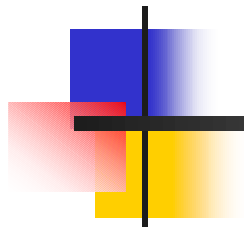
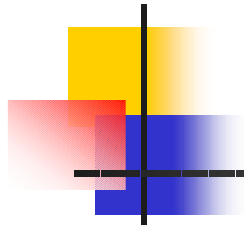


Zwischenbericht zur Diplomarbeit

Christian Wurbs



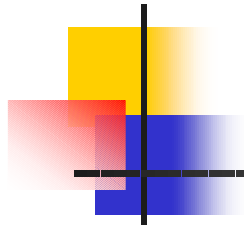
Modellierung .NET basierter Anwendungssysteme mit der UML



Überblick



- Aufgabenstellung
- Was ist .NET ?
- .NET vs. EJB Komponenten
- MDA und EDOC
- Profileanforderung
- .NET Profile
- Aufgaben – Fragen – Ausblick



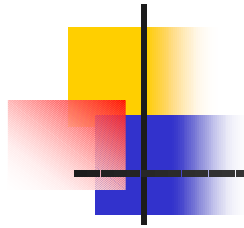
Aufgabenstellung



Erstellung eines Konzeptes zur Modellierung von .NET basierten Anwendungen mittels der UML

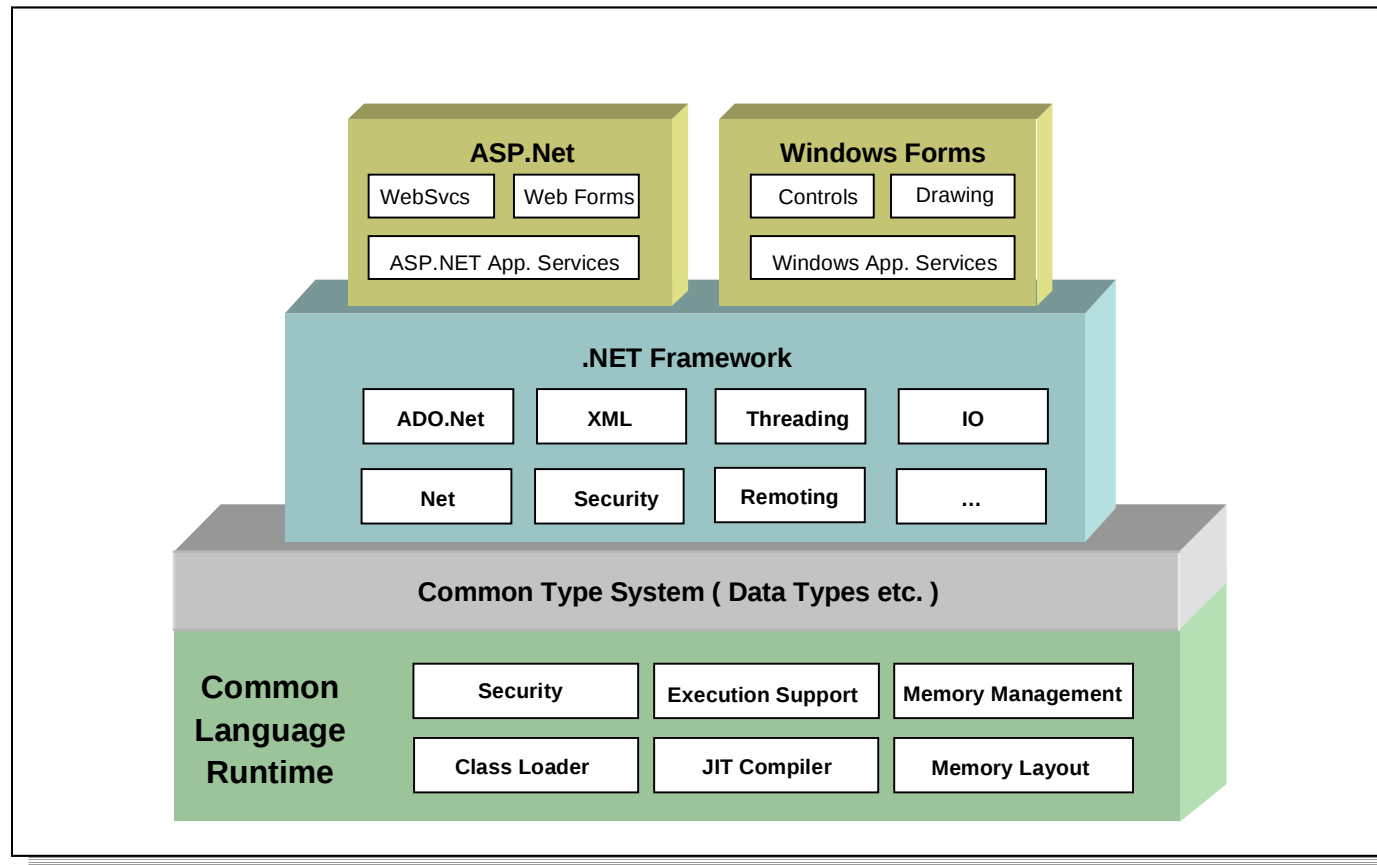
Teilaufgaben:

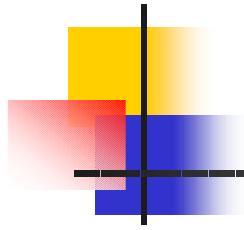
- Gegenüberstellung .NET vs. EJB-Komponenten
- UML Profile für .NET
- Abbildung des Profils auf C#
- Nachweis der Anwendbarkeit anhand eines „Onlineshop“



Was ist .NET ?

.net





.NET Eigenschaften



Programmiersprachen- und Plattformunabhängigkeit

- *durch Zwischensprache (MSIL – Microsoft Intermediate Language)*
- *und Common Type System*

.NET Attribute

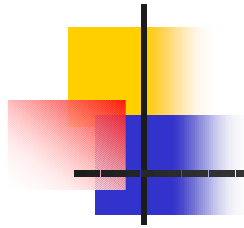
- *deklarative „Spracherweiterungen“, Code der Geschäftslogik bleibt übersichtlich*
- *Zuweisung an Assemblies, Klassen, Methoden, Feldern, Properties ...*
- *Verwendung in Tools (CASE-Tools, IDEs, Compiler) oder Frameworks (Persistenz~)*

Metadaten

- *Selbstbeschreibung der Komponente (der Klassen, Vererbung, Methoden ...)*
- *Identifikation der Komponente:*
 - *Versionsinformationen, Kulturzugehörigkeit*
 - *Verifizierung und Identifizierung durch private und öffentliche Schlüssel*

umfangreiche Sicherheitsmechanismen

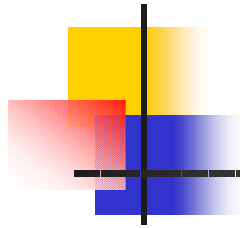
- *Code-Access-Security* ■ *Role-Based-Security* ■ *CryptographyServices*



Überblick



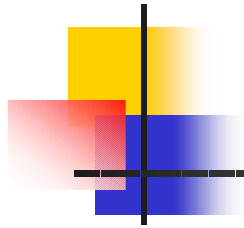
- Aufgabenstellung
- Was ist .NET ?
- *.NET vs. EJB Komponenten*
- MDA und EDOC
- Profileanforderung
- .NET Profile
- Aufgaben – Fragen – Ausblick



