other two microservices,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=122.83) [building the catalog and ordering microservice](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=130.64) [containers and tagging them. By the way,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=133.73) [in a real production environment,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=137.37) [you'd likely have your own private container image registry that](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=139.4) [you push your microservice container images to.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=144.03) [In Azure,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=147.54) [that would typically be the Azure Container Registry. But I'm using Docker](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=148.23) [Hub here just to keep things simple, and it also allows you to use my](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=153.47) [container images if you'd prefer not to have to build them yourself. I can](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=157.85) [push a container image to Docker Hub with the docker push command using](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=163.01) [that tag that I just created. So, let me push each of these three images to](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=167.69) [Docker Hub. And once that's completed, you can see here on my Docker Hub](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=172.84) [page those images that I've just pushed. Now we need to tell Kubernetes to](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=178.1) [run these containers. In the deploy folder of the sample code,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=183.94) [I have all of our Kubernetes resource definition YAML files.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=189.14) [This includes the definitions of the Dapr components that we're using](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=194.14) [and the definitions of the microservices. Let's focus on the](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=198.28) [microservices first. In frontend.yaml, I'm defining a Kubernetes](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=202.93) [service called frontend, which will be exposed on port 8080 and is of](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=208.58) [type LoadBalancer. By the way,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=213.89) [if you're not familiar with Kubernetes resource definition YAML files,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=216.85) [then these will probably seem a bit intimidating.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=221.04) [I'm not going to be explaining every part of this file,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=224.64) [but I do want to point out a few of the most important things.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=227.32) [This file also defines a Kubernetes deployment called frontend.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=231.94) [Let's start by looking at these annotations on the template metadata.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=237.84) [The dapr.io/enabled annotation, which is set to true, is what tells Dapr to](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=243.04) [automatically inject a sidecar into our Pod. The app‑id property is the](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=249.37) [application ID that's needed for service‑to‑service invocation. The app‑port](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=256.09) [property tells the Dapr sidecar which ports our microservice is listening on,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=261.78) [which is port 80, in this case.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=267.25) [And the final annotation tells the Dapr sidecar to use a configuration](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=270.04) [called appconfig, and this is where we set up the tracing,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=274.66) [which we're going to be looking at in more detail later on in this course.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=278.58) [Further down,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=283.44) [you can see that the container image I'm using is](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=284.12) [markheath/globoticket‑dapr‑frontend, which is the image I just pushed](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=287.12) [to Docker Hub,and I've set the ImagePullPolicy to Always,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=292.15) [which is convenient for development, as I might quite frequently push](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=296.86) [new versions of this container image. And we're saying here that our](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=300.79) [container is listening on port 80, which is the same port that we](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=305.03) [specified in the annotation above.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=309.29) [Now let's look at the catalog microservice YAML file. This is simpler.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=312.44) [I didn't declare a service resource, just a deployment.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=317.72) [It's got the name catalog and has also got the dapr.io annotations.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=321.84) [But you'll also notice that I've set an environment variable on](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=327.84) [this container, which sets the SECRET\_STORE\_NAME.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=330.95) [And that's because, if you remember earlier, we decided to make the](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=334.54) [SECRET\_STORE\_NAME configurable by an environment variable so that when we](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=337.97) [run on Kubernetes, we can choose to talk to the default Kubernetes secret](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=342.61) [store that gets created automatically.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=347.21) [And there's also a YAML file here for the ordering microservice, and it's](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=350.74) [very similar to the other two that we just looked at.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=355.01) [In the next part of this demo,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=359.04) [I'm going to show you how we can use the kubectl apply command to](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=360.52) [install these microservices onto our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=364.79) [But before we do that,](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=369.24) [we're going to create some secrets in Kubernetes and see how our Dapr component definition files can reference those secrets.](https://app.pluralsight.com/course-player?clipId=49bbd908-f84e-449b-a554-5588a605d9ee&startTime=370.58)

### [Demo: Running On Kubernetes Part 3 - Secrets In Component Config](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c)

[In this demo, we're going to create some Kubernetes secrets,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=1.14) [and then we're going to see how we can access those secrets](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=5.24) [from a Dapr component configuration file.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=8.54) [And we'll wrap up by actually deploying our microservices](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=11.84) [and our Dapr components into our Kubernetes cluster using](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=15.32) [the kubectl apply command.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=19.38) [First of all,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=22.24) [let me show you the commands that I used to create my Kubernetes secrets.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=22.75) [I've created three with the kubectl create secret generic command.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=27.54) [The first is called blob‑secret,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=33.24) [and it contains a single key value pair where the key is account‑key,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=35.54) [and the value is the STORAGE\_ACCOUNT\_KEY that we](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=40.24) [retrieved earlier on with the Azure CLI.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=43.05) [And we're doing a similar thing with the service bus secret with](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=46.74) [the connection string of the service bus being made available in](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=50.28) [a key called connection‑string.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=54.04) [And finally,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=56.84) [the secret that we're accessing programmatically from the](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=57.83) [catalog service just has an example value.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=60.86) [I've put Event Catalog Connection String from Kubernetes,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=64.94) [and I've used the same value of eventcatalogdb for both the secret name and](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=69.04) [the key because that's consistent with how the local file secret store works](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=74.43) [which were using when we're running locally.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=79.29) [With these secrets in place,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=82.44) [our Dapr component configuration files can make use of these secrets.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=84.09) [For example,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=89.34) [here in the azure‑pubsub.yaml file where we set up Azure service bus as](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=89.98) [the message broker for the pubsub building block,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=95.52) [rather than hardcoding the connection string,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=98.82) [under value,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=102.04) [we have this secretKeyRef property that has a name of servicebus‑secret,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=103.09) [which is the Kubernetes secret name and a key of connection‑string,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=109.74) [which is the name of the key inside that Kubernetes secret.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=115.04) [Remember, we said a single Kubernetes secret can contain multiple keys,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=119.74) [even though we're only using one in this example.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=125.36) [We also need to tell it which secret store to use.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=129.44) [So here, we're just saying Kubernetes,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=132.56) [and this secretStore is set up by default for us when we](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=135.64) [enable Dapr on a Kubernetes cluster, which makes it really convenient to use,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=139.58) [but obviously, if you prefer to use a different secret store,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=145.04) [you could point it there instead.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=148.55) [Now currently,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=150.94) [we haven't pushed any of these component definition](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=151.59) [files into our Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=154.17) [and we can do that quite simply with the kubectl apply command.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=156.44) [Here,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=162.24) [you can see the command to apply the pubsub and](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=162.49) [statestore component definitions.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=165.97) [By the way,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=169.04) [this script also contains code to deploy a number of](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=169.55) [additional components for things like bindings and](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=173) [observability that we'll be using later.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=176.12) [And if you want to install GloboTicket yourself,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=179.64) [then you will need to complete these steps here as well that installs Zipkin,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=182.43) [the cron component, and the MailDev service,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=186.82) [and I'll be explaining more about those later on in this course.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=190.84) [The final thing we need to do to install GloboTicket onto our Kubernetes](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=195.04) [cluster is to install the three microservices themselves,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=199.19) [so we'll do that with kubectl apply pointing at the YAML files for each of](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=203.84) [those three microservices that we looked at earlier.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=208.74) [And I've shown some additional helpful commands here that you can use to](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=212.54) [check that everything is running correctly on your cluster,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=216.37) [as well as examples of how to do things like restart one of](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=219.74) [the microservices or examine the logs,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=223.22) [which can be very useful when you're troubleshooting.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=226.43) [Let me just run this command to see the Kubernetes deployments, for example.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=229.64) [And here, you can see the three microservices that I've installed,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=234.44) [as well as the MailDev and the Zipkin containers that](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=238.44) [we'll see in action later on in this course when we](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=241.73) [implement bindings and observability.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=244.52) [And if like me you're using the Azure Kubernetes](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=247.64) [service to host your Kubernetes cluster,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=250.49) [then this command will launch the Azure portal and navigate to the AKS cluster,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=253.64) [which allows you to explore all of the resources configured in your](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=259.14) [Kubernetes cluster and to check on the health.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=262.78) [Okay, now I do appreciate that was quite a lot to go through,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=266.64) [and if you're new to Kubernetes,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=270.24) [it might seem overwhelming as Kubernetes itself is a very](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=272) [powerful platform with lots of capabilities,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=275.74) [but once you do become familiar with Kubernetes resource definitions,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=279.24) [then you'll recognize that adding Dapr to a Kubernetes microservices](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=283.74) [application is actually relatively straightforward.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=287.61) [In the next demo,](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=291.74) [we're going to check that this is all working as expected and that we can access secrets.](https://app.pluralsight.com/course-player?clipId=666cc7fe-3361-4a26-9c10-8cb212cd270c&startTime=292.64)

