



CONFIGURATION

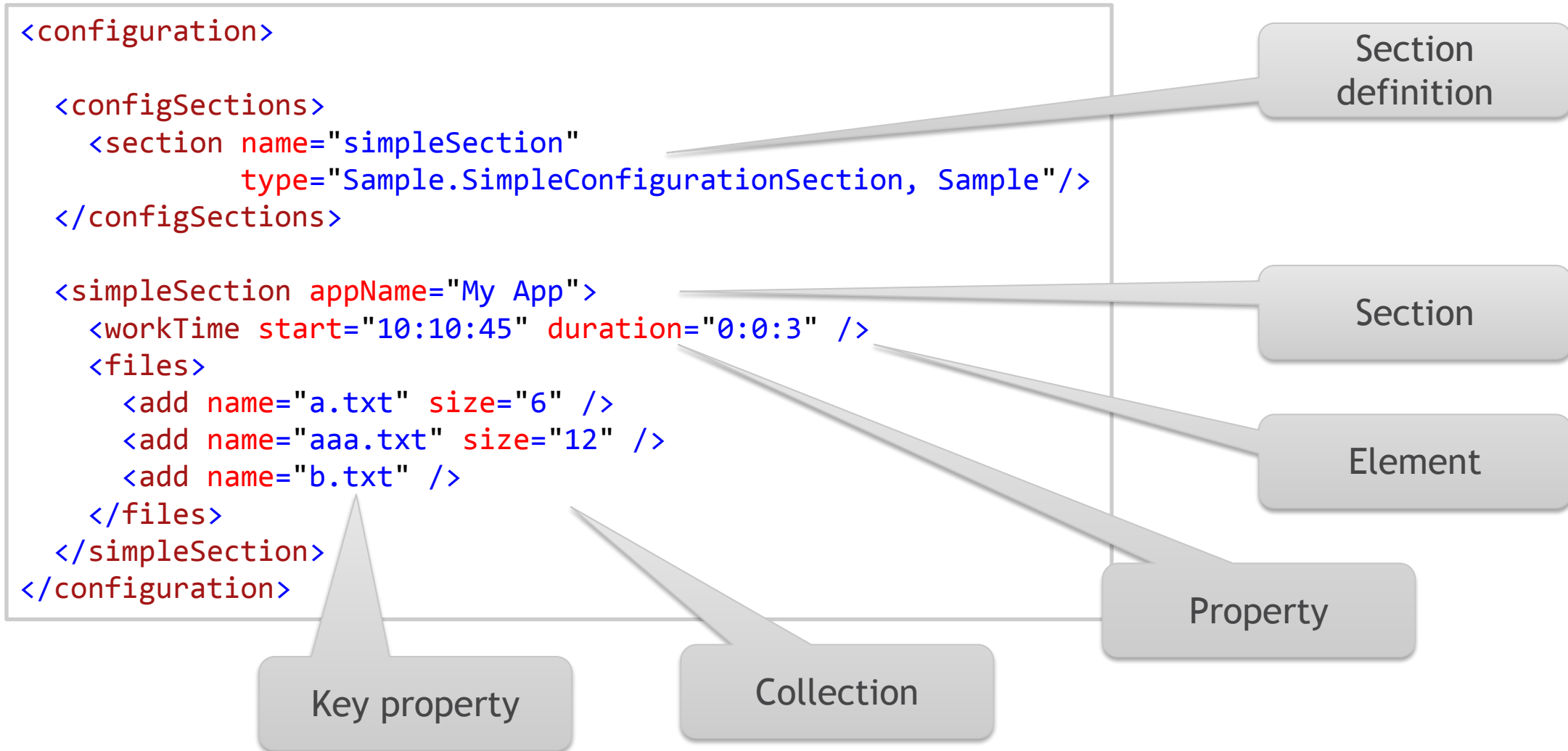
MODULE 4.6

AGENDA

- Basic of configuration files
- Extended features

BASIC CONFIGURATION FILES

CONFIGURATION FILE STRUCTURE



COMMON CLASSES

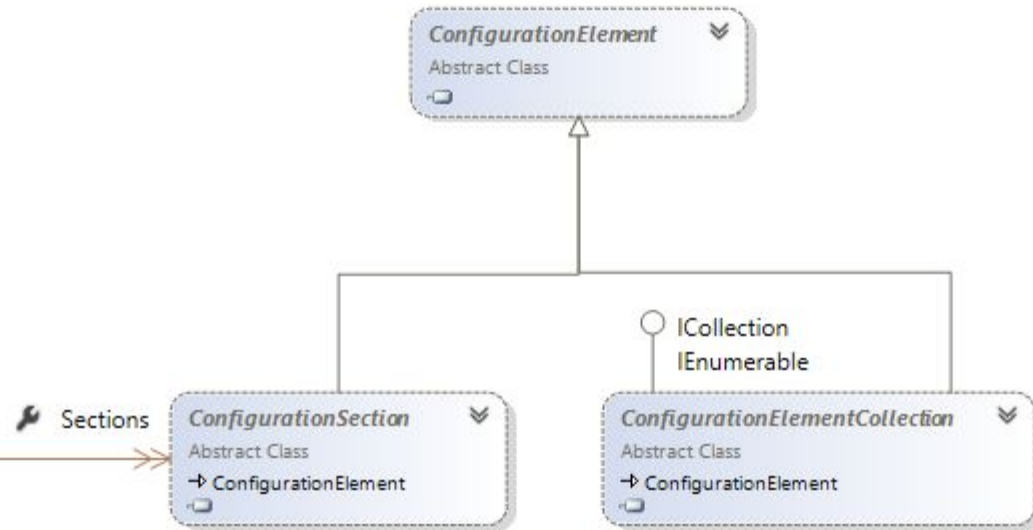
Configuration
Sealed Class

Properties

- AppSettings
- AssemblyStringTransformer
- ConnectionStrings
- EvaluationContext
- FilePath
- HasFile
- Locations
- NamespaceDeclared
- RootSectionGroup
- SectionGroups
- TargetFramework
- TypeStringTransformer

Methods

- GetSection
- GetSectionGroup
- Save (+ 2 overloads)
- SaveAs (+ 2 overloads)



ConfigurationManager
Static Class

Properties

- AppSettings
- ConnectionStrings

Methods

- GetSection
- OpenExeConfiguration (+ 1 overload)
- OpenMachineConfiguration
- OpenMappedExeConfiguration (+ 1 overload)
- OpenMappedMachineConfiguration
- RefreshSection

ELEMENTS

```
public class WorkTimeElement : ConfigurationElement
{
    [ConfigurationProperty("start")]
    public DateTime StartTime
    {
        get { return (DateTime)this["start"]; }
    }

    [ConfigurationProperty("duration")]
    public TimeSpan Duration
    {
        get { return (TimeSpan)this["duration"]; }
    }
}
```

COLLECTIONS

```
public class FileElement : ConfigurationElement
{
    [ConfigurationProperty("name", IsKey = true)]
    public string FileName
    {
        get { return (string)base["name"]; }
    }

    [ConfigurationProperty("size")]
    public int FileSize
    {
        get { return (int)base["size"]; }
    }
}
```

```
public class FileElementCollection : ConfigurationElementCollection
{
    protected override ConfigurationElement CreateNewElement()
    {
        return new FileElement();
    }

    protected override object GetElementKey(ConfigurationElement element)
    {
        return ((FileElement)element).FileName;
    }
}
```

SECTIONS

```
public class SimpleConfigurationSection :
ConfigurationSection
{
    [ConfigurationProperty("appName")]
    public string ApplicationName
    {
        get { return (string)base["appName"]; }
    }

    [ConfigurationProperty("workTime")]
    public WorkTimeElement WorkTime
    {
        get { return (WorkTimeElement)this["workTime"]; }
    }

    [ConfigurationProperty("files")]
    public FileElementCollection Files
    {
        get { return (FileElementCollection)this["files"]; }
    }
}
```


