

Write a Global Logger as a Serilog Wrapper



Erik Dahl

PRINCIPAL ARCHITECT

@dahlsailrunner www.knowyourtoolset.com



Overview



What to log

- Types of entries
- Information to include

How to leverage Serilog

Demo: Establish global logger

Exception handling discussion

Demo: Data-access exception handlers

Types of Log Entries



Usage

Which features are most used?



Performance

Do we have any slow areas?



Error

Which errors occur the most?



Diagnostic

Ad-hoc troubleshooting



Information to Log

Where from?

Product
Layer
Location
Hostname

Who is in play?

User
Customer

What else?

Time spent
Parameters
Browser



Serilog

Structured data logging

Lots of Sinks (more than 50!)

- Text files
- SQL / NoSQL
- Splunk / Seq

Log by Level (6 provided levels)

Find out more

- <https://serilog.net>
- Modern Structured Logging With Serilog and Seq



Leveraging Serilog

Strategy: One Logger per “type”



..... UsageLogger



..... PerformanceLogger



..... ErrorLogger



..... DiagnosticLogger

Easily choose different sinks based on type

Easier consumption by developers

Doesn't obfuscate intent or meaning

Levels still available if needed



Demo



Establish a global logger

- Add Serilog
- Create a class with 4 loggers
- Add a class for info to be logged
- Add a performance tracker
- Verify with a console app



Exception Handling

Use global handlers

Avoid exception logic noise in
your application code

Examples coming

Add value

Derived values

Stored procedure info



Demo



Review logs from Stored Proc error

- ADO.Net
- Entity Framework
- Dapper

Add data access exception handler(s)

- ADO.Net wrapper
- Entity Framework dbInterceptor
- Dapper extension methods



Summary



Built a multi-purpose global logger

Enabled performance tracking

Enabled data access exception handling

Up Next:

Put it to use in various applications!