### [Demo: Running on Kubernetes Part 4 - Accessing Kubernetes Secrets](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3)

[In this final part of our demo,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=1.04) [we'll see the Dapr secret access in action on Kubernetes](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=2.99) [and test that it's all actually working.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=7.56) [Basically, if we've set up everything right,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=10.74) [our state store and pub sub components will be able to use the](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=13.18) [Azure resources that they're pointing at,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=18.15) [and our catalog service will be able to fetch the connection](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=20.75) [string secret from the Kubernetes secret store.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=24.22) [I've included a useful script that helps us find the IP address](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=28.04) [of the frontend service running in Azure,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=32.02) [and I'm using PowerShell to launch a browser to that address on port 8080.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=35.04) [And we can see it's loaded successfully,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=40.74) [I can see the list of events, and I'll put something in the basket,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=43.42) [and I'll quickly go through the whole purchasing process](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=47.27) [entering some data about who has placed the order.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=50.86) [And once I've completed that,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=54.84) [we're going to go back and view the logs to check that](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=56.48) [our secrets are working as expected.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=59.37) [In Kubernetes, to get the logs, you need to know the pod name,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=62.54) [and again,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=66.44) [I've added some helpful commands here to get the pod](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=67.06) [names for each of the microservices.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=69.92) [So I'll run these,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=72.94) [and now I can use the kubectl logs command to ask to](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=75.24) [see the logs from the catalog pod, and in particular,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=79.08) [from the container in that pod called catalog.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=83.04) [If we look at these logs,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=86.74) [we can see that the catalog service was able to](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=87.97) [programmatically access the Connection string secret that](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=91.19) [we added to the Kubernetes secret store.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=94.87) [And if we view the log output from the ordering microservice,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=97.74) [we can see that we received a message about the new order,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=101.64) [and that proves that the pubsub component was successfully able to connect to](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=105.24) [Azure Service Bus to publish and subscribe to that topic,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=110.15) [which means that it was able to access the connection string secret.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=114.42) [That's almost it for this module, and if you've been following along in Azure,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=119.04) [you may now want to delete the resource group that you](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=123.6) [created with the az group delete command,](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=126.12) [and that will mean that you're no longer paying for any of the Azure resources that we created in this demo.](https://app.pluralsight.com/course-player?clipId=7b619c38-cbc3-475d-91fd-5664bb17c5c3&startTime=129.48)

### [Secret Scoping and Module Summary](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e)

[Before we wrap up what we've learned in this module,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=0.94) [I want to highlight one additional feature of the Dapr secrets building](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=3.95) [block, and that's secret scoping. In a large application, we might have](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=8.17) [many secrets and some of those secrets are specific to a single](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=13.79) [microservice. Not every microservice will need access to every secret,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=17.93) [and if we want to follow the security best practice of principle of](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=24.04) [least privilege,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=28.52) [then we only want to allow microservices to access the secrets](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=29.94) [that they actually need, and this page in the official Dapper](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=34.09) [documentation gives some helpful examples showing how to achieve](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=37.98) [different scoping rules.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=42.91) [As you can see in this example, we're only allowing access to](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=45.44) [two specifically named secrets and denying access to all other](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=49.17) [secrets, so for production scenarios,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=53.93) [you might want to consider adding this additional level of](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=57.37) [security to your secret configuration.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=60.25) [In this module, we looked at the Dapr secrets management building block.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=63.84) [We saw how Dapr supports a number of underlying secret stores that your](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=68.15) [component definition can switch between.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=73.36) [In particular,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=76.44) [we used the local file secret store for local development and the](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=77.32) [Kubernetes secret store when we were running on Kubernetes. We saw how](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=81.68) [secrets can be accessed either from within a component definition file](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=86.66) [with the secretKeyRef property or programmatically by calling the Dapr](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=91.14) [sidecar and asking for them, and we used the .NET Dapr SDK to simplify](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=96.66) [doing that.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=102.72) [We also spent quite a bit of time looking in more detail at exactly how we](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=104.14) [can get a Dapr application running on Kubernetes using kubectl apply to](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=108.57) [configure our microservices and Dapr components.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=114.51) [In the next module,](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=118.44) [we're going to see how the Dapr bindings building block can greatly simplify integrating with other services.](https://app.pluralsight.com/course-player?clipId=8ed44498-cab8-4d5c-94b7-5ae336db245e&startTime=119.59)

## [Integrating with External Services Using Bindings](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb)

[Hi.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=1.04) [Mark Heath here, and in this module,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=1.5) [we'll see how the Dapr bindings building block can greatly simplify](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=4.15) [the task of connecting to third‑party services.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=9.11) [To help us understand what Dapr bindings do,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=11.91) [let's remind ourselves of what happens when we use the pub/sub building block.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=16.14) [Suppose microservice A wants to publish a message and](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=21.19) [microservice B wants to subscribe to it.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=26.23) [Thanks to the pub/sub building block,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=28.67) [all microservice A has to do is make a simple HTTP request to its Dapr](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=31.35) [sidecar, and the Dapr sidecar handles all the complexity of actually](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=37.73) [sending that message to a message broker,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=43.32) [whether that be RabbitMQ or Azure Service Bus or any one of](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=46.24) [the supported pub/sub message brokers.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=50.76) [And this has the benefit of greatly simplifying the code in our](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=53.6) [microservice. We don't need to know about all the complexities of how to](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=56.76) [use the SDK for the particular message broker we're using. And what about](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=61.34) [the code that subscribes to the topic?](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=66.63) [Well, we saw that, again,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=69.44) [what happens is that the Dapr sidecar takes care of actually communicating](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=71.2) [with the message broker and subscribing to the topic. And when it receives a](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=76.64) [message, it passes it on to microservice B by calling an HTTP endpoint. And](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=81.7) [this keeps microservice B very simple. It only needs to expose an endpoint](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=88.81) [and doesn't need to deal with any of the complexities of connecting to the](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=94.64) [message broker.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=99.22) [Now,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=101.04) [what's this got to do with bindings? Well, what bindings in Dapr do](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=101.34) [is it takes this general principle of the Dapr sidecar communicating](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=106.57) [with external services on our behalf and expands it to support a](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=111.48) [much wider range of services.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=116.55) [For example, there are bindings for services that can send emails.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=119.14) [So if we use that binding,](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=122.81) [our microservice could send emails very easily just by posting a](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=124.39) [message to the Dapr sidecar. And just like we saw that the Dapr](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=128.76) [sidecar can call us when a message appears on the service bus, this](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=133.09) [principle can also be expanded so that Dapr bindings can trigger](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=138.04) [requests to our microservices whenever something of interest happens](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=142.43) [in an external service.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=147.12) [So in this module, we're going to look at the Dapr bindings building block](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=149.54) [and understand the concepts of input and output bindings.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=153.84) [We'll look at some examples of how to configure bindings to various](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=159.04) [external services, and then we'll see them in action by using them in](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=162.85) [GloboTicket. We'll configure an output binding to allow us to send an](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=167.37) [email whenever we receive a new order, and we'll also look at the cron](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=172.34) [input binding, which gives us a really convenient way of implementing scheduled tasks.](https://app.pluralsight.com/course-player?clipId=07f0ab0c-f924-4107-afa5-67c2bf1659bb&startTime=177.62)

