Logging, Metrics, Distributed Tracing and OpenTelemetry

Christian Nagel

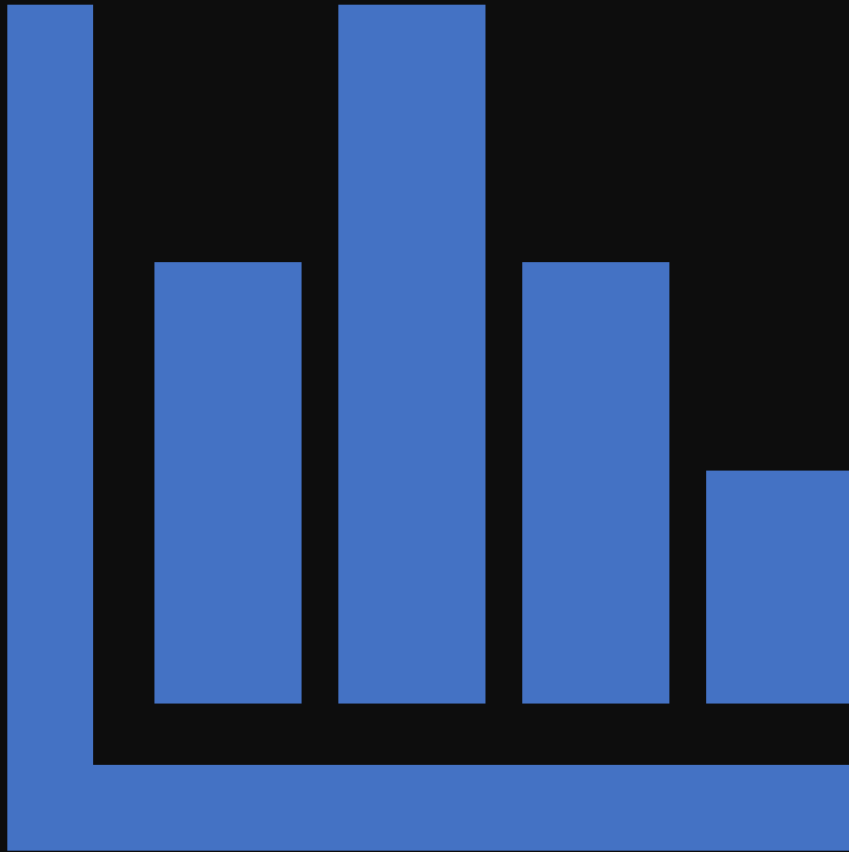https://www.cninnovation.com

Christian Nagel

- Training
- Coaching
- Consulting
- Development

- Microsoft MVP
- www.cninnovation.com
- csharp.christiannagel.com
- @christiannagel

Topics

- Implementing
  - Logging
  - Metrics
  - Distributed Tracing
- Displaying
  - .NET Aspire Dashboard
  - Prometheus and Grafana
  - Application Insights

Logging

Logging use cases

MONITOR APPLICATION BEHAVIOR

DIAGNOSE ISSUES

3 important parts

Providers	Sources	Levels
• Where is the message written to	• Who is logging	• How important is the message

Log Providers

- Console
  - Write messages to the console
- Debug
  - System.Diagnostics.Debug
- EventSource
  - Name: Microsoft-Extensions-Logging
- EventLog
  - Only on Windows

Hierarchical Log Sources

- Microsoft.EntityFrameworkCore...
- Microsoft.AspNetCore.Server.Kestrel
- Microsoft.AspNetCore.Mvc.Infrastructure.ControllerActionInvoker
- ...

