.NET Aspire: Not just for cloud native distributed systems

Slides up at scottsauber.com

scottsauber

Audience

- .NET Developers building APIs or web apps
- Cloud Developers curious what this Aspire thing is

Agenda

- What is .NET Aspire
- Why?
- What makes up .NET Aspire
- Live Demos

Goals

- Expose you to what .NET Aspire is
- Understand that it’s not just for cloud native microservices apps

Who am I?

- Director of Engineering at Lean TECHniques
- Microsoft MVP
- Co-organizer of Iowa .NET User Group
- Dometrain Author
- Blog at scottsauber.com

“.NET Aspire is an opinionated, cloud ready stack for building observable, production ready, distributed applications.”


DRUGS

What is .NET Aspire

- Lots of things
- Local Dashboard for viewing Open Telemetry
- Provides Orchestration
  - Fancy word for running multiple things – APIs, Frontend, Backend, Local database, etc.
- Provides Service Discovery
  - Fancy words for “how do I call you” – URL, connection string, etc.
- Provides Service Defaults
  - Fancy words for “how do I configure you” – HttpClient, Telemetry, etc.
- Provides Integrations
  - Fancy word for NuGet packages to configure external systems like Azure, AWS, Redis, Databases, etc
- I don’t love the Aspire name

Why .NET Aspire

- Every app wants better observability
- Companies usually have opinions on how things are configured
- Orchestrating multiple dependencies and microservices is hard
- Aspire… aspires… to solve all of these

“.NET Aspire is an opinionated, cloud ready stack for building observable, production ready, distributed applications.”

“.NET Aspire is an opinionated, stack for building observable, production ready applications.”

“You should be using .NET Aspire for any kind of applications you’re building.”

What is the Dashboard?

- View all your running apps in one spot
- View logs
- View spans and traces
- Shippable as a Docker image
- Based on OpenTelemetry

What is OpenTelemetry?

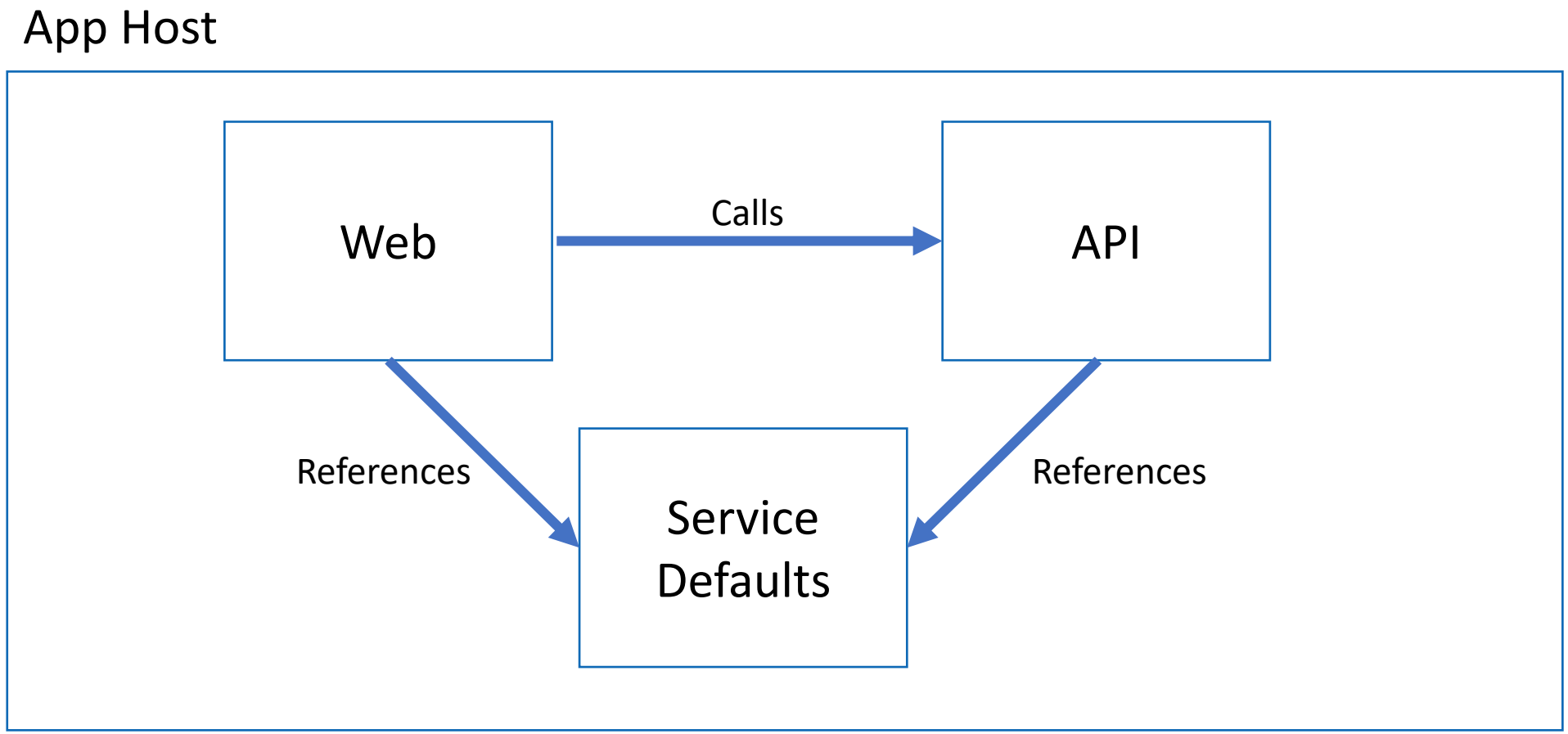
- Open standard all the observability providers are converging on
- Avoid vendor lock-in
- Azure Monitor supports OpenTelemetry
- Been baked into .NET
- Aspire Dashboard speaks OpenTelemetry
- Aspire wires up OpenTelemetry niceities for you

