



Tema 2 PSSC

Dezvoltarea de sisteme software extensibile folosind **Managed Extensibility Framework**

Una din cele mai noi platforme de proiectie software extensibile este Managed Extensibility Framework (MEF). Platforma MEF permite dezvoltarea de aplicații .Net extensibile, deoarece MEF e componentă a platformei .NET 4.0 și tot ce ține de acest framework e conținut în librăria System.ComponentModel.Composition. Această librărie găzduiește atributele ce vor fi folosite în realizarea compunerii și câteva clase destinate tratării diferitelor excepții.

MEF se fundamentează pe trei concepte fundamentale care conferă valoare și utilitate acestei platforme: extensibilitate, detecție și metadata.

Extensibilitate

Extensibilitatea se realizează la execuție (de aceea, ea se mai numește și extensibilitate dinamică). Extensiile sunt integrate și injectate în sistem la rularea aplicației, și nu este nevoie de re-compilare, pentru ca extensiile să poată fi detectate și utilizate ca părți componente ale acelei aplicații. Deși există numeroase alte soluții care tratează această arie a extensibilității, dintre care putem aminti de Managed Addin Framework (MAF) sau de *framework-uri* de IoC (StructureMap, Unity).

Detecție

Acest concept face referire la modul în care detectăm *plug-in*-urile ce vor fi încărcate în sistemul nostru extensibil, decuplat. Premisa de bază este următoarea: la crearea unei clase, dorim ca acea clasă să realizeze ceva specific, un serviciu. Detecția, reprezentând mecanismul de detectare și furnizare a implementărilor astfel încât celelalte componente ale sistemului să le poată utiliza, la nevoie.

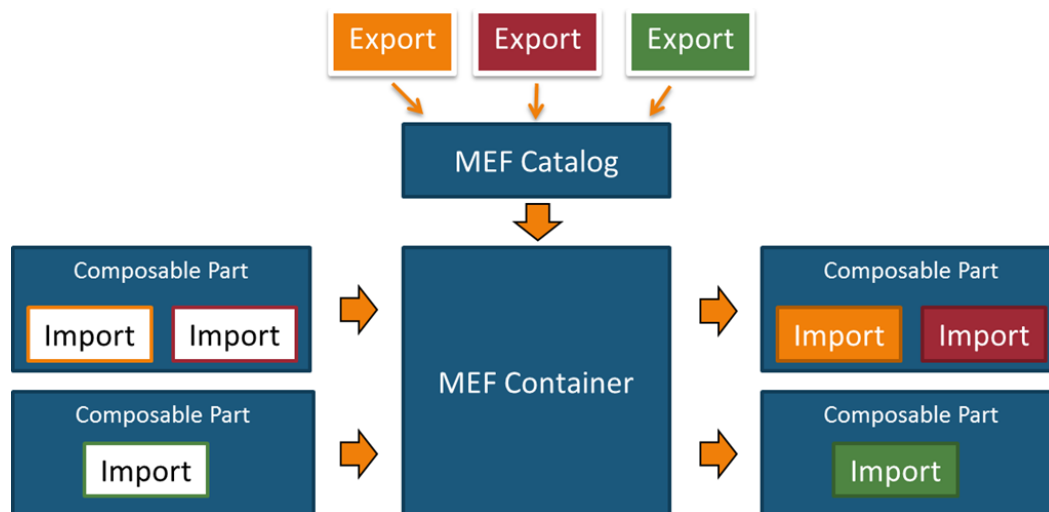
Metadata

Reprezintă tehnologia MEF prin care se pot furniza aspecte, capabilități și informații adiționale despre componente. Prin intermediul metadata, aplicația poate să filtreze și să examineze componentele, fără a cunoaște detalii despre implementarea acestora. Metadata oferă o abordare contextuală de a determina ce componente sunt potrivite pentru mediul curent, filtrându-le în mod corespunzător și apoi, încărcându-le.



Un sistem software este compus din parti compatibile. O astfel de parte oferă servicii altor părți și la rândul ei consumă servicii oferite de alte părți. Principalele concepte care trebuie înțelese pentru a putea folosi platforma MEF sunt:

- 1) export – reprezintă un serviciu oferit de o parte
- 2) import – reprezintă un serviciu consumat de o parte
- 3) contract – un contract este un identificator pentru un export sau un import
- 4) compunere – părțile sunt compuse de către MEF care le instanțiază și apoi face legătura între importuri și exporturi.



Modelul MEF bazat pe compunere implică utilizarea unui mecanism de detecție automată a părților, la execuție. Acest mecanism se realizează prin folosirea unui obiect ce poartă numele de Catalog. MEF oferă *out-of-the-box* patru tipuri de cataloage, clasificare indusă de modalitatea prin care dorim să detectăm părțile:

- * **Type catalog** - detectează toate părțile asociate unei colecții de tipuri.
- * **Assembly catalog** - detectează toate părțile conținute într-un ansamblu (bibliotecă).
- * **Directory catalog** - scanează un director de pe disc, pe baza unei locații specifice
- * **Aggregate catalog** - modalitate prin care să putem grupa toate părțile detectate de cataloage diferite, iar pentru asta, vom întrebuința un catalog agregat care va cuprinde o colecție de cataloage.