Web Caching

CHAPTER 23

ASP.NET and Self-limiting Caching

The art of web caching

EXPENSIVE VS FREQUENT & SCALABILITY

Caching techniques in ASP

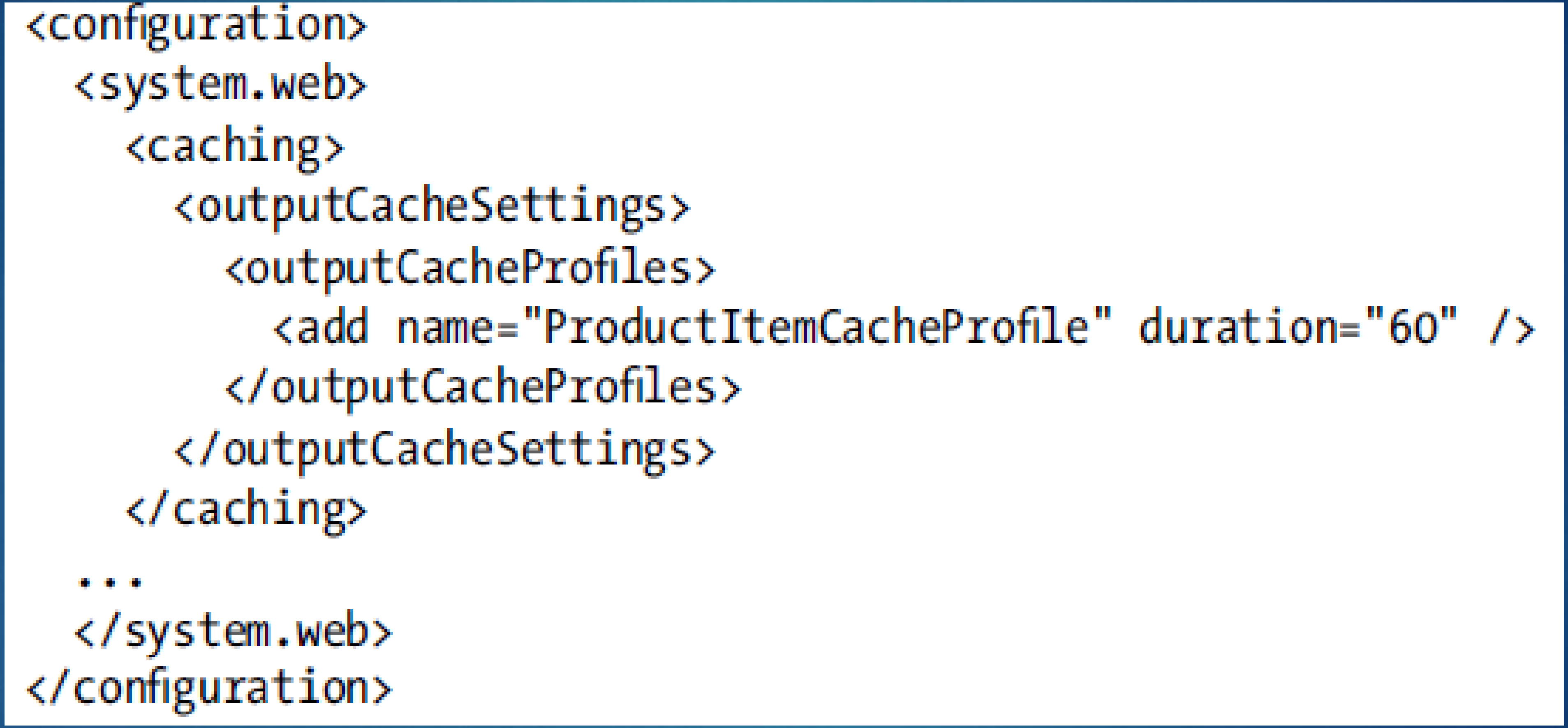
OUTPUT, DATA, FRAGMENT, DATA SOURCE

Output Caching

- Output Cache Directive

```
<%@ OutputCache  Duration=“20”
VaryByParam=“alpha;beta”  %>

<%@ OutputCache CacheProfile="delta"
VaryByParam="pi" %>
```

