Asynchronously Manipulating Resources

Kevin Dockx
ARCHITECT

@KevinDockx https://www.kevindockx.com

Coming Up

Creating a Resource

Supporting Bulk Inserts

Demo

Creating a Resource

Supporting Bulk Inserts

Sending 10 requests to api/books to create 10 books leads to

- Overhead
- Performance loss
- Overuse of threads
  - Scalability suffers

Supporting Bulk Inserts

Supporting bulk inserts leads to

- Less overhead
- Performance increase
- Higher availability of threads
  - Better scalability

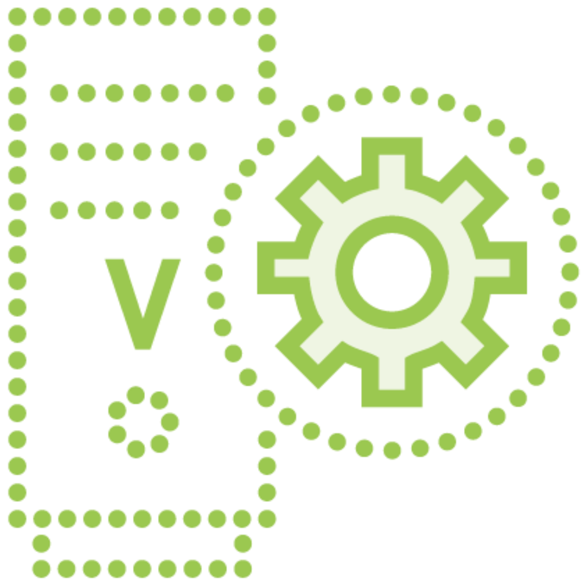
Supporting Bulk Inserts

POST api/books

```
{
  one book
}
```

POST api/bookcollections

```
{
  one collection of books
}
```



Supporting Bulk Inserts

Alternative approach: /bulk endpoint

- A set of resources
- A set of resources and operations on them

Lowers maturity of the API according to the Richardson Maturity Model

Demo

Supporting Bulk Inserts

Demo

Supporting a Location Header for Bulk Inserts

Demo

Testing Bulk Inserts with WebSurge

Summary

Use async for I/O bound work

- Adding an entity to the context isn't I/O bound work

Bulk inserts can help with scalability