### [Supported Dapr Bindings](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b)

[If we take a look here at the official Dapr documentation,](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=1.24) [we can see a list of the currently supported bindings.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=5.04) [It's quite a long list, and one of the things that you might](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=9.44) [notice is that bindings can be input or output.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=12.25) [And output bindings are the most common.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=16.64) [An output binding means that we make a request to the Dapr sidecar,](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=19.74) [and it forwards on that request to the external service. And an input](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=24.07) [binding means that the Dapr sidecar will make a call to our](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=30.02) [microservice letting us know whenever something interesting has](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=33.73) [happened on the external service.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=37.53) [So we can be notified about events and they can be used to](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=40.24) [trigger endpoints on our microservices.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=43.65) [You can see here that in addition to multiple generic bindings,](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=46.84) [there's also several bindings supporting services on specific](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=51.94) [cloud providers, like Azure or AWS. Many of the most popular](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=55.92) [external services that you might use when building microservices](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=61.02) [are already in this list.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=65.06) [And given that Dapr is an open source project,](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=67.19) [this list continues to grow as members of the community](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=70.07) [contribute bindings for additional services. To help us](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=73.52) [understand how you'd use these bindings, let's quickly take a look at some component configurations.](https://app.pluralsight.com/course-player?clipId=e007725b-1e24-42a8-9bd5-cb9d6a48243b&startTime=77.88)

### [Output Binding Example (SendGrid)](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a)

[for our first example binding, let's choose an output binding.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=1.04) [We're going to look at the SendGrid binding.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=5.24) [SendGrid is a service that allows you to send emails,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=8.24) [and here we can see the example component definition.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=12.14) [Like all Dapr components,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=15.94) [it's got a name and the type is bindings.twilio.sendgrid.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=17.56) [The metadata shown here allows us to optionally configure some default values,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=23.54) [such as the email sender and recipient addresses and subject,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=28.64) [but the one required configuration property is your SendGrid API key,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=33.64) [and this is of course a secret,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=38.7) [so it would be a good idea to set this value using a Dapr secret](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=40.46) [reference like we saw earlier in this course.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=44.43) [How would we actually send an email using this binding?](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=48.34) [Well, we'd need to send a request to our Dapr sidecar,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=52.24) [which would have a URL looking a bit like this](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=55.54) [calling the Dapr sidecar on localhost,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=58.84) [using version 1 of the bindings building block,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=62.14) [and talking to the binding named sendgrid.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=65.54) [And then for the body of the request,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=69.34) [this documentation gives us an example of the data that](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=71.64) [we need to pass to the output binding.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=74.94) [The request payload contains a data property,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=77.74) [which for this binding is the HTML body of the email that we want to send.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=81.24) [And there is also some optional metadata we can add,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=86.54) [and this example is showing setting the recipient of](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=90.14) [the email and the email subject, and although the documentation doesn't show it,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=92.85) [we also need to specify an operation in this payload.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=98.21) [As you can see here, the sendgrid binding only supports the create operation,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=102.44) [which you used to create an email.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=107.84) [And once again,](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=110.74) [the Dapr SDK can help us out a lot here by formatting the URL for us and](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=111.79) [correctly constructing the request payload with the data, metadata, and operation.](https://app.pluralsight.com/course-player?clipId=16416439-5861-4d33-9e8d-4db43755795a&startTime=117.33)

### [Interchanging Bindings](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b)

[One really nice capability that many of the Dapr building](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=1.04) [blocks offer is the ability to interchange between](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=4.56) [different backing implementations.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=8.03) [We saw this with the state store building block,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=11.04) [where in local development we used a Redis instance running in Docker,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=13.84) [and when we ran in production, we used Azure Blob storage for our state store.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=18.14) [And the microservice using the Dapr state store building block doesn't](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=23.09) [need to know or care what the actual backing store is.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=28.03) [The code remains exactly the same even when we switch component definitions.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=31.94) [Now, does this apply to Dapr bindings as well?](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=37.05) [Well, actually,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=41.34) [in the case of email bindings that we're going to be using in this module,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=42.18) [it is a very similar story.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=45.89) [We might bind to SendGrid for production and to the](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=48.14) [SMTP binding for local development,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=51.62) [and the same message could be sent to the Dapr sidecar and the](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=54.74) [bindings would deal with the complexities of actually forwarding](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=58.27) [those emails on to the email sending services.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=61.52) [However,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=65.64) [it's not necessarily the case that all external services](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=66.1) [offer the same set of capabilities.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=69.99) [The Dapr bindings for different services might expect different](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=72.84) [metadata in the messages that we send to them,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=76.69) [so be careful of assuming that you can always trivially switch between bindings.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=79.82) [The important thing to remember is that the benefits of Dapr](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=86.14) [bindings are the simplification of the code that we need to write](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=89.49) [to connect to those external services.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=93.44) [All the complexities of connecting and the knowledge of how to](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=95.91) [follow best practices when you're talking to those services is](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=100.11) [contained within the binding, and so even when you might want to](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=103.81) [switch between binding to two services,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=107.56) [but don't have exactly the same capabilities,](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=109.95) [your code can still benefit greatly from the simplicity](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=112.43) [that comes from using Dapr bindings. Let's look next at an example of a Dapr input binding.](https://app.pluralsight.com/course-player?clipId=651e5881-09cf-4664-b9c3-e744e739304b&startTime=116.23)

