

Improving Performance and Memory Use with Streams



Kevin Dockx

ARCHITECT

@KevinDockx <https://www.kevindockx.com>



Coming Up



Advantages of Working with Streams

Using Streams When Reading Data

Using Streams When Sending Data

Testing Performance Improvements

Improving Performance with
Compression



Advantages of Working with Streams



A stream is an abstraction of a sequence of bytes, such as a file, an input/output device or network traffic

- Classes derived from Stream hide specific details of the operating system and the underlying devices



Advantages of Working with Streams



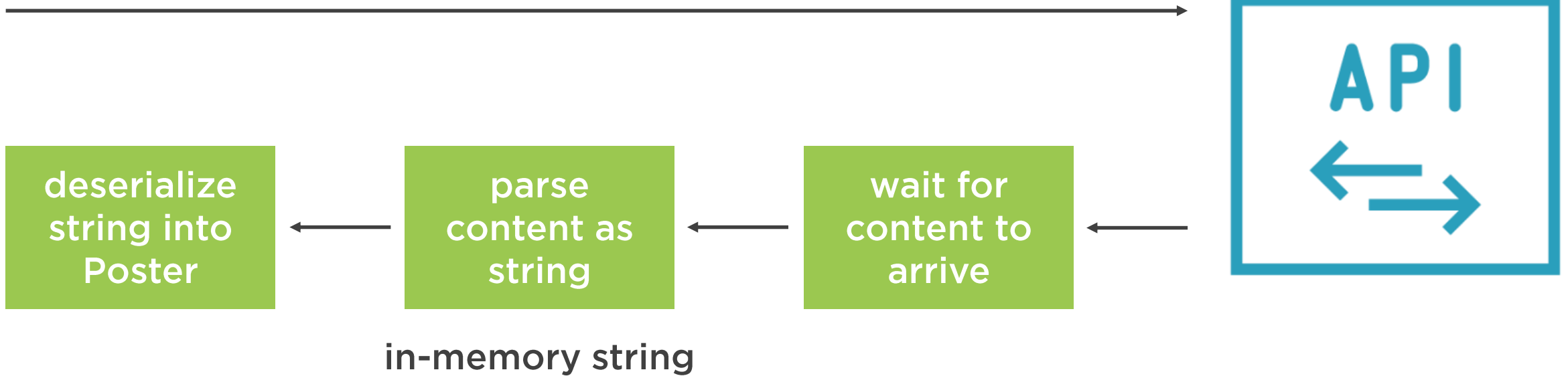
Streams help with avoiding large in-between variables

- Better for memory use
- Better for performance

The API doesn't need to work with streams to get these advantages at client level

