

.NET Programmers and Architects

Here are your options

Oliver Sturm

@olivers

oliver@oliversturm.com



Oliver Sturm

- Training Director at DevExpress
- Consultant, trainer, author, software architect and developer for over 25 years
- Microsoft C# MVP
- Contact: oliver@oliversturm.com

Agenda

- Idea: talk about technology
- Client applications
- Data access
- Web applications
- Services
- Mobile
- Cloud
- Open Source

Client applications

- What's a client application today?
- What was it yesterday?

Client style UI

- WinForms
- WPF
- Silverlight
- WinRT
- ... XAML based

Client style UI?

- HTML

Reasoning about client applications

- For: Direkter Zugriff auf lokale Hardware (keine Einschränkungen durch virtuelle Umgebung), muss kein JavaScript benutzen, Einfache Verwendung von Offline-Daten, Weniger Zielplattformen -> vereinfachtes Testen
- Against: Deployment
- Problem: Lizenzierung

Client UI Application Patterns

- **MVVM**

Reasoning about client UI application patterns

- For: Bessere Struktur, verminderter Pflegeaufwand, Testen, Wiederverwendung, Einarbeitung neuer Mitarbeiter ist einfacher, (Automatische?) Prüfung von Regeln wird möglich
- Against: Patterns sind schwerer zu verstehen als keine Patterns, Disziplin zur Einhaltung ist gefragt
- Diskutierbar: “Ein schlecht implementiertes Pattern ist immer noch besser als gut implementiertes Chaos”

Storing Data

- Relational databases
- NoSQL options
 - Key/value and column family stores
 - Document
 - Data analytics (e.g. MapReduce)
- Reasoning
 - For RDBMS - strukturierte Datenhaltung -> Vermeidung von Redundanz
 - For NoSQL - einfache Skalierung, flexible Datenaufteilung, weniger Struktur

Accessing Stored Data

- Direct access - ADO.NET?
- NoSQL APIs?

ORM

- Entity Framework
- Third parties
- Top Down vs Bottom Up
- DB independence

- Command Query Responsibility Segregation
- Query and Command Models
- Consistency often not Strong - Eventual consistency?
- Back to data access basics?
- Conflicts with ORM?

Reasoning About Data Access

- For direct access: Schnellster Zugriff, flexibelster Zugriff, kuerzester Code, Direktzugriff auf ADO.NET-Funktionalitaet
- For ORM: Objektorientierte Programmierung und Datenpersistenz, geteiltes Modell fuer Lesen und Schreiben
- For CQRS: Aufspaltung von Lese- und Schreibmodellen, Unterstuetzung verschiedener Laufzeitanforderungen an beide Modelle, getrennte Skalierbarkeit

Web Applications

- What is a web application?

Server-side rendering on ASP.NET

- WebForms
- MVC
- Reasoning
 - For WebForms: viele automatische Black Magic-Mechanismen
 - For MVC: Web-Plattform der Zukunft bei MS, Multiplattform-Zukunft, offiziell unterstützt
 - For server-side rendering: Selektive Inhalte, Client “sieht” nichts, was er nicht sehen soll
 - Against server-side rendering: Overhead

HTML “direct”

- ... with little or no server rendering
- Can be delivered by a server
- Reasoning
 - Dies ist eine Client-Anwendung, wie zuvor besprochen, potentiell mit Zusaetzen durch Server-Rendering

Services

- Part of most architectural concepts
- SOA?
- “Real-time web?” SignalR? socket.io?
- Microservices

Web Services

- ASMX? WSE?
- WCF
 - Configurability - decision late binding
 - Complexity
- Reasoning
 - For: Unterstützung von Standards (SOAP, WSDL, WS-xxx)
 - Against: Komplexität und Geschlossenheit vieler WCF-Lösungen

Web API

- Based on ASP.NET MVC infrastructure
- JSON/XML content negotiation
- REST
- Reasoning
 - For: Microsoft's Serviceplattform der Zukunft, offene Dienste, Einfachheit der Implementation
 - Against: Rueckweg zu komplexen Protokollen und Standards auf Web API nicht unterstuetzt

Mobile

- Mobile support as a conceptual module
- Strategic platform?

“Native” Mobile

- Windows Phone SDK
 - More XAML!
- iOS SDK
- Android SDK
- Reasoning
 - For: Performance, Zugriff auf alle Features des Geraets
 - Against: Investitionen pro Plattform

Mobile .NET

- Xamarin
- Reasoning:
 - For: Code-Investitionen koennen gesichert werden, Know-How wiederverwendet
 - Against: Overhead zur Ansteuerung verschiedener Plattformen (UI), alternativ Xamarin Forms

HTML/Hybrid

- HTML (5), JavaScript, CSS
- PhoneGap/Cordova, CrossWalk, nw.js, ...
- Cross-platform
- Reasoning
 - For: Plattformunabhaengigkeit,
 - Against: Performance, “native feel” ist noch nicht immer gegeben

Cloud

- Deployment option
- Managed infrastructure

Cloud functionality

- Supplied services, vertical features
- Base platform functionality
 - Dynamic scalability
 - SLA

Legal considerations

- Locations
- Industry/governmental requirements

Cloud options

- Azure
- Amazon Web Services
- Google App Engine, Compute Engine, Container Engine, ...
- Third parties?

Cloud Reasoning

- For cloud:
- Against cloud:
- For/against specific platforms:

Open Source

- Microsoft does it
- It's a give and take

Open Source Reasoning

- For Open Source:
- Against Open Source:
-

Thank You

Please feel free to contact me about the
content anytime.

oliver@oliversturm.com