Cache Profiling



```
<configuration>
  <system.web>
    <caching>
      <outputCacheSettings>
        <outputCacheProfiles>
          <add name="ProductItemCacheProfile" duration="60" />
        </outputCacheProfiles>
      </outputCacheSettings>
    </caching>
  ...
  </system.web>
</configuration>
```

Data Caching

USING THE ASP.NET CACHE OBJECT

Method 1

- Use the Cache object directly. Similar to a Map

  Cache[“**KEY**"] = **primitive/object/value**;

Method 2

- Using the Cache.Insert() function call

```
Cache.Insert(
    key,
    item,
   dependencies,
   absoluteExpiration,
   slidingExpiration
);
```

- Retrieval

```
Object x = (Object) Cache[“X-KEY”];
```

Method 2 : examples

- Cache.Insert(
  ```
      “KEY",
      obj,
      null,
      DateTime.Now.AddMinutes(60),
    TimeSpan.Zero
  );
  ```
- Cache.Insert(
  ```
      “KEY",
      obj,
      null,
      DateTime.MaxValue,
      TimeSpan.FromMinutes(10)
  );
  ```

Fragment Caching

CACHING CERTAIN PARTS OF WEBPAGES

Multiple Views

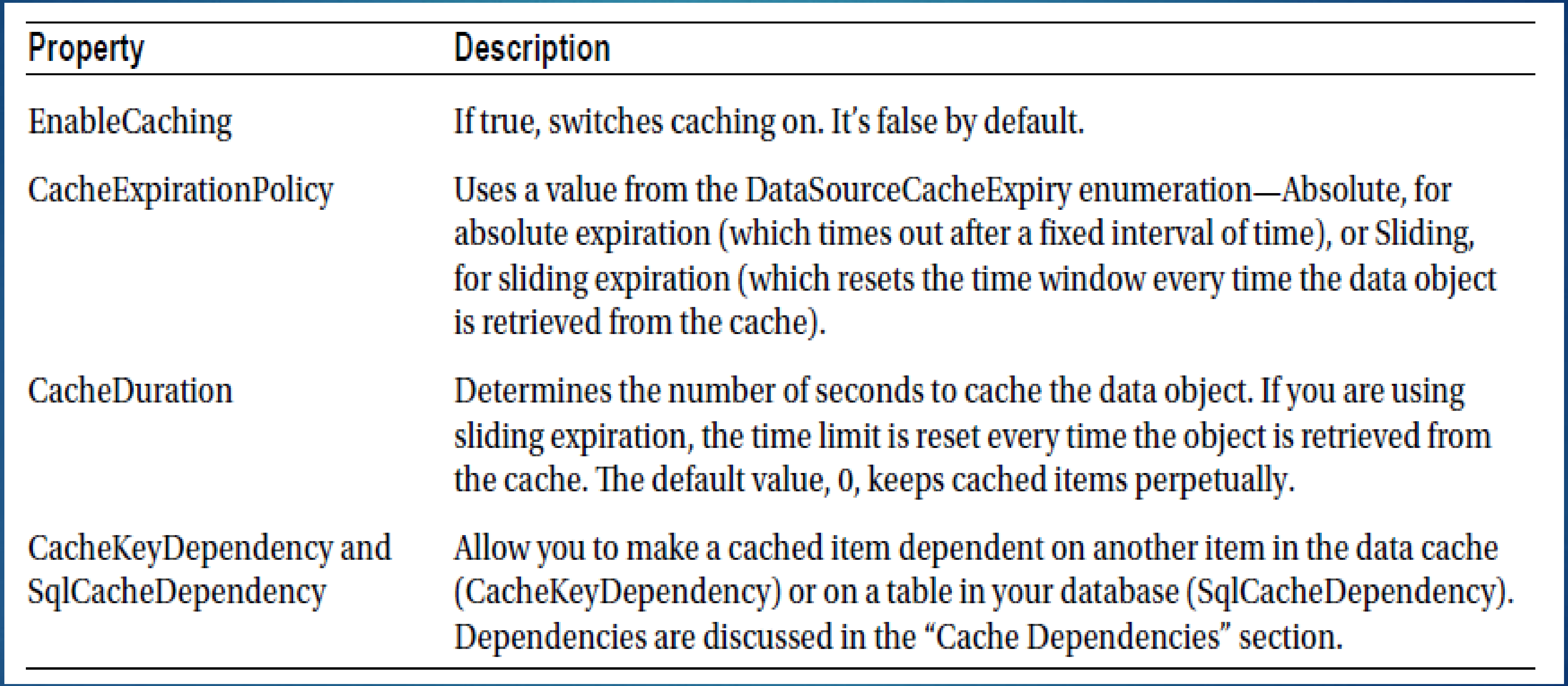
USING DATASET FOR CUSTOM SETS

2 important dataset methods

- ds.copy();
- ds.Tables["Users"].Columns.Remove("Country");

Datasource Caching

Properties



Property	Description
EnableCaching	If true, switches caching on. It's false by default.
CacheExpirationPolicy	Uses a value from the DataSourceCacheExpiry enumeration—Absolute, for absolute expiration (which times out after a fixed interval of time), or Sliding, for sliding expiration (which resets the time window every time the data object is retrieved from the cache).