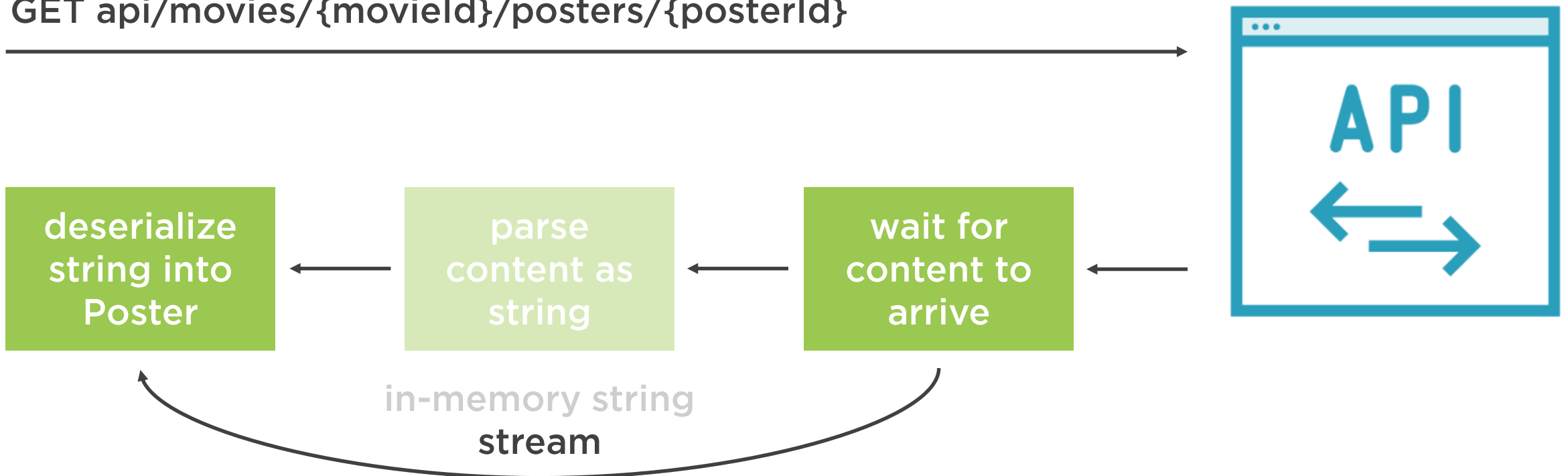
Using Streams When Reading Data

GET api/movies/{movieId}/posters/{posterId}



Using Streams When Reading Data

GET api/movies/{movieId}/posters/{posterId}



Demo

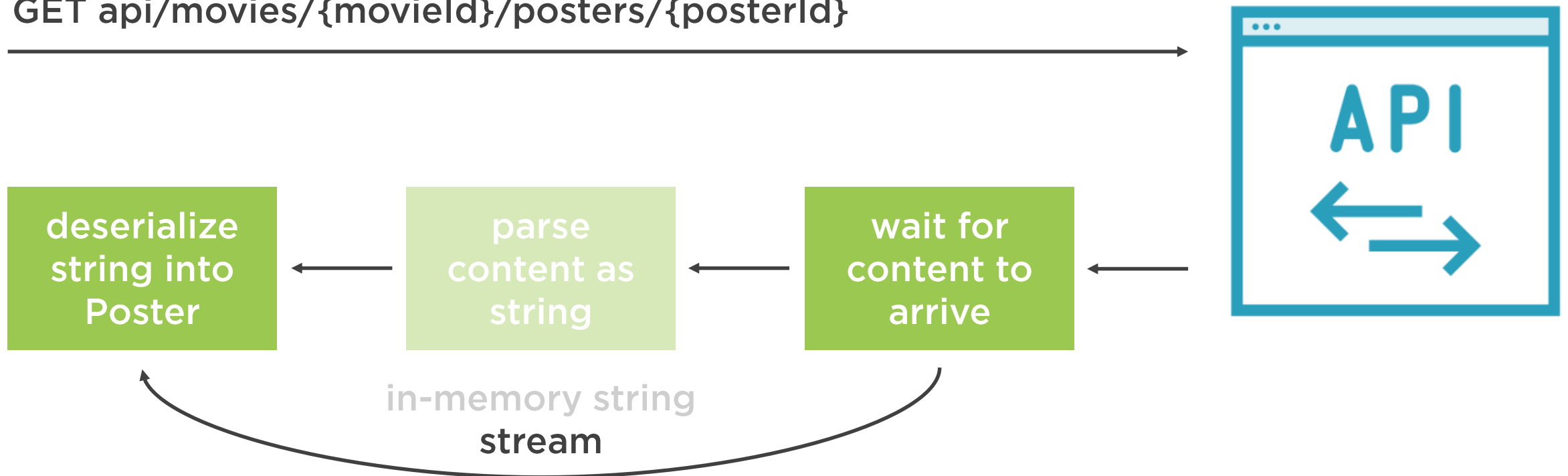


Using Streams When Reading Data



