

.NET Framework

Kann man es sich leisten
nicht umzusteigen?

von
Maik Igloffstein

Kurze Übersicht

- Geschichte von .Net
- .Net Einführung
- .Net Framework
- Die .Net Klassenbibliothek
- Die .Net Runtime Engine
- ...
- ...
- ...



„Information at your fingertips“
(Informationen per Knopfdruck)



„Empower people through great software, any time, any place and on any device“

(Befähigen Sie Menschen, Software zu jeder Zeit, an jedem Ort und an jedem Gerät zu entwickeln)



Geschichte von .Net

- Neue Version von COM (Component Object Models)
 - Neuer Name nicht gerechtfertigt
 - Neues Konzept: „Next Generation Windows Services“ (NGWS) erstellt
 - Umbenennung in .Net (Jahre 2000)
 - Version 1.0 Januar 2002
 - Aktuelle Version: 1.1
-
-

.Net Einführung

Aktueller Stand der Softwareentwicklung

- Neue Welt entsteht
 - Internet nicht mehr wegzudenken
 - Wünsche von Microsoft wurden erfüllt
 - Überall stehen PCs / „Apples“
 - Internet selbstverständlich (u. schnell)
 - Wissenszeitalter -> schneller Zugang zu Informationen
 - Windows noch führendes Betriebssystem
-
-

.Net Einführung

Aktuelle Probleme

- Keine Einheit bei der Programmierung
 - Massig Programmiersprachen
 - Module schwer zusammen fassbar
 - Kaum Portabilität (Betriebssystem abhängig)
 - Spezielle & unterschiedliche Bibliotheken
 - Unterschiedliche Optimierungen
 - Betriebssystemabhängigkeiten
 - Aufwendige Müllbeseitigung
-
-

.Net Einführung

Lösung?

- Einheitliche Bibliotheken
 - Großer Umfang an Bibliotheken und Problemlösungen
 - Einheitliche Verwendung von Bibliotheken
 - Einheitliche Hilfe
 - Frei wählbare Programmiersprache
 - Betriebssystemunabhängigkeit
-
-

.Net Framework

Was ist das .Net Framework

- Programmiermodell
 - Webdiensten
 - Applikationen
 - CLR (Common Language Runtime)
 - Verschieden Compiler
 - Klassenbibliothek
 - komplett objektorientierte
 - konsistente
 - einfache Klassenbibliothek
 - Kostenloses AddOn für Windows
 - ca 22 MB groß + 151 MB deutsches SDK
-
-