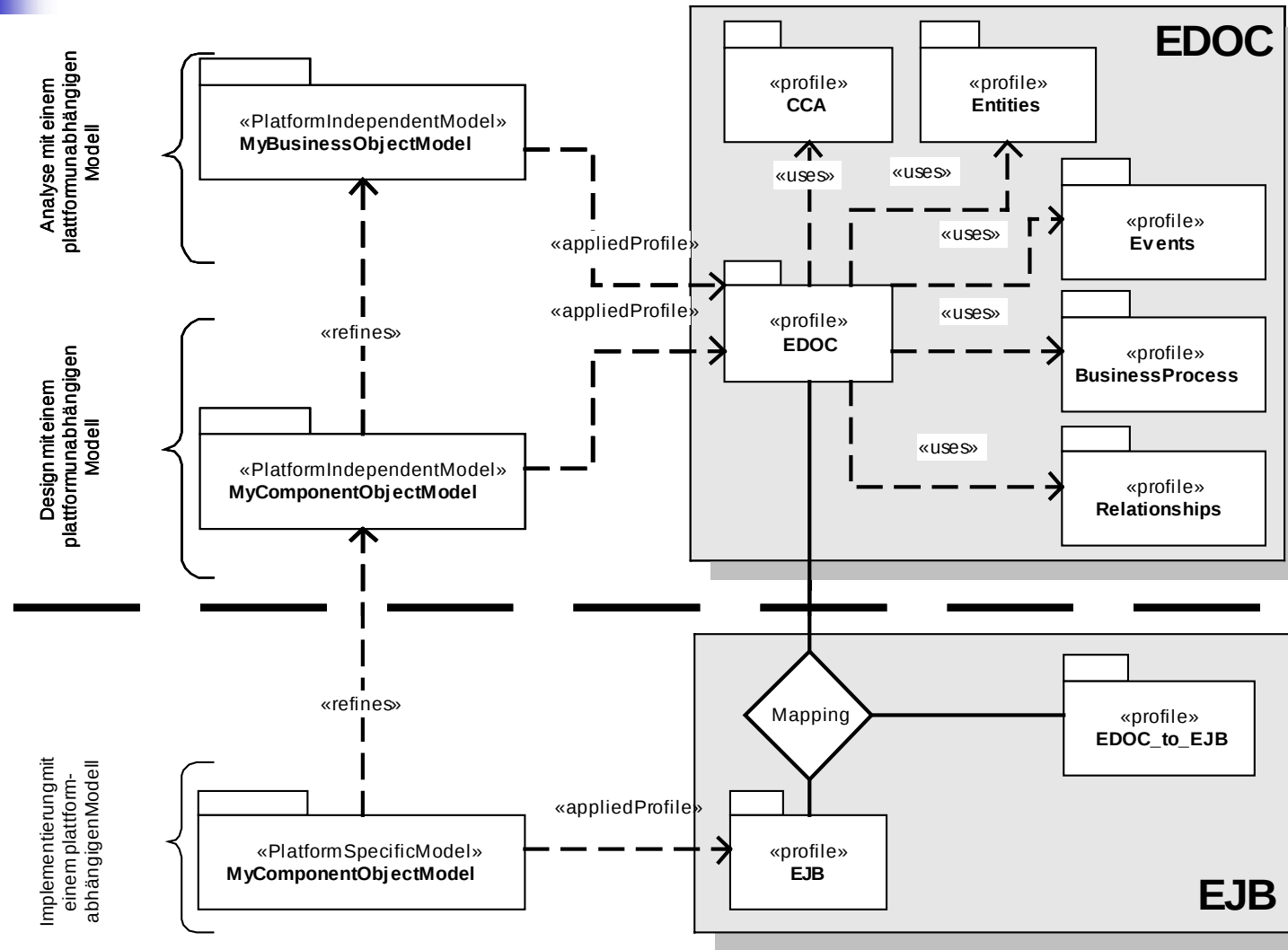
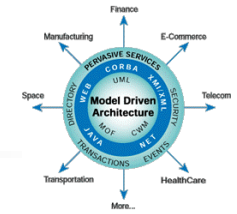
.NET vs. EJB

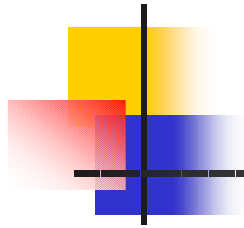


-
- **allgemeine Komponenten** (System.ComponentModel.IComponent)
 - *aus dem MS Component Object Model (COM) hervorgegangen*
 - *können logisch in Containern (ISite) enthalten sein*
 - *entsprechen den Java Beans*
-
- **Enterprise Komponenten** (System.EnterpriseServices.ServicedComponent)
 - *Container Dienste:*
 - ObjectPooling
 - ClientActivation
 - Transactions
 - Synchronization
 - SOAPServices
 - Queuing
 - *StateManagement (wie bei Session Beans) deklarativ anzugeben*
 - *entsprechen den Session Beans*
-
- **Daten Komponenten** (System.Data.DataSet)
 - *InMemory DB (DataRow, DataColumn, DataTable, DataView, PrimaryKey ...)*
 - *Verwendung von DataAdaptern für Persistenz*
 - *integrierter XML und XML-Schema Datenaustausch*
 - *mit Entity Beans vergleichbar*
-



MDA und EDOC



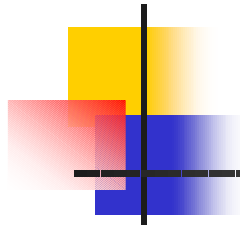


Profileanforderungen



- Allgemein (.NET-Profile):
 - *allgemeine Anwendbarkeit gewährleisten*
 - *nicht auf einen Anwendungsbereich beschränkt sein*
 - *Unterstützung des Softwareentwicklungsprozesses in mehreren Bereichen*