CacheDuration	Determines the number of seconds to cache the data object. If you are using sliding expiration, the time limit is reset every time the object is retrieved from the cache. The default value, 0, keeps cached items perpetually.
CacheKeyDependency and SqlCacheDependency	Allow you to make a cached item dependent on another item in the data cache (CacheKeyDependency) or on a table in your database (SqlCacheDependency). Dependencies are discussed in the "Cache Dependencies" section.

Example

- <asp:SqlDataSource ID=“dataSource1" runat="server"

  ProviderName="System.Data.SqlClient"

  EnableCaching="True" CacheDuration="3600"

  ConnectionString="<%$ ConnectionStrings:StringName %>"

  SelectCommand="SELECT * City FROM test">

  </asp:SqlDataSource>

Filtering using SqlDataSource

```
<asp:SqlDataSource ID="sourceEmployees" runat="server"
  ProviderName="System.Data.SqlClient"
  ConnectionString="<%$ ConnectionStrings:Northwind %>"
  SelectCommand= "SELECT name, City, county, phone FROM Employees"
  FilterExpression=" City='{0}' " EnableCaching="True">
      <FilterParameters>
          <asp:ControlParameter ControlID="DropDownList1" Name="City"
          PropertyName="SelectedValue" />
      </FilterParameters>
  </asp:SqlDataSource>
```

ObjectDataSource Caching

EXACTLY SIMILAR TO SQLDATASOURCE

Cache Dependencies

File/Directory Dependency

- CacheDependency prodDependency = new CacheDependency(
   Server.MapPath(“Products.xml"));

- Cache.Insert("ProductInfo", prodInfo, prodDependency);