USAGE

```
var s = (SimpleConfigurationSection)
    ConfigurationManager.GetSection("simpleSection");

Console.WriteLine("{0} {1} - {2}",
    s.ApplicationName,
    s.WorkTime.StartTime.ToLongTimeString(),
    (s.WorkTime.StartTime + s.WorkTime.Duration).ToLongTimeString());

foreach (FileElement file in s.Files)
{
    Console.WriteLine("{0} - {1}", file.FileName, file.FileSize);
}
```

EXTENDED CONFIGURATION FILES

PROPERTIES AND COLLECTIONS

```
[ConfigurationProperty("size",  
    IsRequired = false,  
    DefaultValue = 0)]  
public int FileSize  
{  
    get { return (int)base["size"]; }  
    set { this["size"] = value; }  
}
```

```
<files>  
  <file name="a.txt" size="6" />  
  <file name="aaa.txt" size="12" />  
  <file name="b.txt" />  
</files>
```

```
[ConfigurationCollection(typeof(FileElement),  
    AddItemName = "file",  
    ClearItemsName = "clear",  
    RemoveItemName = "del")]  
[ConfigurationProperty("files")]  
public FileElementCollection Files  
{  
    get { return  
(FileElementCollection)this["files"]; }  
}
```

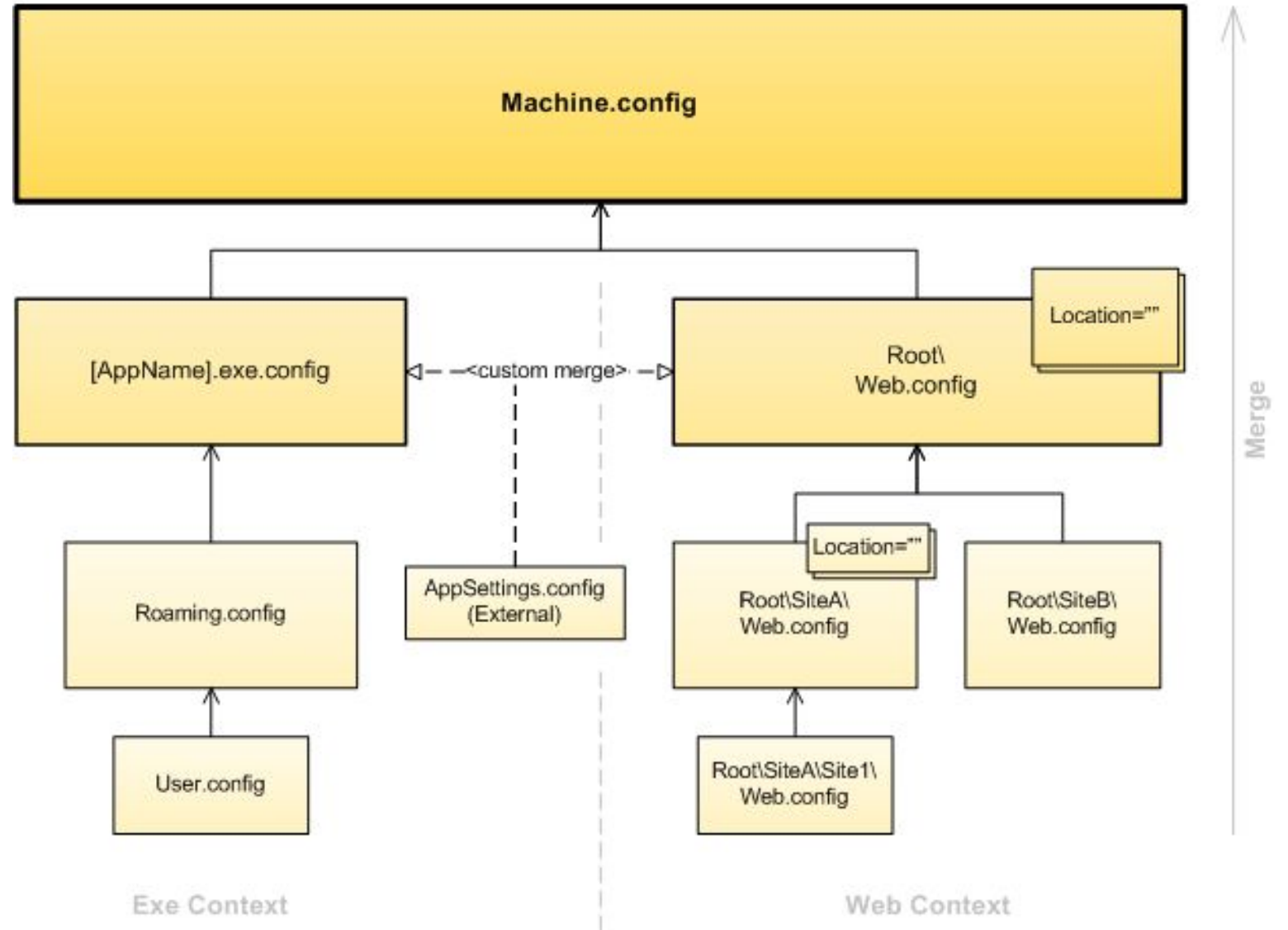
VALIDATORS

```
[ConfigurationProperty("size", IsRequired = false, DefaultValue = 0)]  
[IntegerValidator(MinValue = 0, MaxValue = 1000000)]  
public int FileSize  
{  
    get { return (int)base["size"]; }  
    set { this["size"] = value; }  
}
```