### [Input Binding Example (Twitter)](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc)

[Now let's look at an example of an input binding, and I've picked Twitter.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=1.04) [The Twitter binding is actually both an output and an input binding.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=6.68) [When you're using it as an output binding,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=11.11) [you can use the binding to send tweets,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=14.06) [but for an input binding you can configure search criteria that's looking for](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=17.24) [all new tweets that contain a specific phrase, and then it can notify you](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=22.15) [whenever someone posts a tweet that matches that phrase.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=27.12) [And the documentation page that we're looking at here](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=31.74) [shows us the component definition format.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=34.3) [There's a bunch of Twitter keys and secrets that we need to](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=37.74) [provide in order to be able to connect.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=40.69) [But let's look at the input binding part.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=44.14) [All you do is you provide another metadata property called query,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=47.14) [and the value is set to the term you're searching for.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=52.44) [In this example, we're searching for all tweets that mention Dapr.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=55.64) [And what happens is that the Dapr sidecar connects to Twitter using the](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=59.94) [credentials that you supplied and registers for a webhook callback from](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=64.03) [Twitter whenever a tweet containing this term is posted. And then the webhook](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=68.2) [will be received by your Dapr sidecar and then it will pass it on to your](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=73.91) [service by calling an endpoint that matches the name of your component, so if](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=78.45) [our component was called Twitter, then the Dapr sidecar would call an endpoint](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=83.85) [of /twitter. The format of the body of the request is defined by the binding](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=88.3) [itself.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=94.12) [In this case, it's simply passing on the definitions of the tweets received](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=95.64) [from Twitter, which is actually a fairly complex structure that's defined](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=100) [here in the Twitter developer documentation.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=105.11) [So we've seen that Dapr bindings can simplify the task of connecting to](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=108.74) [external services, as well as receiving notifications from external services](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=113.34) [that can trigger behavior on our microservices.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=119.74) [It's important to remember though,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=123.44) [just like with all Dapr building blocks, bindings are optional.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=125.26) [Just because a Dapr binding is available,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=129.74) [doesn't mean you have to use it. Sometimes,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=131.73) [particularly if you're an advanced user,](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=134.89) [you may still prefer to directly use the SDK of the service that you're](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=137.43) [talking to, as the Dapr bindings won't always necessarily expose the full](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=142.01) [capabilities of the services that they are connecting to.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=147.64) [Let's put Dapr bindings into practice by using them in the GloboTicket application.](https://app.pluralsight.com/course-player?clipId=184bdb11-34bf-466f-993b-9c971f92a5cc&startTime=151.94)

### [Demo: Output Binding - Sending Email with SMTP](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1)

[In this demo, we'll see how a Dapr output binding can be used to send emails.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=1.04) [We're actually going to use the Dapr SMTP binding to implement this.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=7.34) [We'll update the ordering microservice to send an](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=12.44) [email whenever tickets are purchased.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=15.26) [First, let's take a look at the component definition we're going to use.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=18.64) [Here in email.yml, I'm defining a component whose type is bindings.smtp,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=23.44) [and the name I'm giving to my component is sendmail,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=31.14) [and we need to supply a few details.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=34.94) [First, what's the host name of the SMTP server that we want to connect to,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=38.14) [which in our case is localhost, and the port it's listening on,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=43.04) [which is 1025.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=47.51) [We need to supply the username and password,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=49.64) [which normally would be set up as secrets,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=52.74) [but because this is the component definition for local development and we're](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=55.39) [not actually connecting to a real email sending service,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=59.32) [in this case, hardcoded values are acceptable.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=62.74) [Notice the scopes section at the bottom.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=66.44) [Sometimes in Dapr,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=69.44) [we want our components to be available to all our microservices,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=70.56) [but in this case,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=74.74) [we only want the ordering microservice to access this component,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=75.5) [so I've set the scope to be simply ordering.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=80.14) [For local development,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=83.64) [it would be really useful to have an SMTP server we can use,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=84.82) [and I've chosen to use an open source component called MailDev.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=88.84) [This combines an SMTP server and a web interface that allows](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=93.24) [us to view the emails that have been sent,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=98.14) [and this is really convenient for local development.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=100.84) [We can start it locally by using the docker run command](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=104.24) [to start the maildev/maildev image,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=107.83) [and mapping ports 80 and 25 to 3 port numbers.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=111.64) [In this example, I'm using 1080 and 1025.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=116.64) [Now let's look at the code to actually send an email](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=121.64) [using the Dapr output binding.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=124.81) [I've put all the code that sends an email here in the EmailSender class.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=127.54) [As you can see, in the constructor,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=132.64) [we're taking a DaprClient instance as we want to make use of the](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=134.67) [Dapr C# SDK to simplify sending the email.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=138.86) [Then in the SendEmailForOrder method,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=144.04) [we construct a dictionary containing the metadata for this](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=146.94) [email which includes the sender and recipient email](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=150.33) [addresses and the email subject line.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=154.25) [You might notice that these are the exact same](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=158.04) [metadata names for the sendgrid binding,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=160.31) [which is nice.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=162.91) [So in this particular example,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=164.36) [we could swap out an SMTP binding component definition for a](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=166.19) [sendgrid component binding definition,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=170.75) [and there'd be no required code changes at all.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=173.31) [And the body contains the HTML formatted message content.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=176.54) [Finally,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=181.14) [here we're sending the request to our Dapr sidecar for](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=181.79) [it to pass it on to the SMTP server,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=185.26) [and we're using the InvokeBindingAsync method on the Dapr client.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=188.34) [The first parameter is the binding name which we called sendmail](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=194.04) [and then the operation which is simply create.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=199.14) [Then we pass in the data, which for this binding is the email body,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=202.94) [and finally, any additional metadata, which is our dictionary in C#.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=208.14) [And so the Dapr client has saved us from having to construct](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=214.34) [the URL and HttpRequest payload ourselves.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=217.46) [If we now take a look in the OrderController class which contains the endpoint](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=222.64) [that receives a new order from the Dapr pubsub component,](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=227.4) [we're now making a call to the SendEmailForOrder method on the emailSender.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=231.94) [Okay, so now I've shown you all the code that will send an email for an order, let's test it out.](https://app.pluralsight.com/course-player?clipId=bfbef0da-c5e2-406d-ac9e-7f3a90ae2af1&startTime=237.84)

### [Demo: Testing Sending an Email](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508)

[In this demo,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=1.04) [we'll put the code that we just looked at for the output binding](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=2.05) [into action by sending an email and then checking that our](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=5.59) [maildev Docker container picks it up.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=10.06) [And I'll also show you how I've configured the same](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=12.59) [maildev service in Kubernetes.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=15.55) [I've already started the maildev container locally](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=18.54) [using the command that we saw earlier.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=21.34) [And so we need to start all three microservices,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=24.34) [which we'll do in the same way that we've done before,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=27.23) [with the start‑self‑hosted PowerShell script in each microservice folder.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=30.24) [Once they've started up, let's visit the GloboTicket home page,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=35.84) [and we'll add something into our basket and we'll](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=40.04) [complete the whole checkout process.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=43.39) [And once we've got to the point where we've submitted the order,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=45.81) [a message is going to be sent to the ordering](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=49.62) [microservice using the pub/sub building block,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=52.1) [and that's going to trigger our output binding that should have sent an email.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=55.64) [So let's check it worked by visiting our MailDev web server.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=60.34) [We exposed it on localhost port 1080,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=65.04) [and here we can see that the email has been passed on to](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=68.54) [our SMTP server by the Dapr binding, and it's got the correct subject,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=71.31) [body, and recipient.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=76.4) [By the way, if you want to try this out in Kubernetes,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=78.3) [you can also install the maildev container onto a Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=82.1) [And if we take a look in the aks‑deploy script that we've gone](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=86.58) [through in some detail earlier in this course,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=90.9) [you can see here this kubectl create deployment command that](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=93.64) [deploys the maildev container image and the kubectl expose](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=97.78) [command that makes ports 25 and 80 available.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=102.66) [And the binding component definition for maildev that](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=106.66) [we're using in Kubernetes, you can see here in the](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=110.06) [email.yaml file in the deploy folder,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=113.63) [and it's exactly the same as the one we used in self‑hosted mode,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=117.24) [except now that the host is maildev rather than localhost.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=121.88) [Finally, back in the aks‑deploy script,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=127.14) [here's where we deploy that email.yaml component into our Kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=130.11) [And I've also included a helpful kubectl port‑forward command that you](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=135.81) [can use if you want to access the maildev web server on a Kubernetes](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=141.21) [cluster from your local development machine.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=145.88) [You just run the kubectl port‑forward command, mapping the maildev](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=149.24) [service onto a local port, I've chosen 8081,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=153.36) [and then if you visit localhost port 8081,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=157.52) [you'll see the same maildev UI that we just saw.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=161.16) [Let's quickly run that port‑forwarding command,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=165.44) [and if I visit port 8081,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=167.94) [then I can see the MailDev instance that's running on my Kubernetes](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=170.89) [cluster in Azure, and you can see that there's a few email notifications](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=174.59) [in here from tests that I've done previously.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=179.46) [Of course, in production you wouldn't use something like maildev.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=182.38) [You'd either point this SMTP binding at a real SMTP](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=186.86) [server that was going to send emails,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=190.74) [or you'd make use of an email sending service like the SendGrid](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=192.85) [output binding that we looked at earlier.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=196.78) [And the great thing is that we can change the binding](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=199.74) [destination with no code changes whatsoever.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=202.07) [In this demo, our code is simply talking to a binding called sendmail,](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=206.24) [but it doesn't need to know or care what the actual email sending service behind the scenes is.](https://app.pluralsight.com/course-player?clipId=be3774be-988e-4c28-b9ec-a90a1836e508&startTime=210.94)