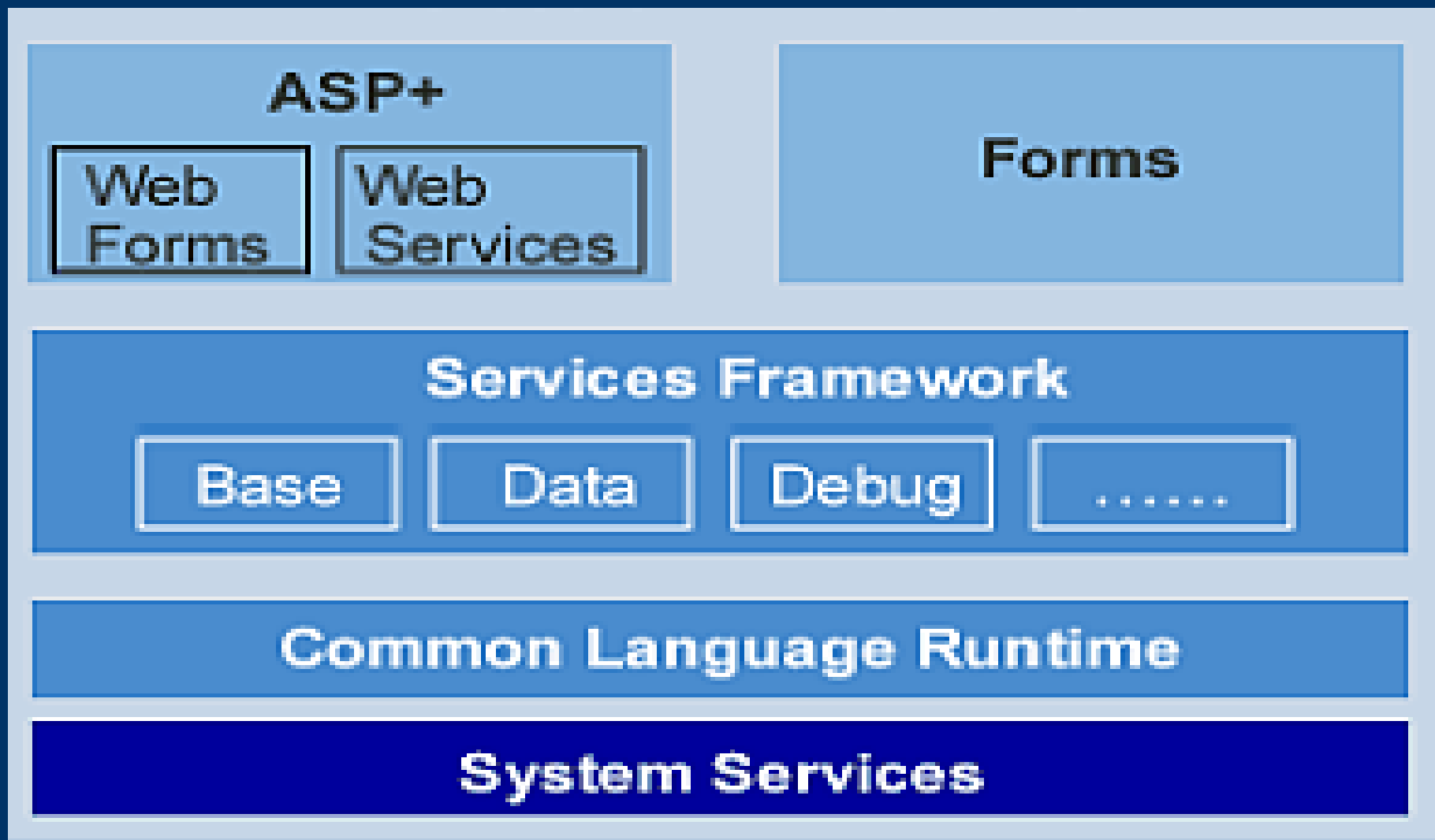
.Net Framework

3 Schichten Modell

- Runtime Engine
 - für Threadmanagement (ähnlich JVM)
 - Garbage Collection
- Allgemeine Klassenbibliothek (API)
- ASP+ (Erweiterung von ASP)
- Forms für Anwendungen



.Net Framework



Die .Net Klassenbibliothek

Allgemeines
.Net Framework Class Library (FCL)



Die .Net Klassenbibliothek

Inhalt der Klassenbibliothek

- 2246 öffentliche Klassen
- 4500 Klassen (insgesamt (optionale AddOns))
- Formulare
- Steuerelemente
- Fertige Problemlösungen

