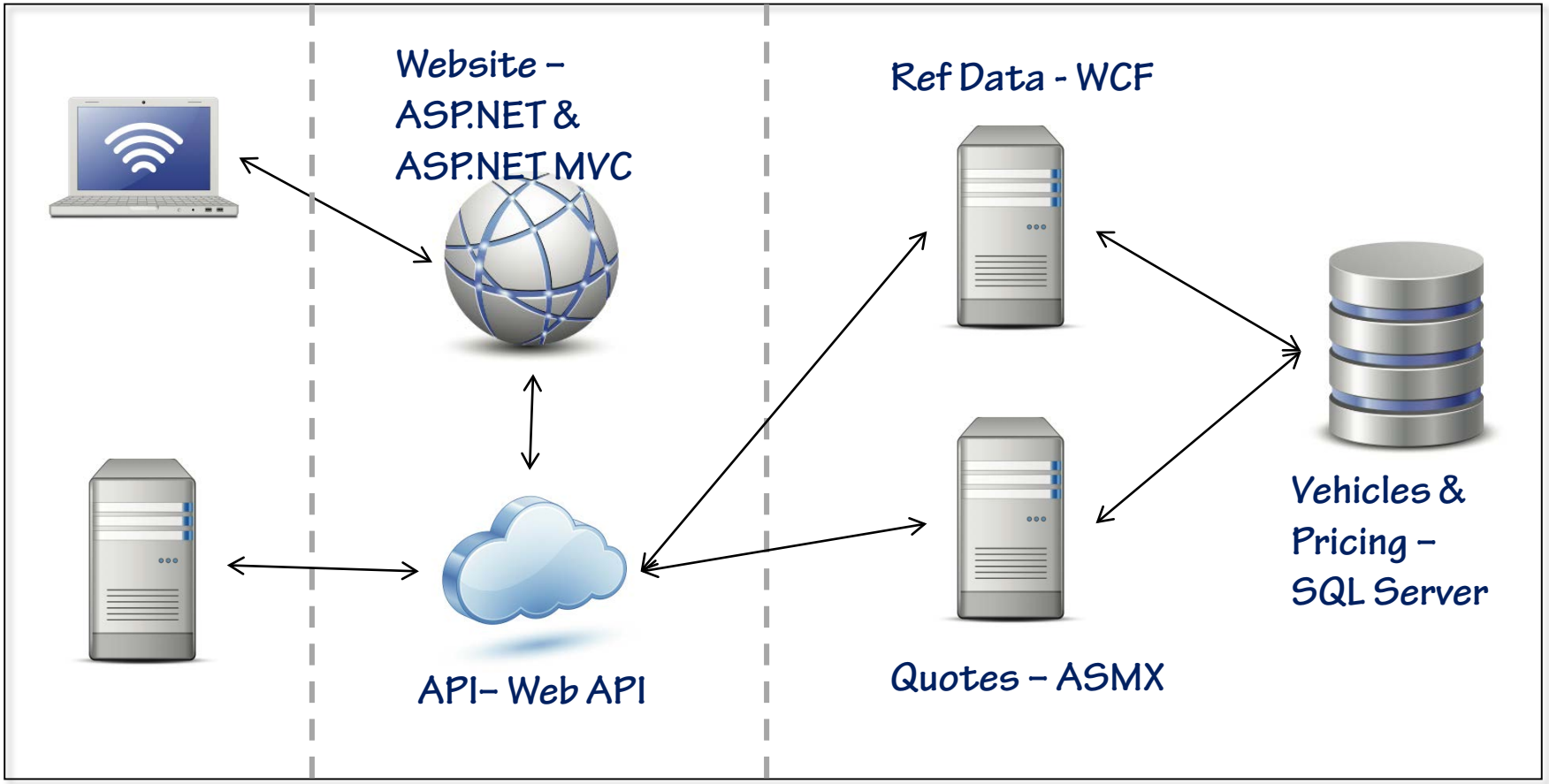
Caching in the .NET Stack: Inside-Out

Elton Stoneman
elton@sixeyed.com

pluralsight