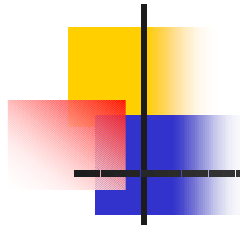
- Speziell (EDOC_to_dotNET):
 - *Möglichkeiten der .NET Plattform bestmöglich unterstützen*
 - *Programmiersprachenunabhängigkeit im Rahmen von .NET gewährleisten*
 - *Verwendung von Attributen*



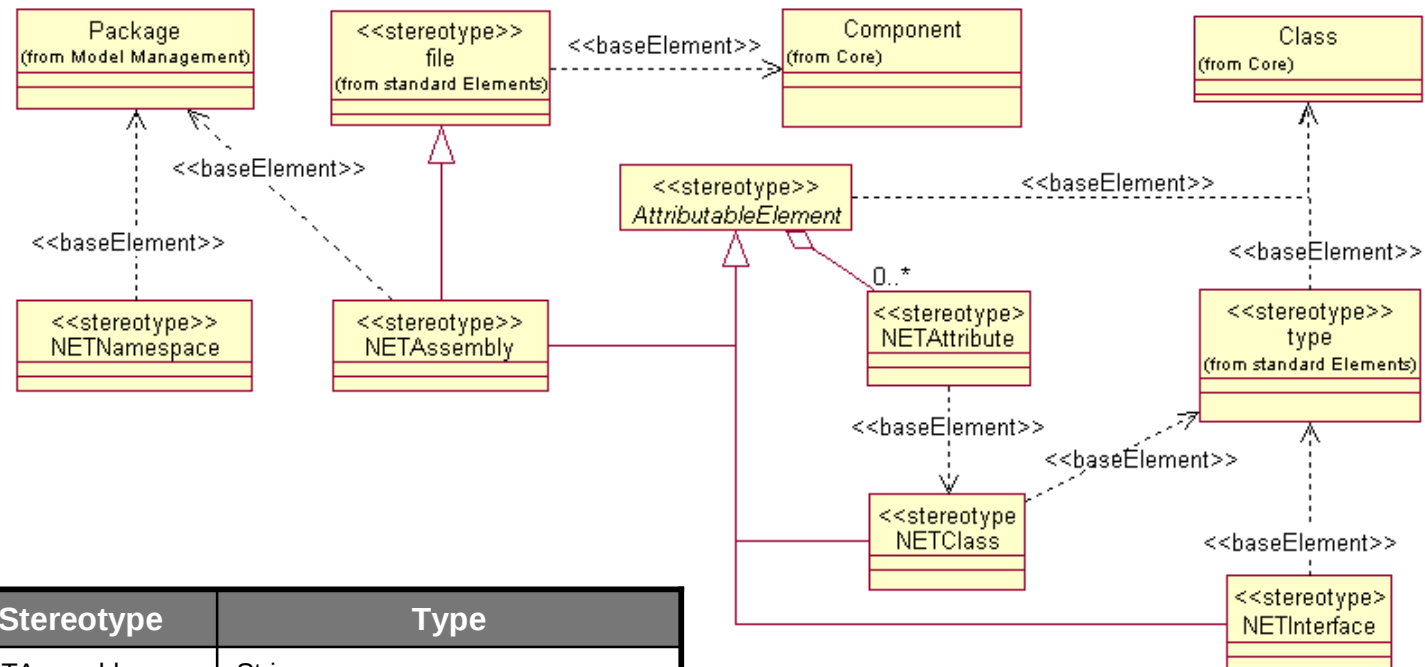
.NET Profile (allg.)



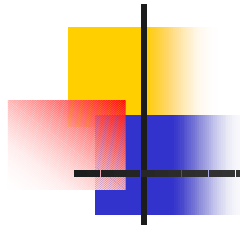
- „bottom up“ Ansatz von C# ausgehend
 - späteres „Refactoring“ zu .NET
- Anwendung von EDOC Profilen:
 - *Entities Profile:*
 - allgemeinen Komponenten (IComponent)
 - Daten Komponenten (DataSet)
 - *Business Process Profile:*
 - Enterprise Komponenten (ServicedComponent)
- Sourcecodeumsetzung:
 - *Stereotypes als .NET Attributklassen*
 - *Tagged Values als Membervariablen der Attributklassen*
 - *Constraints könnten mit Toolhilfe verifiziert werden*



