

# Evolutionary history of the RabbitMQ .NET Client towards concurrency

# Goals

target

Familiarity with the RabbitMQ client

Understand how concurrency is hard with the existing client approach

Experience the process of evolving the client

# Process

Understand the synchronous path

Optimize the synchronous path

Introduce a redundant asynchronous path

# Recap

reminder

Find the **IO-bound** paths in your code

Get rid of **lock contentions**

If possible use a **redundant** code path  
that is **async all the way**

# Recap

reminder

Gradually extend IO-bound paths

Never block threads

Only custom offload where really necessary

# Recap

reminder

On .NET 5 consider **ValueTask** to benefit from pooling

**Leverage** existing battle tested **reliable** BCL components where you can

If you can, **contribute back to OSS** that you use in your company

Thanks

# Slides, Links...

[github.com/danielmarbach/AsyncRabbitMQ](https://github.com/danielmarbach/AsyncRabbitMQ)

[docs.particular.net/transport/rabbitmq](https://docs.particular.net/transport/rabbitmq)



# Q & A

[daniel.marbach@particular.net](mailto:daniel.marbach@particular.net)

[@danielmarbach](#)  
[particular.net/blog](http://particular.net/blog)  
[planetgeek.ch](http://planetgeek.ch)

# Thanks