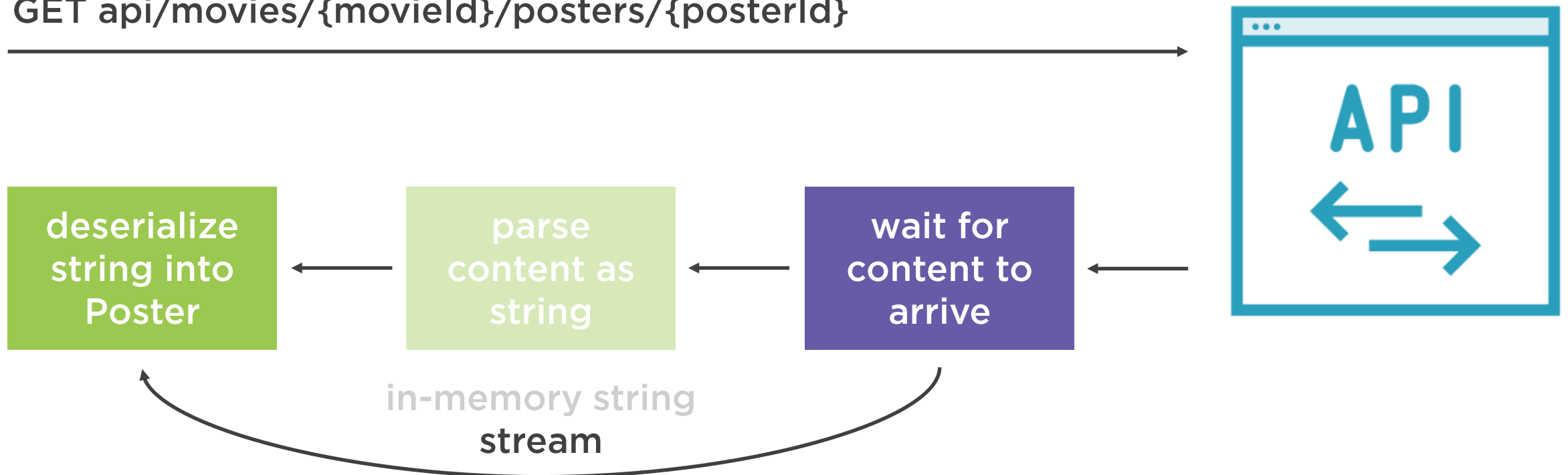
Improving Memory Use and Performance with `HttpCompletionMode`

`GET api/movies/{movieId}/posters/{posterId}`



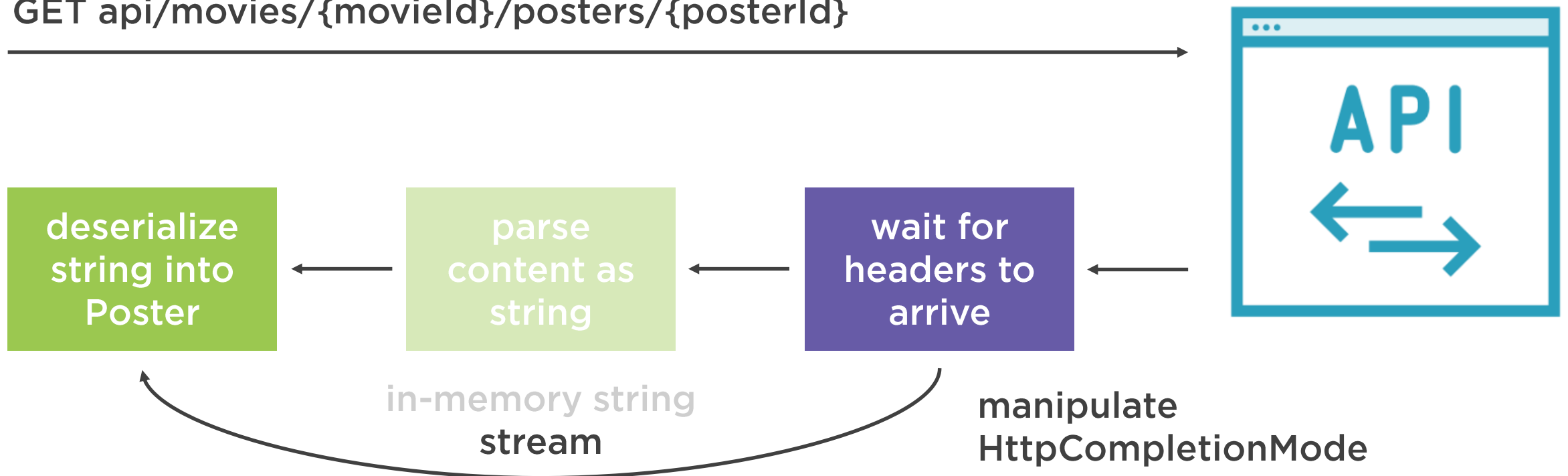
Improving Memory Use and Performance with HttpCompletionMode