hardcore developer training

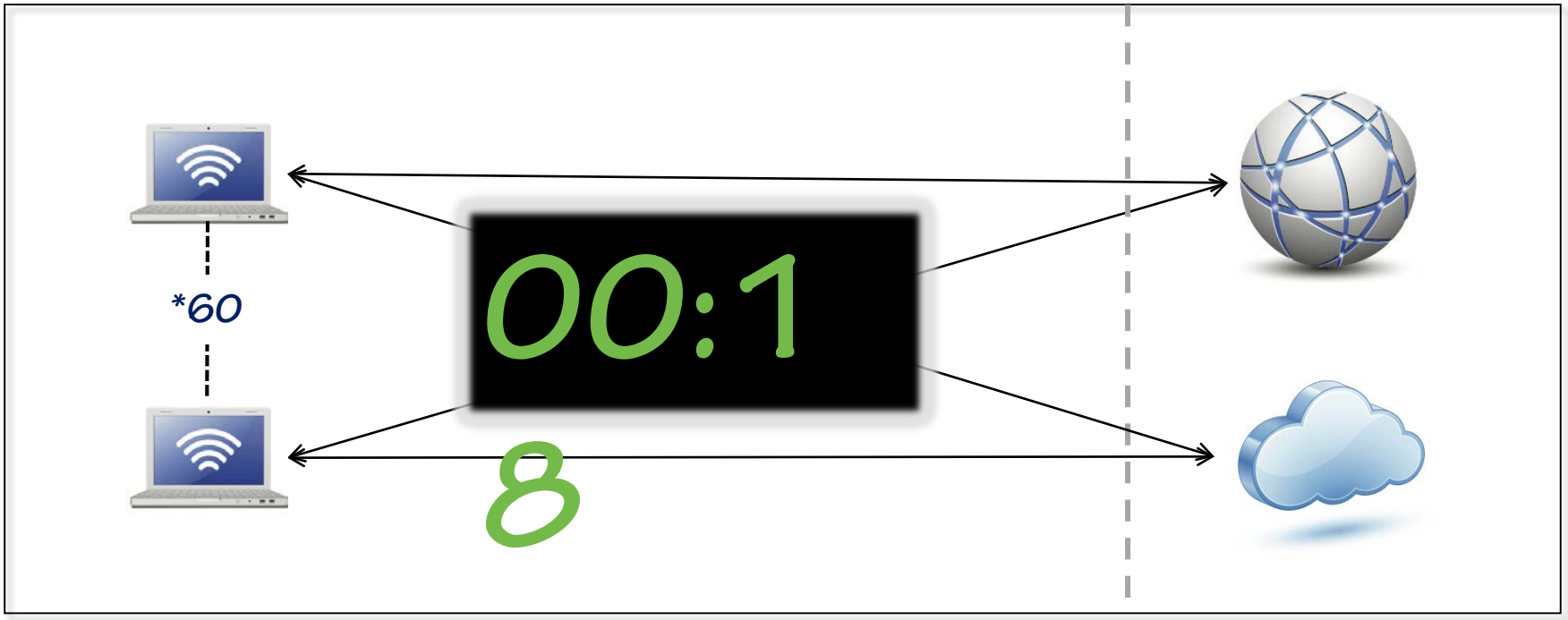
Demo Solution

- V1: no caching



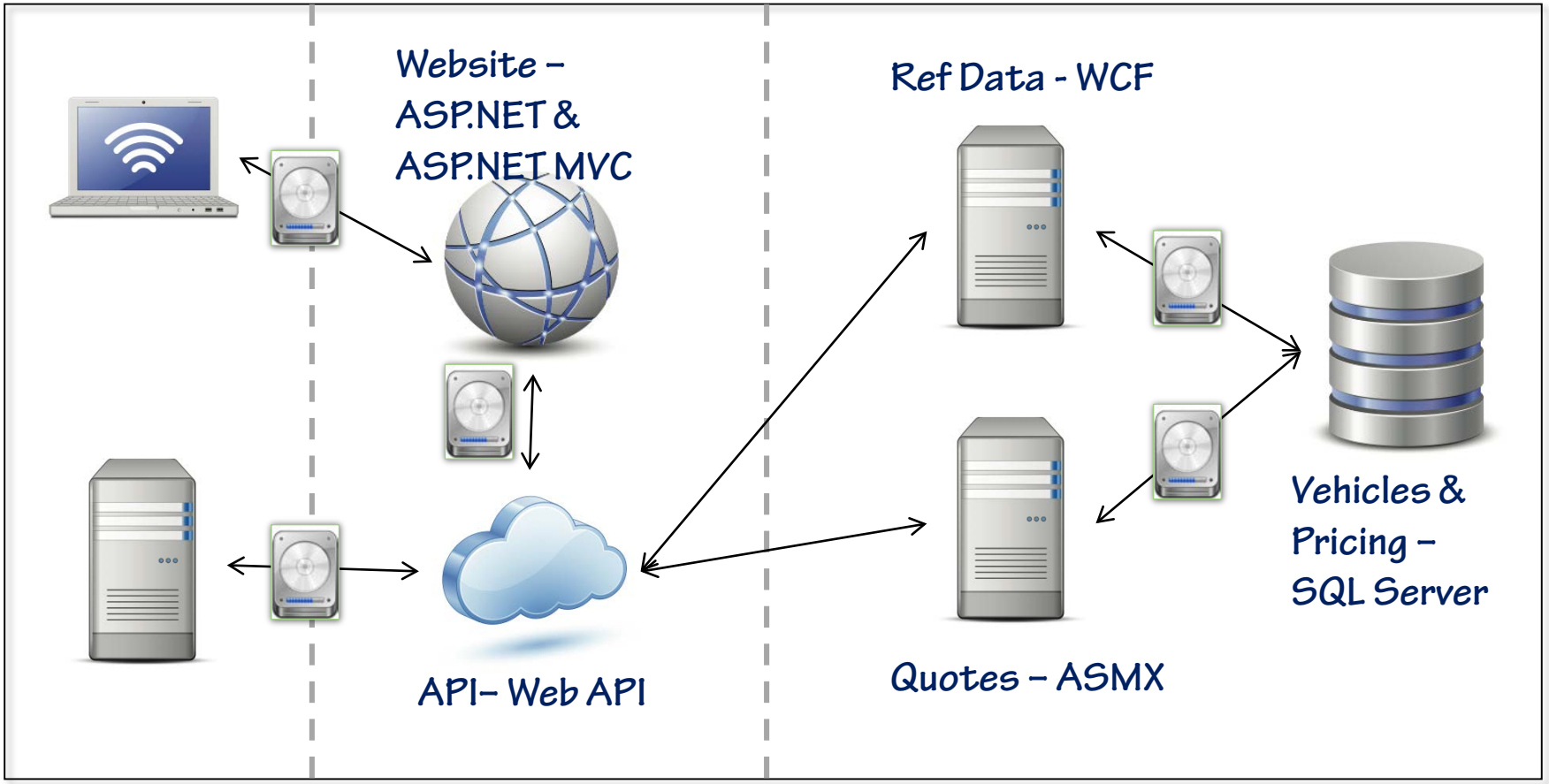
Demo Solution

- **V1 performance**
  - Before caching: 60 concurrent users
  - Scenario: 46 seconds
  - Page load: 5-18 seconds