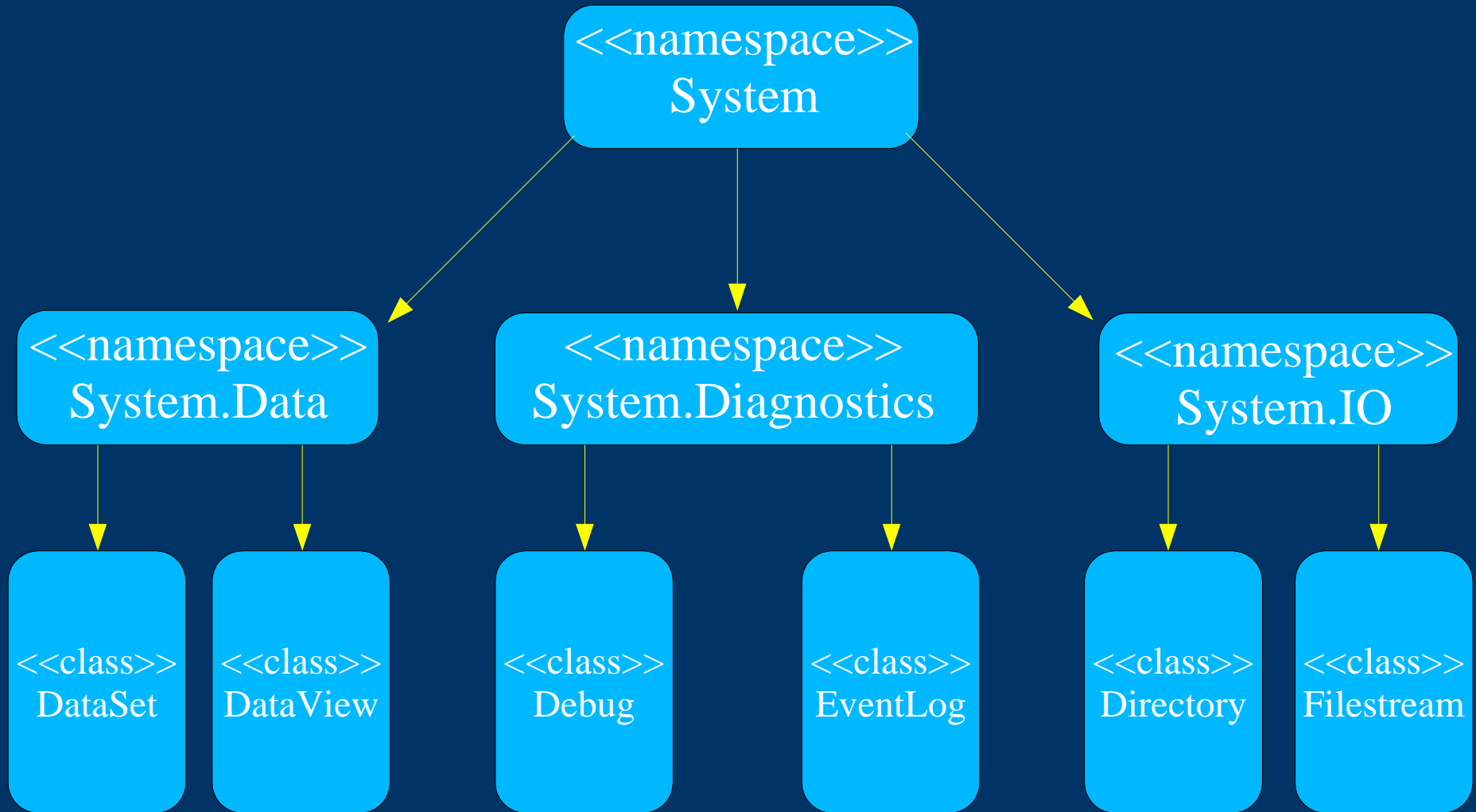


Die .Net Klassenbibliothek

Arten der Klassen in .Net

- Basisklassen
 - ADO.NET-Datenklassen
 - SQL-Klassen
 - XML-Klassen
 - ASP.NET-Klassen
 - Windows Forms-Klassen
-
-

Die .Net Klassenbibliothek



Die .Net Runtime Engine

don't panic



Die .Net Runtime Engine



Die .Net Runtime Engine

Inhalt:

- IL (Intermediate Language / Managed Code)
- CLR (Common Language Runtime)
- JiT (Just in Time Compiler)



Die .Net Runtime Engine

IL (Intermediate Language)

- Programme basieren nicht mehr auf der Windows API
- Quellcode → IL
- Nicht vom Betriebssystem ausführbar
- IL Programme sind „Betriebssystemunabhängig“



Die .Net Runtime Engine

CLR (Common Language Runtime)

- Laufzeitumgebung für alle .Net Programme
- Führt IL Code aus
- Bietet große Anzahl von Standarddatentypen
- CTS (Common Type System)
-



Die .Net Runtime Engine

JiT (Just in Time Compiler)

- Umwandlung von IL Code in Betriebssystem verständlichen Code
- Optimierung für verschiedene Prozessoren
-



- ASP.NET Entwicklerbuch
 - Plattformunabhängigkeit :) S26
 - Bei Seite 12 Klassenbibliothek
 - SAGEN: Aufbau in DLLs (Managed Code)
 - Code Sicherheit bei CLR