GET api/movies/{movieId}/posters/{posterId}



Improving Memory Use and Performance with HttpCompletionMode

GET api/movies/{movieId}/posters/{posterId}



Demo



Improving Memory Use and Performance
with `HttpCompletionMode`



Demo



Improving Code Reuse with an Extension Method



Demo

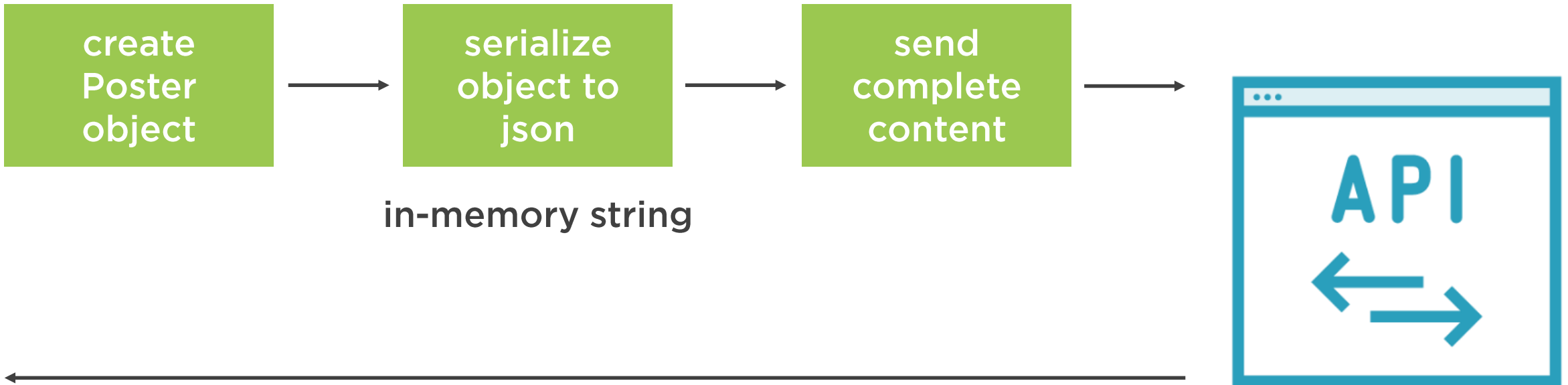


Testing Performance Improvements When Reading Data



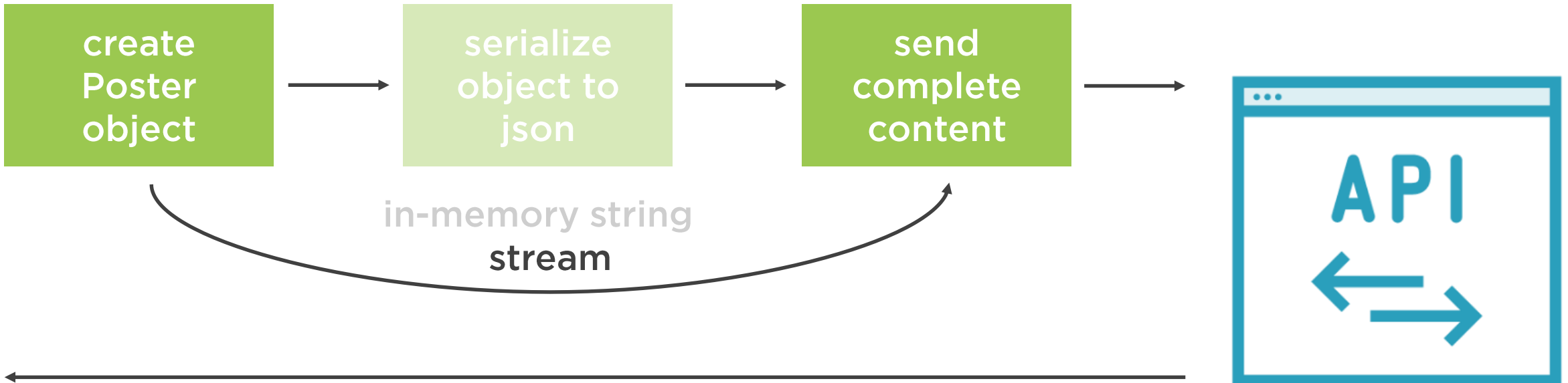
Using Streams When Sending Data

POST api/movies/{movieId}/posters



Using Streams When Sending Data

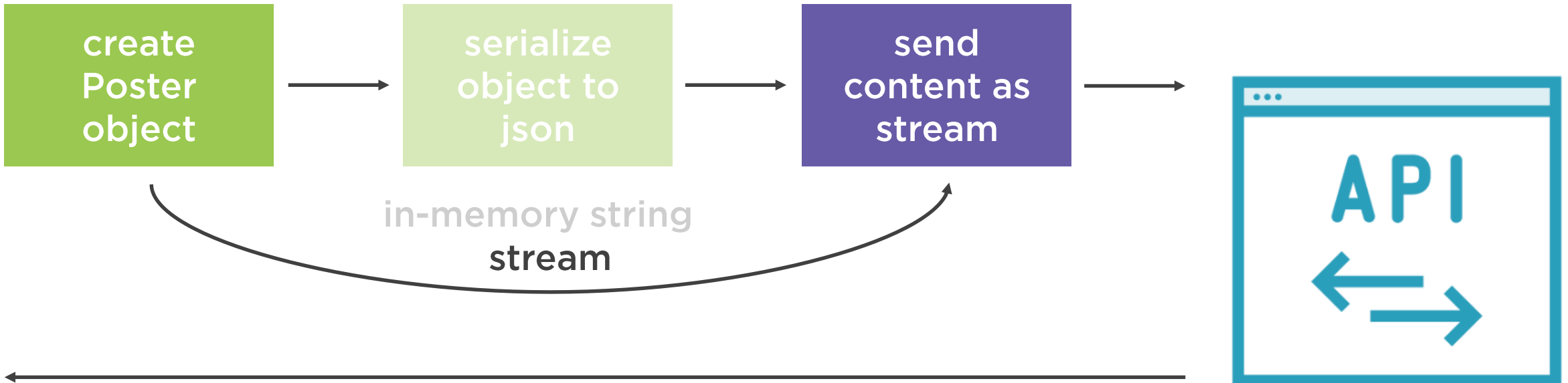
POST api/movies/{movieId}/posters



Using Streams When Sending Data

