

0.1 MEF - Extentebility Framework

- Ligger i standard frameworket
- Skulle være universelt anvendeligt
- Skal kunne udvide ved blot at uploade ny dll-fil
- *Unity* gør konfigurerbar - ikke udvidelse
- *PRISM* er rettet mod brugergrænseflader
- *System.Addin* vil også lave plugins, men sikkerheden er lidt for god. Det er også meget kompliceret at bruge.
- Kan stille interfaces til rådighed
- Der er ikke en centraliseret konfiguration for MEF
-

En *Hello World*-opbygning:

```

1 using System.ComponentModel.Composition;
...
3 public class Program
{
5     public static void Main(string[] args)
    {
7         Program p = new Program();
        p.Run();
9     }

11    public void Run()
    {
13        Compose();
    }

15
16    private void Compose()
17    {
18        var container = new CompositionContainer();
19        container.ComposeParts(this);
20    }
21 }

```

- Contract
 - Normalt et interface (ofte)
 - Normalt også en datatype (men kan være en string)
 - Skriv *[Export]* over klassen der skal eksporteres
 - Skriv *[Export(typeof(...))]* for at specificere en klasse fra et interface
 - Skriv *[Export("funktionsNavn")]* for at specificere en properties
 - Der kan bruges *Shared* og *Non-Shared*
- Import
 - *[Import]* for klasse

- *[ImportingConstructor]* for en hel klasse
- Hvis der skal importeres flere af samme type, skriv *[ImportMany(AllowRecomposition=true)]*
- Kan gøres *Lazy*, hvilket vil sige, at den først initieres, når den skal bruges.
- Catalog
 - Opslag over kendte kontrakter, dog ikke initieret endnu
 - * *TypeCatalog*: Tilføj selv typer
 - * *AssemblyCatalog*: Finder assemblies med bestemt type
 - * *DirectoryCatalog*: Finder et bestemt katalog og dennes indhold
 - * *AggregatinCatalog*: En samling af de tre overstående
- Composition Container
 - Denne laver arbejdet
 - Skal have et catalog og initierer funktioner og klasser.