Log Levels

- Trace
- Debug
- Information
- Warning
- Error
- Critical

Filtering

Filter logging based on providers, sources, and log levels

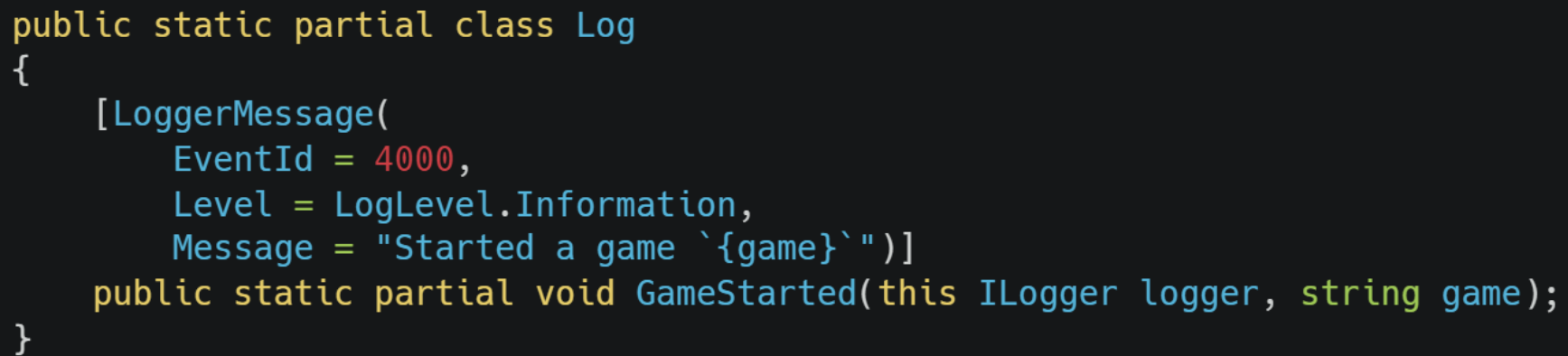# Creating Logs
ILogger and *ILoggerFactory*

- Inject *ILogger<T>* to specify the log category
- Inject *ILoggerFactory* to specify the log category as string

Structured / Semantic Logging

- Don't use interpolated strings with logging

- Log methods use templates

- Placeholder names can be used for filtering

Source generated logging

```
public static partial class Log
{
    [LoggerMessage(
        EventId = 4000,
        Level = LogLevel.Information,
        Message = "Started a game `{game}`")]
    public static partial void GameStarted(this ILogger logger, string game);
}
```

Check event log level for complex parameters

```csharp
[LoggerMessage(
    EventId = 4005,
    Level = LogLevel.Information,
    Message = "Returned {NumberGames} games using {Query}")]
private static partial void QueryGames(this ILogger logger,
    int numberGames, string query);

public static void QueryGames(this ILogger logger,
    IEnumerable<Game> games, string query)
{
    if (logger.IsEnabled(LogLevel.Information))
    {
        QueryGames(logger, games.Count(), query);
    }
}
```

Metrics

Metrics use cases

- Monitor system behavior

- Monitor solution behavior

- Trigger alerts

- Performance monitoring

Built-in .NET counters

- NET Runtime
- DI Container
- System.Net
- ASP.NET Core
- EF Core
- …

- .NET Tool
- dotnet counters

Writing custom counts

- Meter
  - Track instruments
- Counter
  - Requests per second or cumulative total
- UpDownCounter
  - Report positive and negative numbers
- Gauge
  - Non-additive values (e.g. room temperature)
- Histogram
  - Arbitrary values likely to be statistic meaningful
- Observable[Counter|UpDownCounter|Gauge]
  - Delegate invoked to retrieve value

.NET 8 Metrics Updates

- AddMetrics – register with DI container
- IMeterFactory
  - create multiple meters
- MetricCollector
  - Record metric measurements

Distributed Tracing

Distributed Tracing

- How do services communicate?
- Where did the error originate?
- Which service is a bottleneck?
- What services are dependent on others?

.NET Activitiy

- Id
  - Hierarchical structure
  - separated by '.' or '_'
- Tags
  - Information logged with activity
- Baggage
  - Logged with activity and passed to child
- Events
  - Time-stamped messages (e.g. "done now")

Distributed Tracing

- Trace information across service boundaries
- ActivitySource.StartActivity
  - Start an activity
  - Hiearchical ids (W3C TraceContextId)
  - ID passed with HTTP header
- Add Information
  - Tags (written to activity trace)
  - Baggage (written to activity trace and forwarded to child activities