### [Demo: Cron Input Binding](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a)

[For our input binding demo, we're going to use the cron binding,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=1.04) [and this allows us to run scheduled tasks.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=5.54) [Now this is a bit of a special case,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=8.84) [because this binding isn't connecting to any specific external service.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=10.91) [However,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=16.44) [scheduled tasks are an extremely common requirement in microservice](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=17.03) [applications, and so the fact that Dapr provides a way for us to implement](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=21.18) [them with bindings is potentially a big time saver.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=25.52) [We're going to create a very simple scheduled task in the](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=30.14) [catalog microservice as an example.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=33.2) [Let's imagine that GloboTicket want to randomly pick an event to go on special](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=36.15) [offer every hour in order to encourage customers to keep logging onto the](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=41.58) [website to see what the current special offer is.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=45.85) [Let's start by looking at the component definition file for our input binding.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=50.44) [It's pretty straightforward.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=55.64) [Our component is of type bindings.cron and it has a name of scheduled.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=57.54) [The name is important,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=63.24) [because the Dapr sidecar is going to call an endpoint with this name on our](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=64.48) [microservice every time that the scheduler fires. We obviously need to tell it](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=69.24) [how often to fire, and the cron binding supports regular cron job expressions,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=74.64) [which are very flexible. And there's also a simplified syntax available that](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=80.31) [I'm taking advantage of here.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=85.56) [I've set mine up to run every 5 minutes, as I don't want to wait ages](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=88.04) [while I'm testing this. Earlier in this module,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=92.36) [we talked about the scopes section of the component definition file, which](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=95.8) [lists the microservices that this component applies to.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=100.51) [It's only the catalog microservice that has an endpoint for the scheduler to](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=104.34) [call, so I've just put catalog here. Remember that when we start our](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=108.15) [microservices in self‑hosted mode, we're pointing at this components folder to](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=113.56) [say that this is where all of the component definitions for our microservices](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=118.54) [are, and so even though all three of our microservice sidecars will see this](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=122.66) [YAML file because it's in the folder, only the catalog microservice will](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=127.36) [actually use it because the application name of the catalog service is listed](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=132.23) [in the scopes here.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=137.91) [Now let's look at the endpoint that's going to be called when the](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=139.84) [scheduler fires. I've created a new ASP.NET controller in the](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=142.76) [EventCatalog microservice called ScheduledController.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=147.91) [It's very simple to set this up.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=152.34) [The Dapr sidecar will simply make a post request to the /scheduled](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=154.74) [endpoint, and so I've said that this controller is listening on the base](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=160.72) [path of scheduled, and this OnSchedule method will accept HttpPosts to that](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=165.26) [endpoint. Inside this method, I can then do whatever I want to on the](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=171.87) [schedule, such as triggering the logic to put one of our products on](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=176.6) [special offer, as you can see in this example, and I'm also writing a](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=180.8) [message to the log so that we can easily check that this endpoint is](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=185.61) [getting called correctly.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=189.6) [We're ready to test this, and we don't even need all of our microservices](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=191.84) [running. I'll just start the catalog microservice, and I'll wait a couple of](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=195.42) [minutes for the scheduler to fire. And sure enough, here in the logs we can](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=200.55) [see the message that shows our scheduled endpoint has been called. And if I](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=205.38) [start the frontend service and visit the home page, we can test that it's all](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=211.41) [working by waiting a few minutes, and now when I refresh the page, we should](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=215.56) [see some prices change as a different event has been placed on special offer.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=221.08) [One nice thing about the cron binding is that to run it on Kubernetes,](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=226.94) [we don't need to make any changes whatsoever to our component definition file.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=231.16) [So here in the deploy folder, which defines the components we're using on AKS, I've got exactly the same cron.yaml file.](https://app.pluralsight.com/course-player?clipId=6efb503d-0d9a-45a7-ab50-6ade8db1217a&startTime=236.14)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c)

[In this module,](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=1.04) [we looked at Dapr bindings, which can greatly simplify the code](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=1.99) [required to communicate with external services.](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=6.47) [We saw that there's two types of binding, input and output bindings.](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=10.54) [Input bindings allow events that happen in external services](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=15.94) [to trigger requests into our microservices.](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=20.06) [And we looked at examples of Twitter search results and scheduled tasks. And,](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=23.84) [output bindings allow us to send data to external services, and the examples we](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=29.72) [looked at were sending emails with SendGrid and with an SMTP server. And, these](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=35.16) [are only a small selection of examples.](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=41.59) [Dapr comes with many bindings to popular services,](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=44.14) [and because it's an open source project that accepts community contributions,](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=47.68) [that list is continuing to grow. Right, we're almost](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=52.53) [at the end of this course now,](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=56.57) [but there is one more building block that I want to show you and](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=58.04) [that's how Dapr can help us to monitor what's happening in production with the observability building block.](https://app.pluralsight.com/course-player?clipId=afdf9f5d-007f-4634-8f59-029c90fbbe7c&startTime=61.28)

## [Monitoring in Production with Observability](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a)

### [Module Introduction](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a)

[Hi.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=1.04) [Mark Heath here, and in this module,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=1.5) [we're going to learn how the Dapr observability building](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=4.03) [block can help us understand exactly what's going on in](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=7.64) [our microservices application.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=12.11) [One of the biggest challenges associated with operating a microservices](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=14.94) [applications in production is that you need to be able to understand the big](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=19.05) [picture of the overall health and performance of the system.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=23.82) [You need to be able to both detect problems quickly so that](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=28.84) [you can fix them and to troubleshoot issues to understand](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=32.43) [afterwards what went wrong.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=36.58) [And once again,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=39.24) [Dapr is able to help us by providing a variety of](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=40.41) [capabilities to give us greater insight into the health and](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=43.96) [performance of our overall application.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=48.46) [In this module, we'll start off by taking a look at several](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=51.84) [helpful observability capabilities provided by Dapr,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=55.15) [including distributed tracing, the Dapr dashboard,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=59.74) [health checks, metrics, and sidecar logging,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=64.34) [and then we'll see how to configure our GloboTicket application to](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=69.64) [collect tracing information that we can view with Zipkin. And we'll](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=73.45) [wrap up by taking a look at some of the additional capabilities of](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=78.16) [Dapr that you might want to explore,](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=81.78) [as well as learning about some resources that will help you in your](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=84.34) [journey to adopt Dapr for your own applications. So let's get started.](https://app.pluralsight.com/course-player?clipId=aa24996d-cace-4b32-aa42-d638d88f413a&startTime=87.78)