Validator	Sample
IntegerValidator / LongValidator	[IntegerValidator(MinValue = 0, MaxValue = 1000000)]
TimeSpanValidator / PositiveTimeSpanValidator	[TimeSpanValidator(MaxValueString = "0:30:0")]
StringValidator	[StringValidator(MaxLength = 3)]
RegexStringValidator	[RegexStringValidator("[abc]{3}")]
CallbackValidator	[CallbackValidator(CallbackMethodName = "Validate", Type = typeof(MyValidator))]

CONFIGURATION HIERARCHY

- %SYSTEMROOT%\Microsoft.NET\Framework\\CONFIG
 - machine.config
 - web.config



CONFIGURATION FILE MODULARIZATION

App.config

```
<configuration>
  <configSections>
    <section name="simpleSection"
      type="Sample.SimpleConfigurationSection, Sample"/>
  </configSections>
  <simpleSection configSource="User.config"/>
</configuration>
```

User.config

```
<simpleSection appName="My App">
  <workTime start="10:10:45" duration="0:0:3" />
  <files>
    <file name="a.txt" size="6" />
    <file name="aaa.txt" size="12" />
    <file name="b.txt" />
  </files>
</simpleSection>
```

PREDEFINED SECTIONS

```
<configuration>

  <connectionStrings>
    <add name="myDB" connectionString="server=localhost"/>
  </connectionStrings>

  <appSettings>
    <add key="Key1" value="1"/>
    <add key="Key2" value="2"/>
  </appSettings>

</configuration>
```

```
Console.WriteLine(
    ConfigurationManager.ConnectionStrings["myDB"].ConnectionString);

Console.WriteLine(
    ConfigurationManager.AppSettings["Key2"]);
```

ADDITIONAL MATERIALS

Deep Dive into .Net Configurationn

- [Unraveling the Mysteries of .NET 2.0 Configuration](#)
- [Decoding the Mysteries of .NET 2.0 Configuration](#)
- [Cracking the Mysteries of .NET 2.0 Configuration](#)

THANK YOU!