.NET Tools

dotnet trace
- Performance analysis

dotnet counters
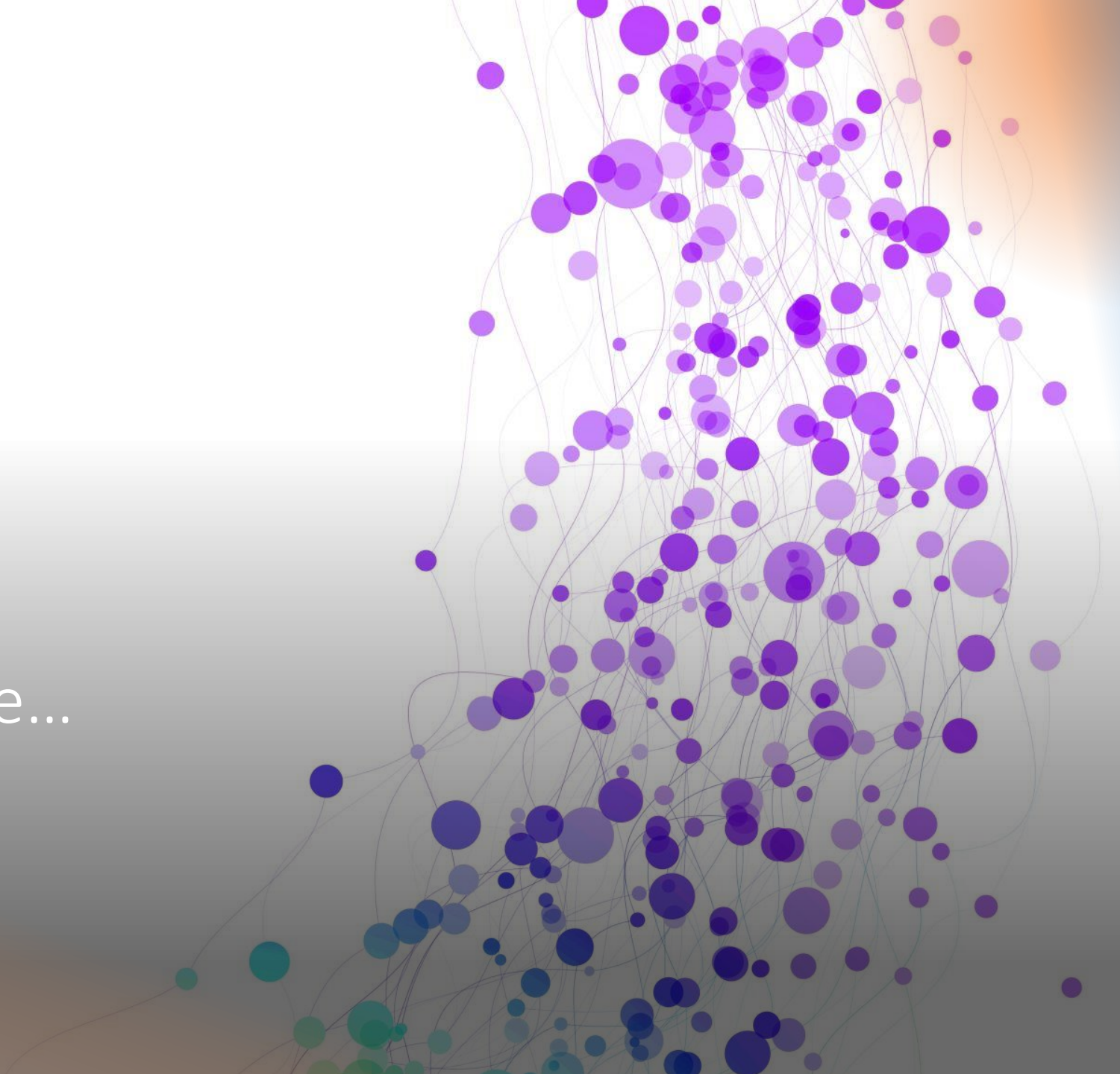- Investigate performance counters

dotnet monitor
- Processes, dumps traces, metrics, logs, operations…

OpenTelemetry
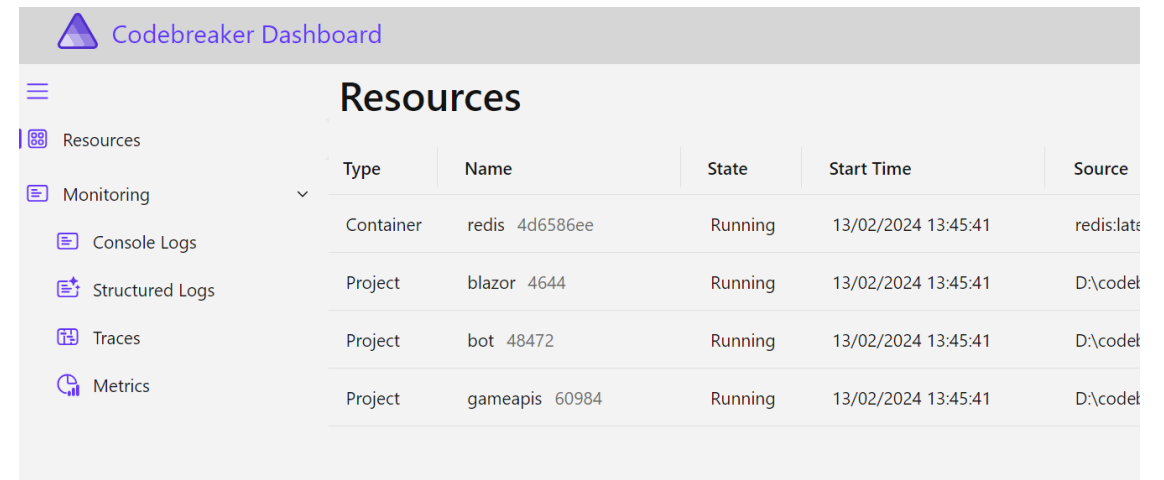
- https://opentelemetry.io
- Instrumentation
    - C++, .NET, Erlang/Elixir, Go, Java, JavaScript, PHP, Python, Ruby, Rust, Swift
- Collectors
    - Vendor-agnostic way to receive, process, export telemetry data