### [Distributed Tracing](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e)

[We'll start off by looking at the distributed tracing capability of Dapr.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=1.14) [Tracing allows us to understand the way that our](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=5.32) [microservices are interacting with each other,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=9.02) [and it also gives us a way to correlate the logs from different microservices](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=12.84) [in order to piece together what happened when we're trying to understand an](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=17.47) [operation that involves more than one microservice.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=21.56) [By default,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=24.44) [Dapr uses the Zipkin protocol for distributed traces and metrics collection.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=26.69) [The Zipkin protocol is widely supported,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=32.84) [meaning that you can choose from multiple different back ends,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=35.84) [including Zipkin, Stackdriver, and New Relic.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=38.89) [Dapr also supports the OpenTelemetry protocol, allowing you to export your](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=43.44) [traces to many other back ends, including Azure Monitor,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=48.94) [Datadog, and SignalFx.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=52.86) [And the way it works is that the Dapr sidecar contains tracing](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=55.84) [middleware. This means that all communications via the Dapr side car](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=60.26) [will automatically be eligible for tracing.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=65.13) [That means if you're using the Dapr building blocks that we've](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=68.94) [looked at in this course, you just need to turn on tracing, and](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=71.93) [you'll get a very comprehensive picture of what's going on in your](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=75.75) [application with minimal effort.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=79.61) [Obviously, if you're connecting to other services without using Dapr,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=82.74) [then it's up to you to ensure that you configure](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=86.95) [tracing for those communications.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=89.17) [The way you set up tracing in Dapr is via a configuration file similar to](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=92.34) [the component definition files that we've looked at before. Here we can](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=97.54) [see an example of the configuration that I'm using in Kubernetes. This is](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=101.65) [a Configuration definition, and I've given it the name appconfig. The key](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=107.2) [part here is in the spec section where we've set up tracing with a sample](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=113.47) [rate of 1. This is the probability that any given tracing span will be](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=118.98) [sampled. One is the maximum value, so this means capture all traces,](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=124.35) [which is good for development, but might result in too](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=130.24) [large a volume of tracing data in production, so you might](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=133.2) [set this 0.01 sample only 1%.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=136.96) [And of course, we could set it to 0 to completely disabled tracing.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=141.74) [We've also configured it to use the Zipkin protocol and told it where the](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=146.64) [endpoint is that it needs to send the tracing information to. In this case, it's a Zipkin container running on the kubernetes cluster.](https://app.pluralsight.com/course-player?clipId=34650536-ea4d-41df-bd68-fa3361a7c28e&startTime=151.08)

### [W3C Trace Context](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea)

[As well as sampling tracing information, Dapr is able to](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=1.04) [propagate W3C tracing context information.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=4.88) [What does that mean?](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=9.44) [Well, let's consider what happens when we load the GloboTicket home page.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=10.84) [First, there's a request to the frontend microservice. That in turn makes a](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=15.74) [request to the catalog microservice to fetch the list of events, and the](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=20.61) [catalog microservice uses the Dapr sidecar to fetch a connection string](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=25.8) [secret and then talks to a database. And the frontend microservice also uses](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=30.66) [the Dapr state store building block to find out if there are any items in](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=36.15) [your shopping basket.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=40.35) [As you can see, even in the context of simply loading a single web page,](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=42.54) [several different services and external components are involved. And that](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=47.4) [picture could get even more complicated in the future as additional features](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=52.55) [are added to GloboTicket. And this means that if something goes wrong during](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=56.98) [the loading of that page,](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=61.62) [we need to piece together all of the logs from the different](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=63.25) [components to understand what happened. And what the W3C trace](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=66.31) [context does is it allows us to pass around a unique identifier that](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=71.53) [ties together all of these operations.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=76.57) [Dapr is able to both generate trace context and to propagate already](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=80.04) [existing trace context using the industry standard W3C trace context](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=85.68) [specification. And this means that any other services you interact with](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=91.79) [that can interoperate with this standard can also tie their logs into](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=96.65) [the same trace context.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=100.96) [It also means that in your own logging code,](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=103.54) [you can access the trace context and ensure that it's](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=105.96) [recorded alongside each of your log messages.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=109.34) [And the way this capability works is with HTTP headers.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=113.34) [The traceparent header contains a unique identifier for the incoming request](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=117.74) [and can be used to tie information from all systems together.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=123.14) [And there's also a tracestate header that can propagate additional](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=127.74) [state information between services. And if you're interested in](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=131.63) [learning more about this capability,](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=136.06) [there's an article here on the Dapr documentation website](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=138.28) [that includes code samples in various languages to show you](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=141.83) [how you can access these headers, as well as how you can create your own trace context.](https://app.pluralsight.com/course-player?clipId=94843c89-902b-4387-bbf0-8958f85499ea&startTime=145.72)

### [Health Checks](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208)

[In a microservices architecture,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=1.04) [it's generally considered best practice for your](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=3.28) [microservice to expose a health check endpoint,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=5.91) [and this can be used to report whether your](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=10.24) [microservice is running correctly or not.](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=12.05) [For example, has it started up,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=14.61) [and is it able to successfully communicate with its downstream dependencies?](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=16.9) [And one of the key benefits of having health check endpoints is that container](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=22.74) [orchestrators like Kubernetes can use them to know whether a newly started](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=27.4) [microservice is ready to start receiving traffic,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=32.9) [and they can also use them to detect when your microservice](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=37.14) [has a problem and needs to be restarted.](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=40.13) [Of course,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=43.94) [when you're running Dapr on Kubernetes, for every](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=44.48) [microservice, there are two containers,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=47.68) [your microservice itself and the Dapr sidecar, and they both run inside a](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=50.84) [Kubernetes Pod. It's your responsibility to create the health check endpoint on](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=56.7) [your microservice if you'd like Kubernetes to monitor it, and the Dapr sidecar](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=62.81) [comes with a built‑in health check endpoint.](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=68.43) [This can be accessed at v1.0/healthz,](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=71.84) [and this allows Kubernetes to determine the overall health of the Pod by](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=77.04) [calling the health check endpoints for the sidecar and for your](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=81.35) [microservice. And Kubernetes will only consider a Pod to be healthy if](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=85.19) [all of the health check endpoints for the containers running inside it report that they're healthy.](https://app.pluralsight.com/course-player?clipId=b534578c-6104-4cdb-8772-c71ed8a0d208&startTime=90.81)

### [Dapr Metrics](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265)

[Another observability benefit of Dapr is that it](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=1.04) [exposes a Prometheus metrics endpoint.](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=4.38) [This allows you to gain better insight into how Dapr is behaving and create](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=8.14) [alerts that warn you if there's a problem in production.](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=12.79) [This endpoint is configured by default and has a default port number](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=16.24) [of 9090. And if we look here on GitHub, we can see a list of the](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=20.18) [metrics that Dapr is able to track for us.](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=24.93) [These include counting various failure conditions,](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=28.54) [such as component initialization problems or MTLS authentication failures.](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=31.53) [And it also includes several performance‑related metrics,](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=38.14) [such as latencies for HTTP requests, and so this can be a helpful way of detecting performance issues in production.](https://app.pluralsight.com/course-player?clipId=97f2c622-9162-4adb-957a-9857947fb265&startTime=41.44)