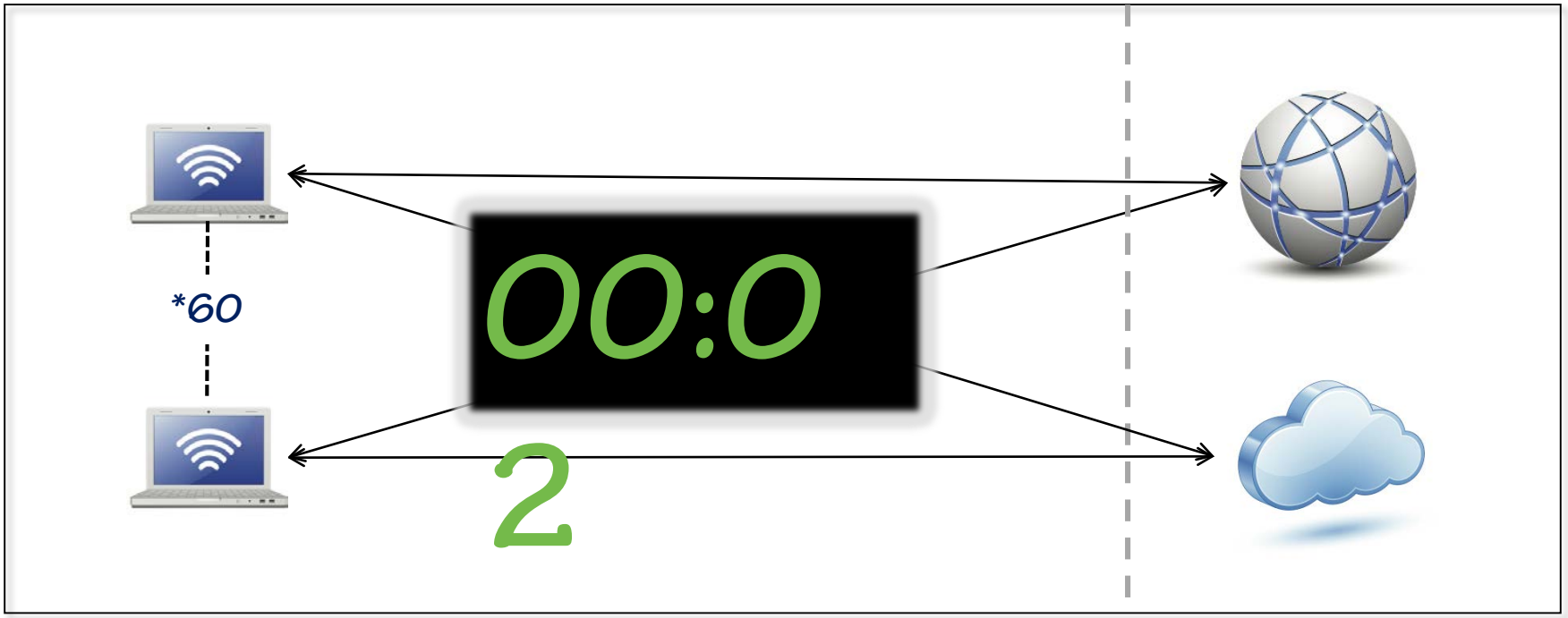
Demo Solution

- V2: solution caching in key areas; HTTP caching for consumers



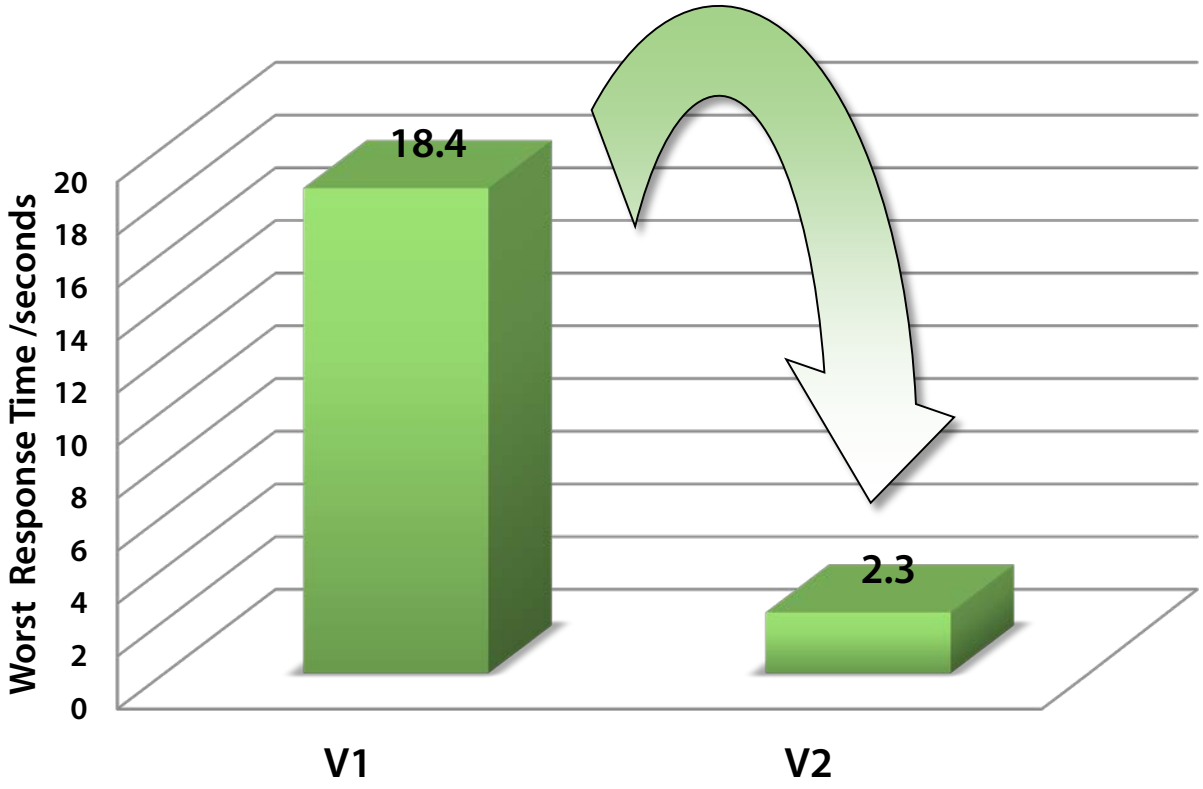
Demo Solution

- **V2 perfomance**