.NET Aspire and more…

.NET Aspire Dashboard

- Resources
- Monitoring
  - Console Logs
  - Structured Logs
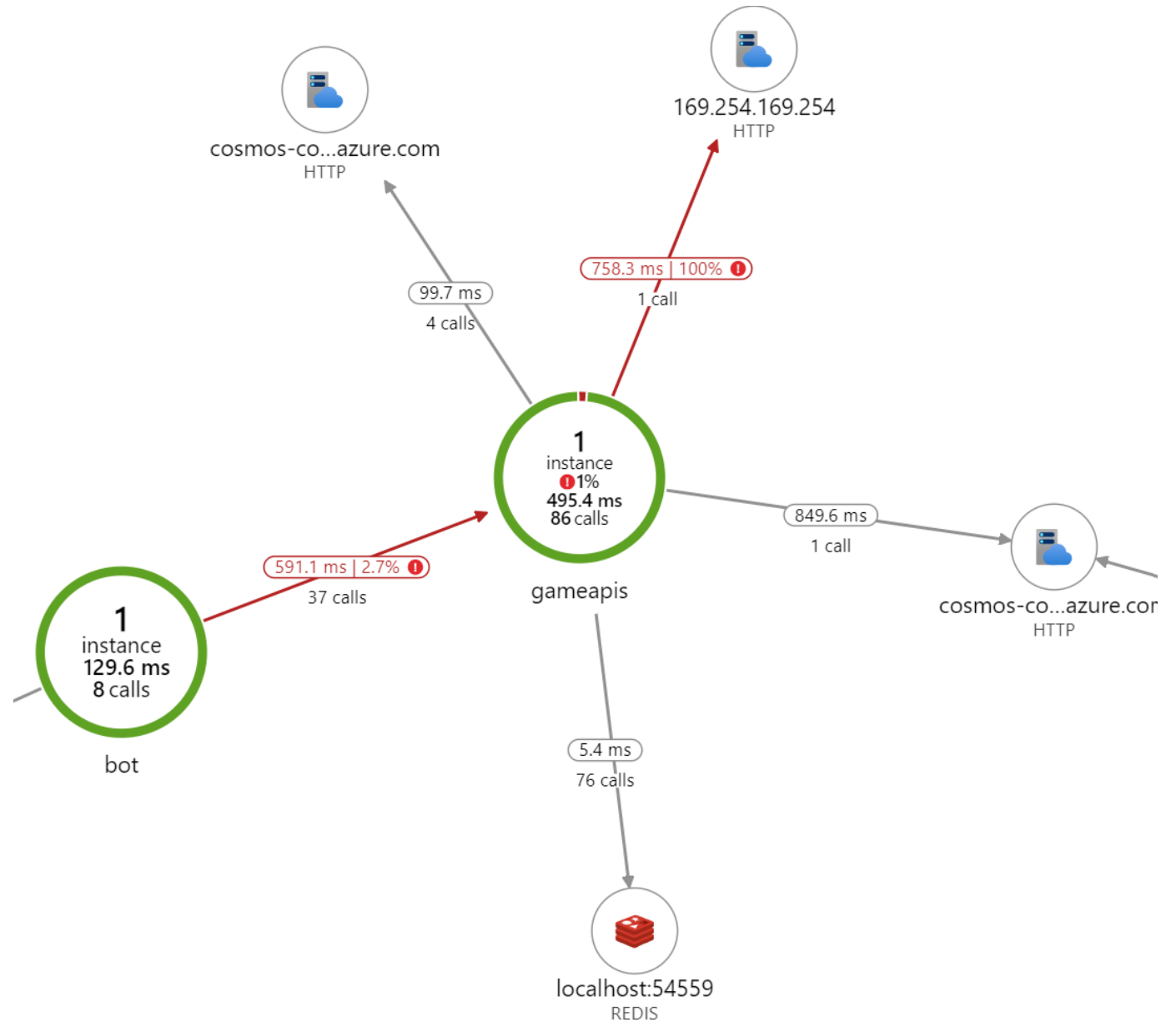 - Traces
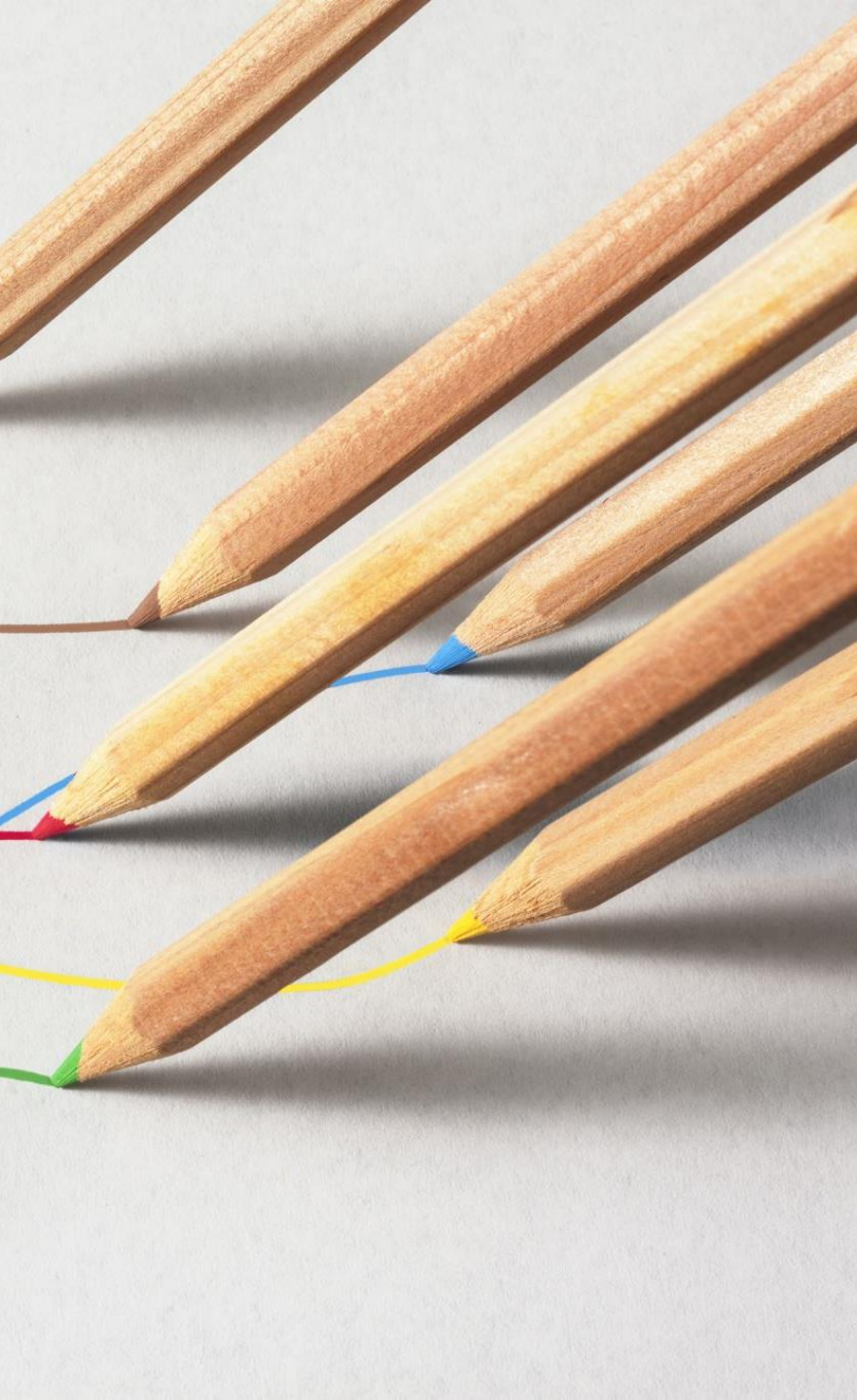 - Metrics