### [Dapr Sidecar Logging](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b)

[Another way that Dapr helps us with observability is that](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=1.04) [each sidecar emits logging messages.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=4.69) [You've already seen these if you've run Dapr in self‑hosted mode.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=7.25) [For the most part, you can safely ignore these messages.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=11.58) [They're simply providing low‑level information about](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=15.39) [what the Dapr sidecar is doing.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=18.55) [But one situation in which I find the Dapr sidecar log messages very helpful](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=21.14) [is when you have problems with component configuration.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=25.61) [Maybe you made a mistake in the YAML configuration for a component,](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=27.84) [or perhaps the resource you're trying to connect to is unavailable.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=32.74) [And these are the types of error that the Dapr](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=35.51) [sidecar is able to report for you.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=38.6) [So if you're encountering issues, whether running in self‑hosted mode](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=41.44) [or in Kubernetes, make sure that you also examine the Dapr logs, as](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=45.06) [well as the logs from your own application.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=49.74) [The logs from Dapr can be configured to be logged in JSON](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=52.54) [format, and you can direct them to log collectors such as](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=55.9) [Fluentd or Elasticsearch. And once again,](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=59.75) [the Dapr documentation is the place to go to find out how to configure this for the systems that you want to integrate with.](https://app.pluralsight.com/course-player?clipId=37125cdf-dd20-4380-b17c-4b795df6600b&startTime=63.26)

### [Demo: Dapr Dashboard](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16)

[Another great feature that we can use to help us with](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=1.04) [observability is the Dapr dashboard.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=4.02) [This is a web‑based user interface for Dapr,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=7.44) [allowing you to see useful information about the Dapr components,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=10.34) [configuration, and applications that you're running.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=14.49) [The Dapr dashboards, like the rest of Dapr, is open source, and its repository](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=18.54) [can be found here on GitHub, which is a good place to start learning more about](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=23.67) [its capabilities and make any feature requests.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=28.12) [It's very easy to get started with, and you can see here on the GitHub](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=32.04) [page the instructions for launching the dashboard.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=36.17) [We've already got the Dapr CLI installed, so all we need to](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=39.74) [do is run the dapr dashboard command.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=43.06) [If you want to run the dashboard against Kubernetes, you also need to](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=46.34) [pass the ‑k switch and, optionally, the ‑n switch to specify the](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=50.01) [Kubernetes namespace you installed Dapr in.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=56.1) [Let's take a look at the Dapr dashboard on a Kubernetes installation of](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=59.84) [GloboTicket. I'll launch it with dapr dashboard ‑k, and it shows me the URL](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=64.18) [it's running on, which is localhost 8080. And, of course, what's happening](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=70.49) [here is that I'm essentially port forwarding from my local machine onto the](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=75.62) [Kubernetes cluster. Here we can see it's showing me the version of the Dapr](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=80.04) [control plane that I have installed and that my Dapr installation is](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=85.06) [healthy.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=89.25) [It's also showing me the Dapr applications that I have](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=90.64) [running. These are our three microservices,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=93.75) [the catalog, frontend, and ordering microservices.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=97.18) [I can navigate into one of these Dapr applications, such as the catalog](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=101.24) [microservice, and see some useful information about it, such as its](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=105.47) [application ID and Dapr HTTP port number. And you can see that there are](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=109.93) [several other views, such as Metadata, which is showing us the](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=115.08) [annotations on this application.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=118.83) [The Configuration tab shows us the full Kubernetes Pod configuration, which](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=121.54) [includes lots of useful information, such as all of the environment](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=126.55) [variables that have been configured for our containers. But perhaps most](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=130.34) [useful of all for troubleshooting is the Logs tab. Here we can access the](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=134.66) [logs from both the sidecar, which is this container called daprd, and our](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=139.51) [microservice itself, which we can access by changing the container to](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=145.53) [catalog. And we can even get it to highlight a keyword that we're looking](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=149.94) [for in the logs. So I can look for the word scheduled, for example. Let's](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=154.3) [go back to the home page and follow this More Information link.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=160.15) [This takes us to a view that shows us the Dapr services that are](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=164.94) [running in our cluster and allows us to check what version they are](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=168.3) [and ensure that they're all healthy.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=172.81) [Another view we can access is the Dapr Components view. This shows us all](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=175.34) [of the components that we've configured. And you can see here that we've](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=180.4) [got an Azure Service Bus pub/sub component, a cron binding for our](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=184.18) [scheduled task, an SMTP binding to send email, and an Azure Blob storage](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=188.96) [binding for our state store. And I can drill into one of these bindings to](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=194.77) [see additional information, including the YAML component configuration.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=200.04) [Finally,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=205.84) [there's a view of Dapr Configurations. In here, we can see not only the default](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=206.45) [daprsystem configuration, but also the appconfig configuration file that I](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=212.2) [showed you earlier. Here we can see that all three of our microservices have](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=217.59) [been configured to use this appconfig configuration. And if we look at the YAML](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=222.63) [for the component configuration,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=228.74) [we can see that this includes setting up the Zipkin tracing,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=230.71) [which is what we're going to be using in our next demo.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=234.55) [As you can see,](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=238.14) [the Dapr dashboard is a really powerful way of getting insight into the overall health of both Dapr and your own microservices.](https://app.pluralsight.com/course-player?clipId=2c83ea99-4d2f-4c1c-8b8b-41da9e36fc16&startTime=238.97)

### [Demo: Distributed Tracing with Zipkin](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df)

[In this demo, we'll see how the distributed tracing in Dapr works by](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=1.24) [configuring tracing and viewing some of those traces in Zipkin.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=6.53) [Let's start off by looking at how I've configured tracing with Dapr. Here,](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=11.84) [we're looking at my appconfig file, which we've discussed already. And I](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=15.91) [explained that the samplingRate of 1 means that we're going to capture all](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=20.77) [traces, and we're using the Zipkin protocol and sending these traces to the](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=25.21) [zipkin container that's running on our cluster. And here in the](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=30.62) [aks‑deploy.ps script, which has got all of the commands that I used to](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=35.13) [deploy GloboTicket into Azure, you can see here is the bit that creates our](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=40.02) [Zipkin container.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=45.74) [We're using kubectl create deployment to create a Kubernetes deployment](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=47.34) [running the openzipkin/zipkin container image, and then we're using kubectl](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=52.32) [expose deployment to make that deployment available locally in the cluster on](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=58.75) [port 9411. And then this call to kubectl apply is deploying the configuration](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=64.56) [file that we just looked at.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=71.33) [I've also included a helpful command here. In order to access the Zipkin](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=73.84) [instance that's running on our Kubernetes cluster, I can use kubectl](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=78.25) [port‑forward to make the service we exposed available on localhost port](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=82.66) [9412. And by the way, the reason I didn't choose 9411 is because I've](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=88.2) [already got another local Zipkin instance running on my machine on port](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=94.54) [9411, which was created when I installed Dapr in self‑hosted mode. So let me](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=99.25) [run this command to set up the port forwarding to Kubernetes. And now I'll](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=105.84) [visit the Zipkin instance in my Kubernetes cluster by visiting localhost](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=110.85) [port 9412.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=116.26) [And here we can see the Zipkin user interface that lets us](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=118.64) [search the traces that have been captured.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=122.25) [I'll click RUN QUERY, and you can see that it shows me a list of](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=125.34) [recent traces. And we can also see the duration of those traces,](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=129.7) [which is really helpful to quickly identify where there might be](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=134.34) [some performance issues.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=137.62) [Let's take a look inside this top one by clicking SHOW. And](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=139.54) [here, it gives us a breakdown of all of the individual](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=143.59) [requests involved in that trace.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=146.77) [We can see that we access the state store to fetch a shopping basket, and](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=150.14) [we published an event, which the ordering microservice picked up, and we](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=154.9) [called invokebinding to send an email, and it breaks it down to show us the](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=159.6) [time taken by each step in the trace.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=164.34) [We can also look in more detail at individual requests in the trace such as](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=167.94) [the call here to fetch the basket from the state store.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=172.88) [Let's also take a look in this Dependencies view.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=177.04) [If I perform a search for some recent traces, we can see that it draws a graph](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=180.64) [showing the Dapr applications, which are our three microservices. And then these](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=185.62) [animated dots moving across the lines show us the communication between those](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=190.81) [services. As we can see, we're making a lot of requests from the front end to](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=195.93) [the catalog microservice and occasionally one to the ordering microservice, which is what we'd expect.](https://app.pluralsight.com/course-player?clipId=cf59745e-6869-460b-8fb6-863ce14ea2df&startTime=201.29)