POST api/movies/{movieId}/posters

StreamContent



Demo



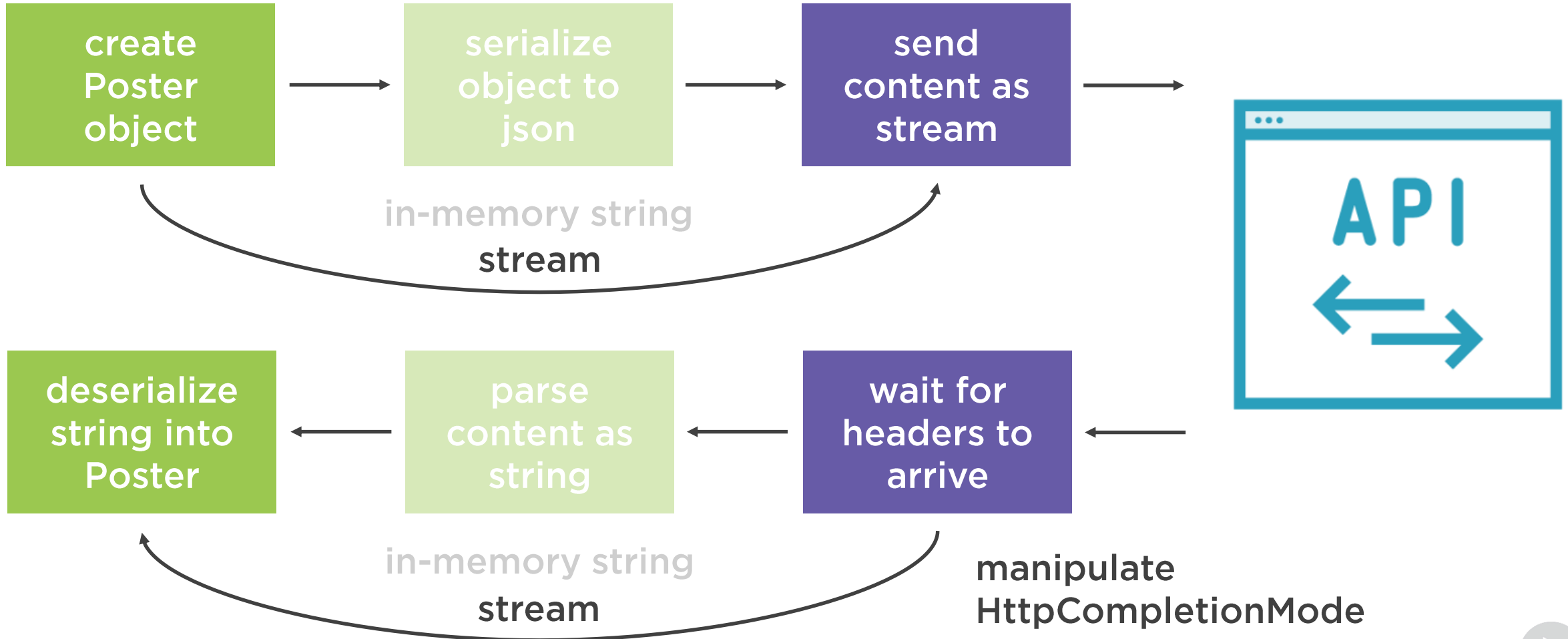
Using Streams When Sending Data



Using Streams When Sending and Reading Data

POST api/movies/{movieId}/posters

StreamContent



Demo



Combining Streams When Sending and Reading Data



Demo



Testing Performance Improvements When Sending Data



On Streaming, Memory Use, and Performance



Creating and disposing streams can cause some overhead

- You may see a direct impact on performance

On Streaming, Memory Use, and Performance



Using streams ensures memory use is kept low

Minimizing memory can also minimize garbage collection, which has a positive impact on performance

On Streaming, Memory Use, and Performance



Always use streams when reading data

Also use streams when sending large amounts of data

If you're not sure, test

Demo



Working with Compression



Summary



Streams are the preferred way of interacting with an API

- Reduced memory footprint
- Improved performance

Streams can be used both when reading and sending data

- Use `HttpCompletionMode` to start streaming the response once response headers have arrived

Summary



Enable compression by setting the Accept-Encoding header & enabling automatic decompression on the HttpClientHandler instance