Before we demo…

Sample Aspire App Structure

- 4 csproj’s
- .Web = Frontend
- .ApiService = API the Frontend calls
- .ServiceDefaults = Sets some defaults (OTEL, Health Checks, HttpClient defaults, etc) – shared by .Web and .ApiService
- .AppHost = Orchestrator booting up all the things

Sample Aspire App Structure



Demo

Dashboard FAQ Answered

- This is a local only tool
- Yes it’s a Docker image, you could host it yourself
- But storage…
- You should probably use AppInsights or Datadog or w/e for non-local

Feature Check-In

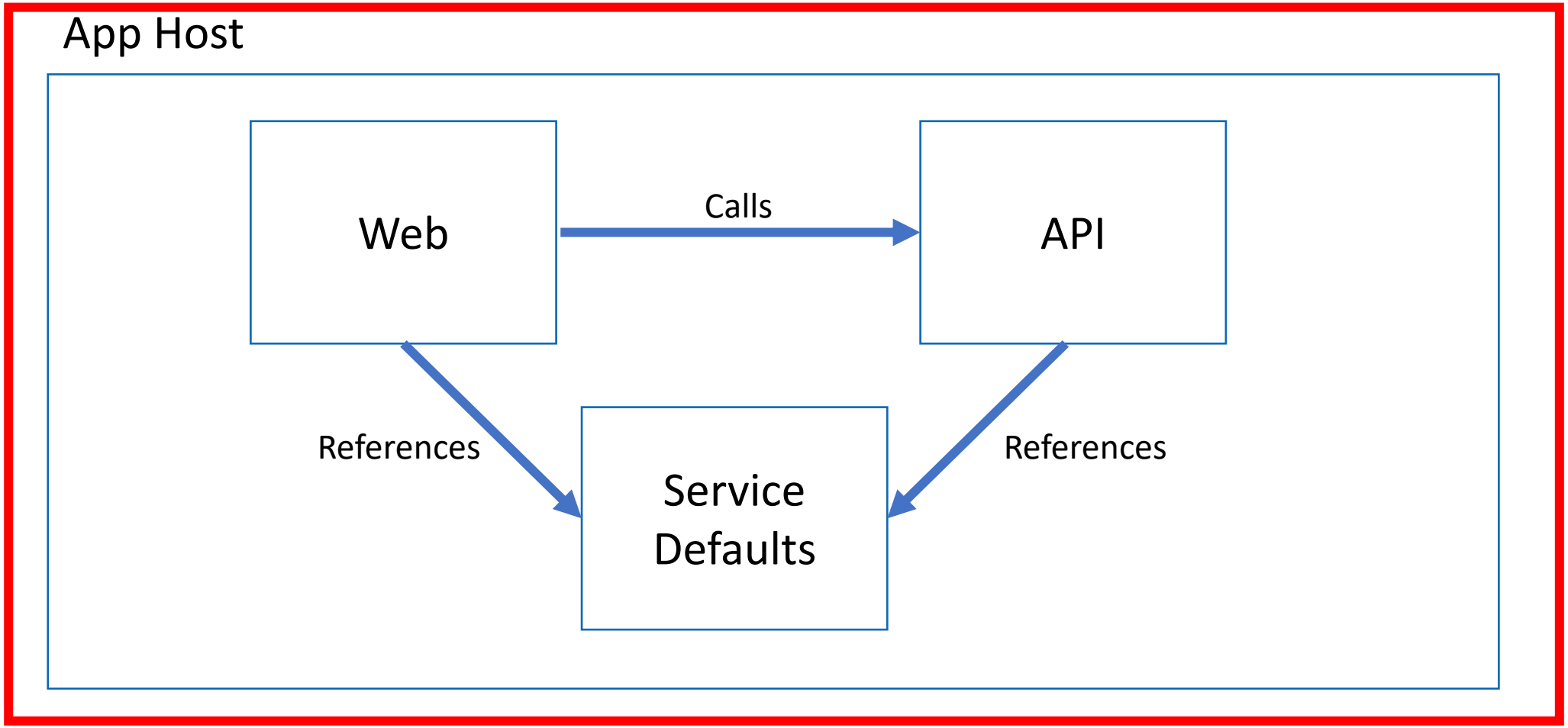
Feature	Useful for cloud-native apps?	Useful for non-cloud-native apps?
Dashboard	✅	✅

Questions about the Dashboard?

What is the AppHost?

- AppHost is the orchestrator
- Makes sure it boots up all your dependencies
- Could be APIs, Frontends, Databases, etc
- Almost every app has at least one dependency – not just cloud native distributed apps