Prometheus and Grafana

- Using Docker images
- ASP.NET Core configured dashboards
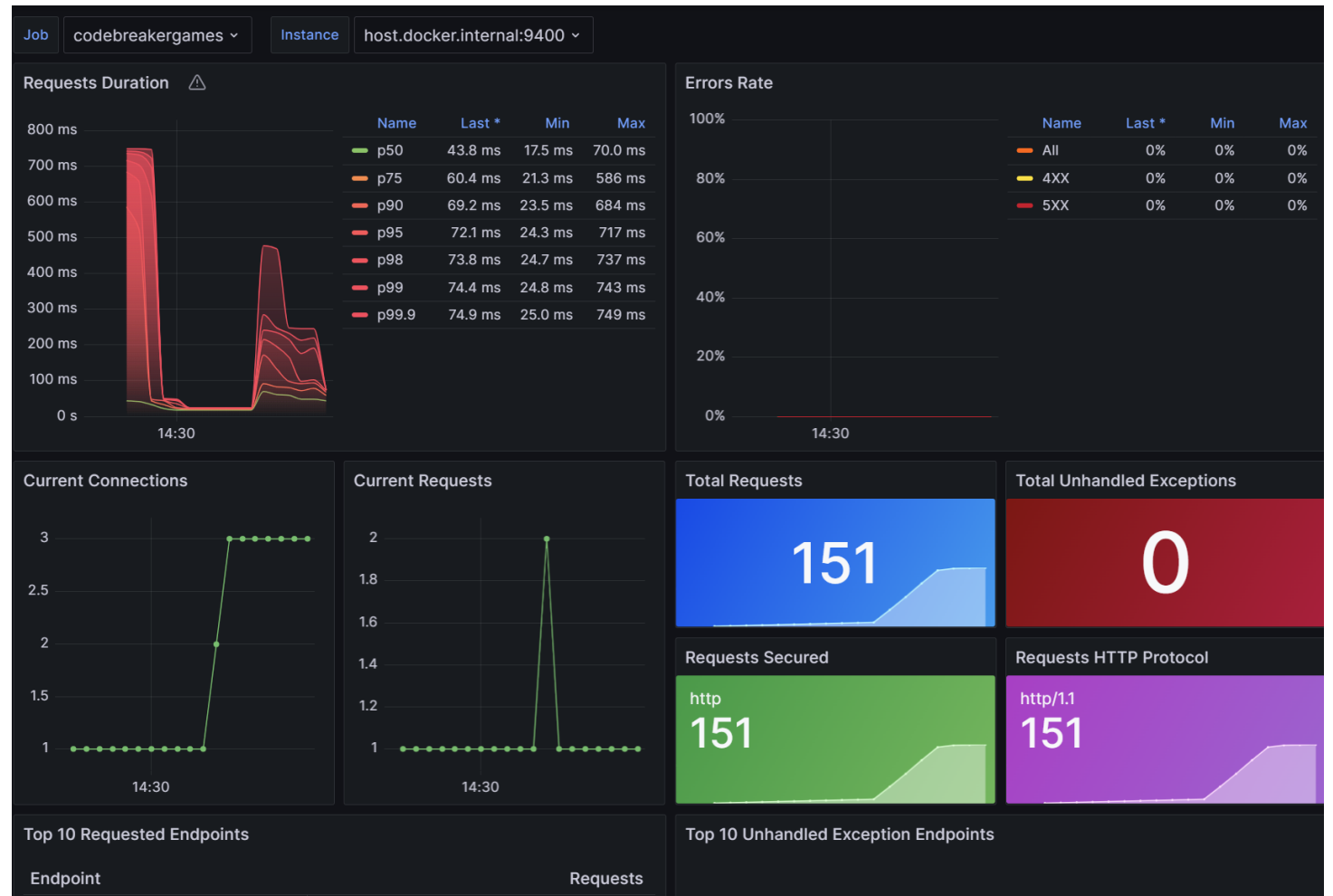
And Azure Log Analytics Workspace

Application Insights

Summary

- Logging
- Metrics
- Distributed Tracing
- .NET Aspire

Thank you for joining!

Questions?

- https://github.com/cnilearn/bastaspring2024
- https://csharp.christiannagel.com
- https://www.cninnovation.com