

Behavior chaos strategy **CHEAT SHEET**



Basics

This proactive chaos **strategy injects a custom action** just before the actual operation call.

You can configure the behaviour of the strategy via the **ChaosBehaviorStrategyOptions** object.

Specify simple behavior

```
new ResiliencePipelineBuilder()
    .AddChaosBehavior(new ChaosBehaviorStrategyOptions
    {
        InjectionRate = 0.5,
        BehaviorGenerator = static args =>
            ChangeRequestHandlingAsync(
                RequestHandling.RejectWithClientError,
                args.Context.CancellationTokens)
    })
```

Specify behavior dynamically with switch expression

```
new ResiliencePipelineBuilder<HttpResponseMessage>()
    .AddChaosBehavior(new ChaosBehaviorStrategyOptions
    {
        BehaviorGenerator = static async args =>
        {
            var rnd = Random.Shared.NextDouble();
            RequestHandling handling = rnd switch
            {
                < 0.4 => RequestHandling.RejectWithClientError,
                _ => RequestHandling.RejectWithServerError
            };
            await ChangeRequestHandlingAsync(handling, args.Context.CancellationTokens);
        }
    })
```

Specify asynchronous delegate for injection notification

```
new ResiliencePipelineBuilder()
    .AddChaosBehavior(new ChaosBehaviorStrategyOptions
    {
        OnBehaviorInjected = static async args => await NotifyAsync(args.Context)
    })
```