Sample Aspire App Structure



Demo

Feature Check-In

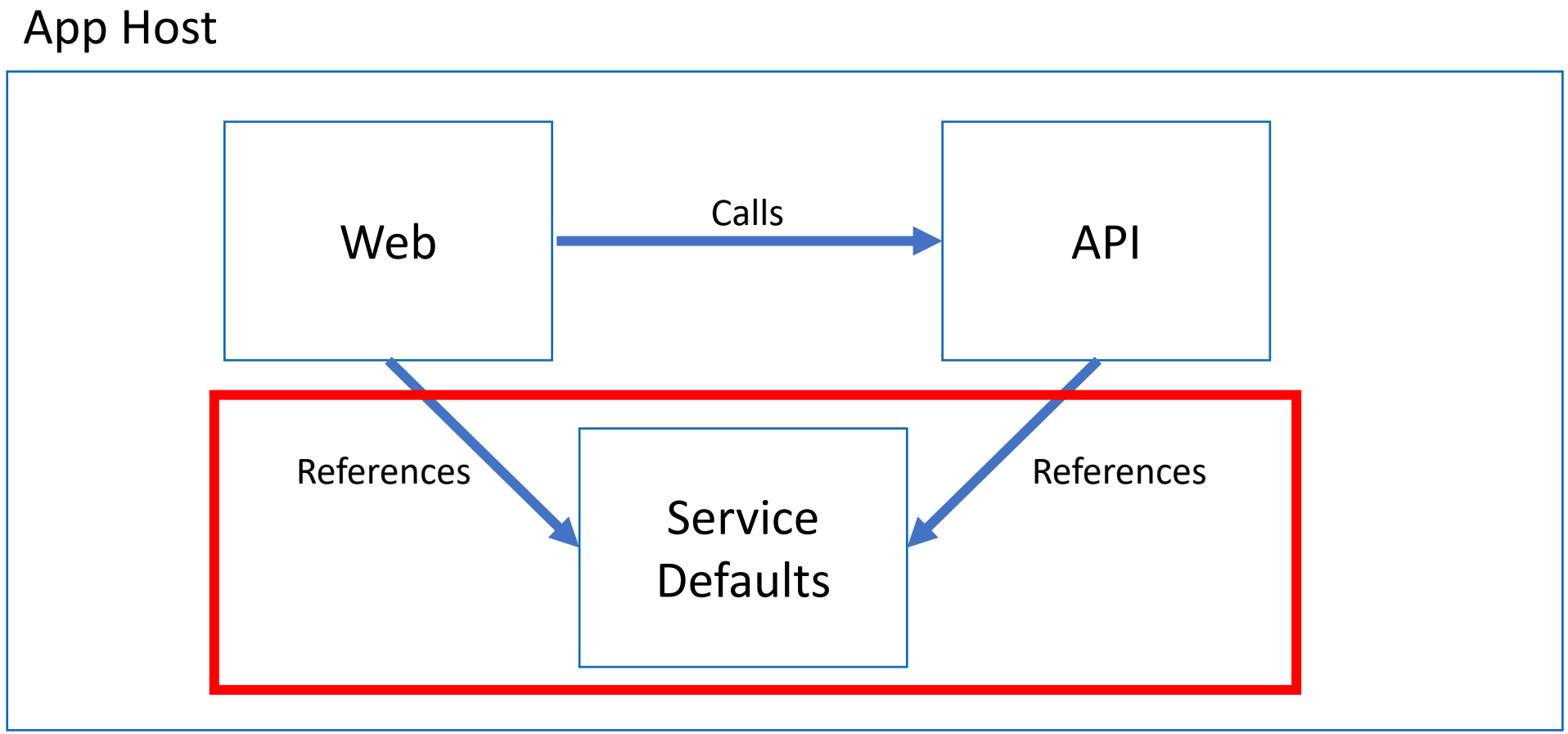
Feature	Useful for cloud-native apps?	Useful for non-cloud-native apps?
Dashboard	✅	✅
AppHost	✅	✅

Questions about the AppHost?

What are the ServiceDefaults?

- How you configure your application
- OpenTelemetry, HttpClient resiliency, Health Checks, Service Discovery, etc.
- Could publish this as a NuGet package internally so all apps use the same ServiceDefaults

Sample Aspire App Structure



Demo

Feature Check-In

Feature	Useful for cloud-native apps?	Useful for non-cloud-native apps?
Dashboard	✅	✅
AppHost	✅	✅
ServiceDefaults	✅	✅

Questions about ServiceDefaults?

What are Integrations?

- NuGet packages to simplify configuration to common services
- Postgres, Redis, Azure Key Vault, Azure Service Bus, etc.
- Publish data to OTEL
- All vetted integrations live under the “Aspire.” prefix ie Aspire.Azure.Storage.Blobs or Aspire.MongoDB.Driver
- https://learn.microsoft.com/en-us/dotnet/aspire/fundamentals/integrations-overview#available-integrations

```
builder.AddAzureServiceBusClient(connectionName: "servicebus");
```

- Registers ServiceBusClient as singleton with reusing Azure tokens
- Automatically sets up Logging, Tracing, Telemetry
- Applies any custom configurations

Demo

Feature Check-In

Feature	Useful for cloud-native apps?	Useful for non-cloud-native apps?
Dashboard	✅	✅
AppHost	✅	✅
ServiceDefaults	✅	✅
Integrations	✅	✅

Questions about Integrations?

Current State

- Everything I’ve shown is fully usable today

Future State

- Deploy using Aspire (can today technically with azd…)
- Infrastructure as Code using Aspire

Takeaways

- Understand what .NET Aspire is
- Why .NET Aspire is useful
- Realize it’s useful for any app – small or large… cloud native or not…

Resources

- The Microsoft Docs on Aspire
- 9/18 – Azure Developers - .NET Aspire Day 2024
- This slide deck

Questions?

Slides up at scottsauber.com

scottsauber

Thanks!

Slides up at scottsauber.com

scottsauber