### [Additional Dapr Capabilities](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53)

[We're close to the end of this course now,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=1.04) [and we've explored several of the most important Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=3.18) [The ones we've used in our GloboTicket demo application are state management,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=7.84) [service invocation, pub/sub messaging, secrets,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=12.66) [input and output bindings, and observability.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=16.94) [And that's a pretty powerful set of capabilities that will be](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=21.24) [useful to anyone who's building a microservices application. But](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=24.38) [it's not all that Dapr has to offer.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=29.17) [In particular,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=31.44) [I want to highlight three capabilities that we've not used in this course,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=32.51) [but that you might want to explore and consider whether](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=36.94) [they're a good fit for your application.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=39.56) [The first is actor support.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=42.34) [Dapr comes with a very powerful building block that supports](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=44.54) [using the virtual actor architectural pattern.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=48.02) [With the actor pattern, you write code in self‑contained units called actors,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=51.74) [which receive and process messages one at a time.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=57.14) [And actors can send messages to other actors, as well as create new actors.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=60.94) [And it's possible to support a very large number of actors simultaneously.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=66.04) [And although perhaps it's not a very widely used pattern,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=71.24) [there are some types of problems that it's ideal for.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=74.63) [And the great thing with Dapr is that you have this](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=77.94) [building block available to you,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=80.07) [and so if at any time you identify a problem as being](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=82.1) [a good fit for the actor pattern, you can easily make use of it.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=85.17) [The second is the configuration building block.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=90.14) [The reason we didn't focus on this in this course is at the time of](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=93.54) [recording it was still in alpha and subject to change,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=96.83) [but it helps with another very common problem with microservices,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=100.94) [which is having somewhere to store and manage all of the](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=104.59) [configuration settings for your microservices and getting](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=107.83) [notified when those settings are modified.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=111.17) [Finally,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=114.64) [Dapr offers some middleware components that can run as a](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=115.09) [pipeline inside your Dapr sidecar,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=118.53) [transforming requests and responses as they go through.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=121.54) [The tracing functionality that we looked at earlier is an example](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=126.39) [of some middleware that's enabled by default.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=130.04) [One particularly noteworthy middleware component is the OAuth 2 middleware,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=133.74) [which enables the OAuth 2 authorization code flow on a web API.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=138.94) [Again,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=144.04) [this has the potential to take some complexity out of your](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=144.87) [microservice code and move it into the Dapr sidecar.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=148.26) [And there's also a rate‑limiting middleware component,](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=152.14) [which is another good example of the kind of added value that middleware can offer.](https://app.pluralsight.com/course-player?clipId=458f9da7-5746-47d2-8b6b-2b3e355e5e53&startTime=154.9)

### [Dapr Community Resources](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d)

[Before we wrap up this module and the course,](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=1.04) [I want to point you to some of the key community resources that](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=3.88) [you'll want to be aware of as you continue your journey of](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=7.7) [evaluating Dapr and incorporating it into your own microservices.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=11.16) [And some of these we've seen already.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=15.08) [First of all,](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=18.54) [the official Dapr documentation site is your best starting point](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=19.31) [for learning about building blocks and component configurations.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=23.61) [And the Dapr GitHub repository is a great place to report bugs and](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=27.08) [make feature requests, as well as learn about the roadmap for](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=32.99) [future versions of Dapr.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=36.7) [Another great thing about Dapr is that approximately every two weeks they](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=38.84) [host a Dapr community call that you can watch on YouTube.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=42.97) [This gives you updates on the features that they're currently working on, and](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=47.74) [there's an opportunity for you to ask your own questions.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=51.64) [Dapr also have a community on Discord which is a very active community](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=55.34) [with discussion channels for all of the building blocks, as well as](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=59.92) [individual programming language support.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=63.35) [It's a great place to ask for help with issues you're facing, and I](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=66.14) [really like the show and tell channel where people share the cool](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=69.77) [things that they've built with Dapr.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=73.34) [It's a great way to learn about things that can be](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=75.24) [achieved using Dapr. And finally,](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=77.37) [Microsoft have made a free Dapr for .NET Developers e‑book available,](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=80.64) [which is an excellent resource that works you through](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=85.74) [building a traffic control sample application and uses many of the Dapr building blocks.](https://app.pluralsight.com/course-player?clipId=ace5a025-293d-4deb-9416-24880040f95d&startTime=88.05)

### [Module Summary](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2)

[In this module,](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=1.24) [we learned about several of the powerful features](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=2.05) [that Dapr offers for observability.](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=4.98) [We saw how tracing can be captured using the Zipkin protocol](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=7.28) [and that Dapr supports W3C tracing headers.](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=12.34) [We also learned about additional observability metrics,](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=16.34) [sidecar logging, health check endpoints,](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=20.04) [and the Dapr dashboard.](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=22.83) [And we also explored some of the additional Dapr capabilities that you](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=25.34) [might wish to explore after watching this course, such as the actors and](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=28.69) [configuration building blocks, as well as middleware for things like OAuth](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=33.37) [and rate limiting. And I pointed you in the direction of some additional](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=37.9) [learning resources that will help you to keep up to date and get help as](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=42.42) [you adopt Dapr,](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=47.01) [including the community call videos and the Discord channel. And this is also a](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=48.33) [good time to remind you that the sample code for this course is available here](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=53.61) [on GitHub The main branch contains the complete GloboTicket application that](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=58.13) [uses Dapr and includes scripts that help you run it locally or in Azure. And](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=63.49) [although we didn't use it in this course, there are also some Docker Compose](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=69.31) [files, which demonstrate how you could run it locally using Docker Compose. And](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=73.17) [if we look at the branches, there's a before branch, which has the starting](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=78.61) [point of the GloboTicket without Dapr, and I've also got some branches in there](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=82.89) [for specific versions of Dapr, and I hope to be able to create more of these as](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=87.46) [new versions of Dapr are released, although I found so far that the changes](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=92.38) [required to move between Dapr versions are very minimal.](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=97.38) [Thanks for watching this course, I hope you found it helpful, and](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=101.94) [feel free to provide feedback on the Pluralsight discussion board or on the GloboTicket Dapr GitHub repository.](https://app.pluralsight.com/course-player?clipId=13afebff-df4e-4957-890f-91bf4e5c7be2&startTime=105.31)