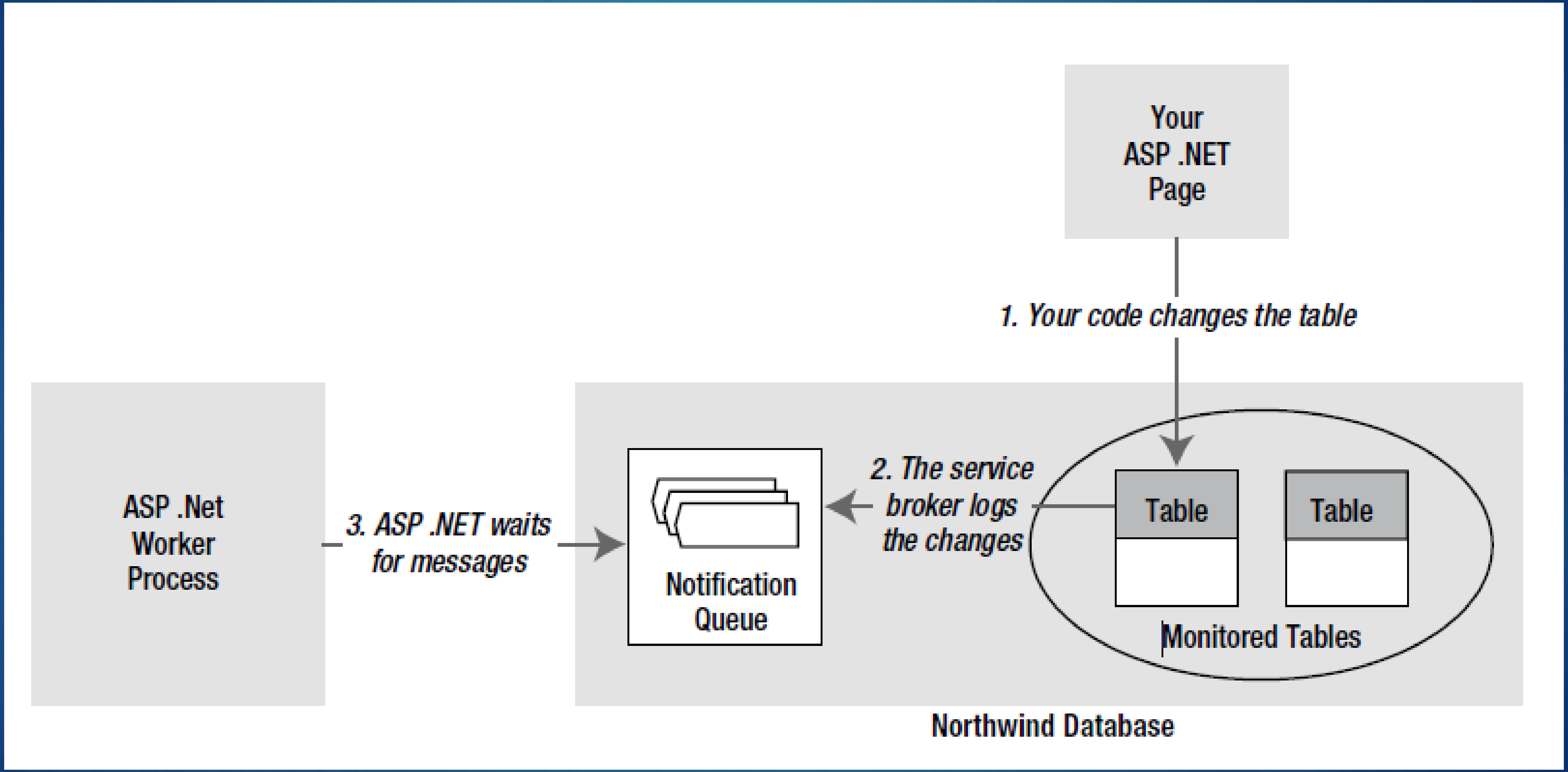
Cached item Dependency

-
  ```
  Cache["Key1"] = "Cache Item 1";
  Cache["Key2"] = "Cache Item 2";

  string[] dependencyKey = {“Key1”, “Key2”};

  CacheDependency dependency = new CacheDependency(
    null,
    dependencyKey
  );
  ```

-
  ```
  Cache.Insert("Key3", "Cache Item 3", dependency);
  ```

SQLDataSource Dependency

How it works


Your ASP .NET Page
1. Your code changes the table
ASP .Net Worker Process
3. ASP .NET waits for messages
Notification Queue
2. The service broker logs the changes
Table
Table
Monitored Tables
Northwind Database

Mandatory Conditions

- You must fully qualify table names in the form [Owner].table
- Your query cannot use an aggregate function, such as COUNT(), MAX(), MIN(), or AVERAGE()
- You cannot select all columns with the wildcard *
- Example :
  SELECT EmployeeID, FirstName, LastName, City FROM dbo.Employees

Enabling Service Broker

- USE Northwind
- ALTER DATABASE Northwind SET ENABLE_BROKER
- GO

Initialising the Caching Service

- SqlDependency.Start(connectionString);
- In your cs file

Creating the SqlCacheDependency

- SqlConnection con = new SqlConnection(connectionString);
- String query = “SELECT EmployeeID, FirstName, LastName, City FROM dbo.Employees”;
- SqlCommand cmd = new SqlCommand (con, query);
- DataSet ds = new DataSet();
  - adapter.fill(ds,”employees”);
- SqlCacheDependency empDependency = new SqlCacheDependency(cmd);
- Cache.Insert("Employees", ds, empDependency);

AIT ends