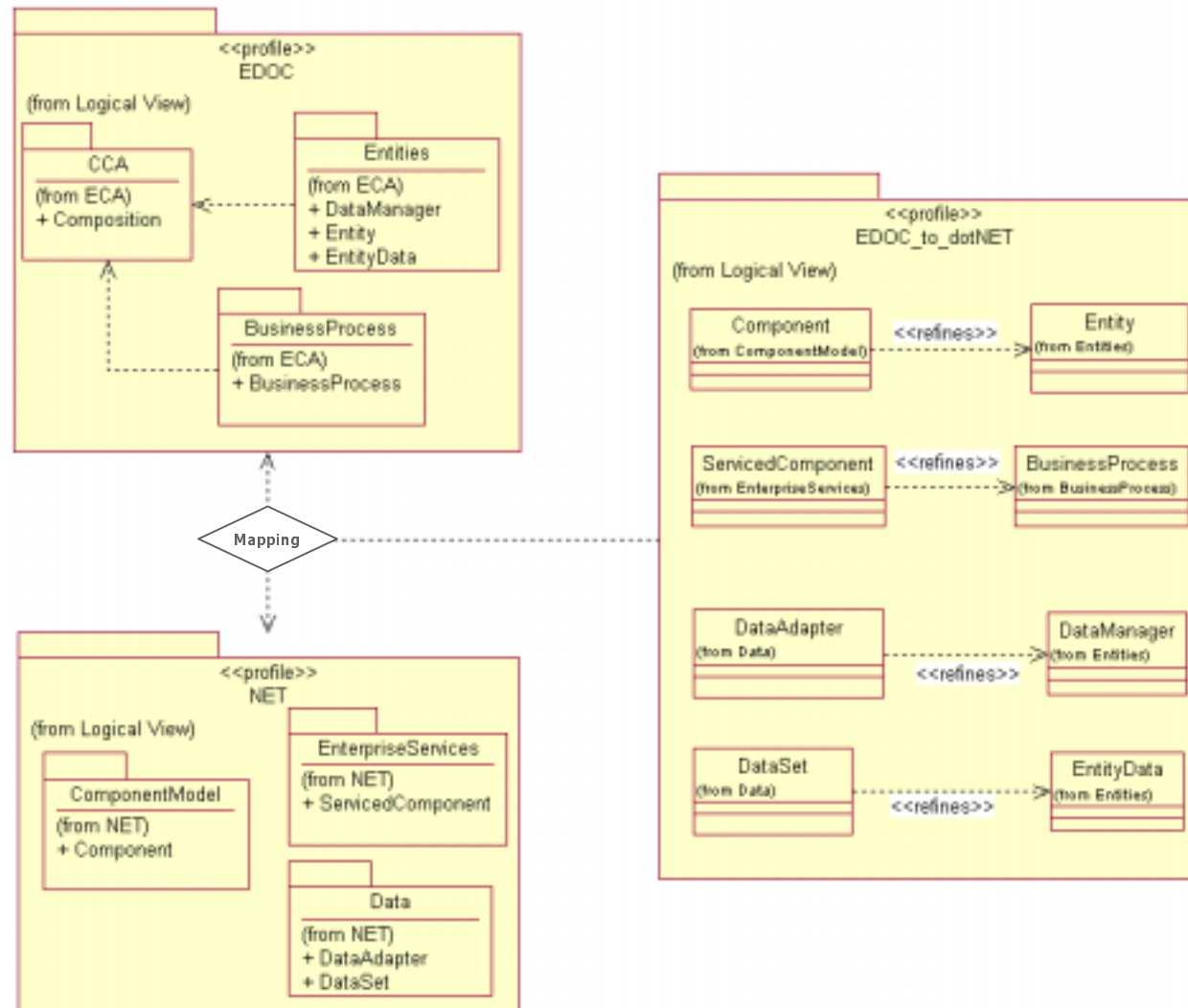
.NET Profile (CTS)

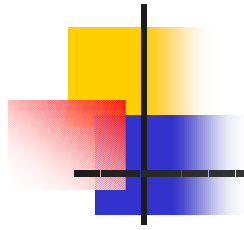


Tag	Stereotype	Type
Culture	NETAssembly	String
VersionInfo	NETAssembly	String
StrongName	NETAssembly	String
AttributUsage	NETAttributeClass	SetOf {All, Assembly, Interface, Class, Method, Property, Field...}
AllowMultiple	NETAttributeClass	Boolean



.NET Profile (EDOC)





Aufgaben – Fragen – Ausblick



offene Aufgaben:

- „Vervollständigung“ des .NET Profils
- Nachweis am Anwendungsbeispiel „Onlineshop“

offene Fragen:

- geeignete Transformationsregeln

Ausblick:

- .NET-Sourcecodeerzeugung (MSIL, C#, Eiffel.NET ...)
- Toolunterstützung bei der Anwendung von Profiles