  - With caching: 60 concurrent users
  - Scenario: 8 seconds
  - Page load: 1-2 seconds



Demo Solution

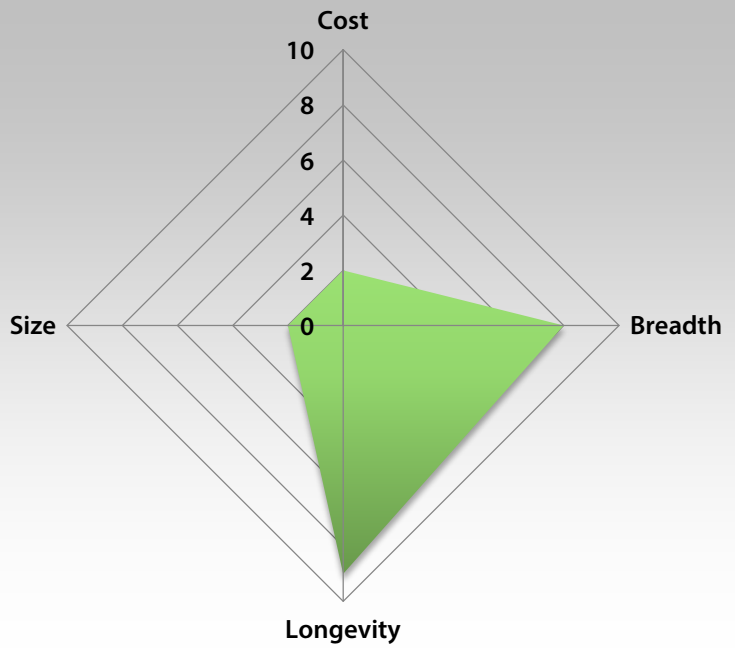
- **KPI: slowest page response**
  - From 18.4 seconds
  - To 2.3 seconds



Guidelines

- Cache Item Decision Matrix

Cost	2
Breadth	8
Longevity	9
Size	2



- Decide correct caching strategy

Guidelines

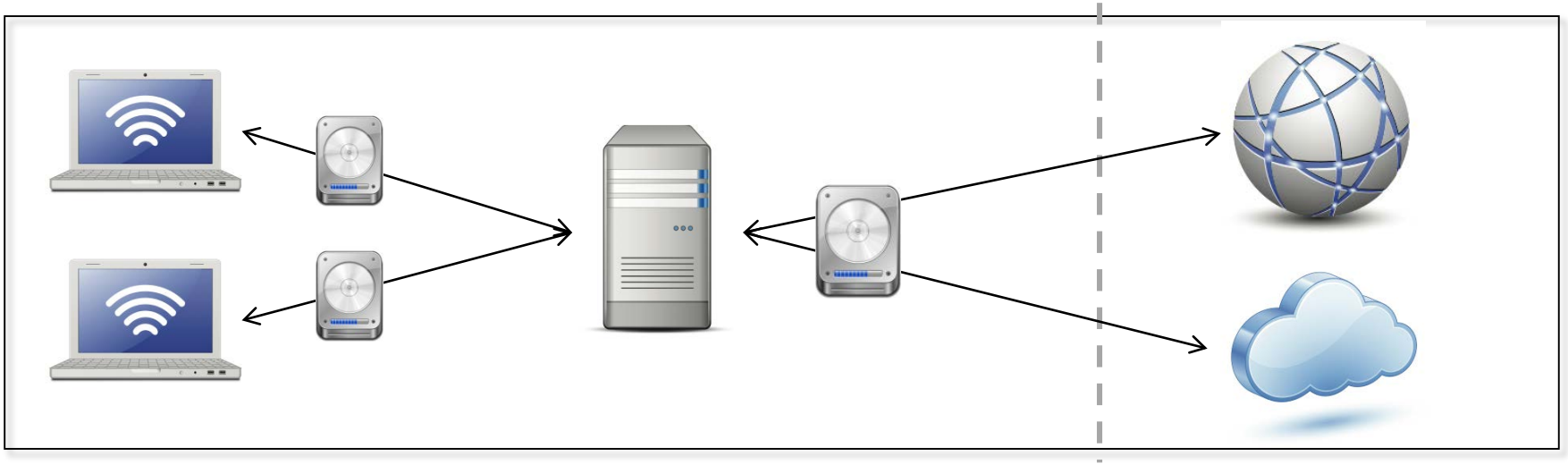
- **Caching Inside the Solution**
- **Multiple cache stores**

	Option 1	*Option 2*
Level 1: fast local	**.NET MemoryCache**	**NCache Express**
Level 2: fast remote	**AppFabric Caching**	**Memcached**
Level 3: large persistent	**DiskCache**	**Azure Table Storage**

- **Configurable cache framework**
  - Simple implementation
  - Low-cost instrumentation
- **Cache aggressively**

Guidelines

- **Caching Outside the Solution**
- **Private and public caches**



- **Follow HTTP recommendations**
  - Validation: Etag and Last-Modified
  - Expiration: Cache-Control
- **Cache conservatively**

What haven't we covered?

- **Inside the Solution**
- **Cache Stores**
  - Local - Enterprise Library Caching Application Block
  - Remote - redis
  - Persistent - MongoDB, SQL Server etc.
- **Advanced cache store functionality**
  - Read-through
  - Write-behind
  - Invalidation
- **ASP.NET Output Caching**
  - Donut & donut-hole approaches

What haven't we covered?

- **Outside the Solution**
- **Custom consumers**
  - HTTP stack
- **Caching proxies**
  - Reverse proxies
- **Cache-Control header**
  - Obscure directives
- **Cache-busting**
  - Querystring approaches

Aims of the Course

- **Caching Techniques**
  - Internal caching with AOP
    - Local, remote, distributed cache stores
  - External caching on the client
  - Invalidation and expiration
- **Technologies**
  - MemoryCache, NCache Express, AppFabric Caching
  - memcacheD, Azure Table Storage, disk
  - HTTP
- **Decision criteria**
  - Longevity, breadth, cost and size