**ArcGIS Runtime SDK for WPF und MVVM**

**MVVM**

Das Model-View-ViewModel-Pattern (MVVM) ist ein Vertreter einer ganzen Familie von Pattern, deren vorrangige Aufgabe es ist, die Daten von deren Darstellung zu trennen. Die Ursprünge dieses Pattern liegen schon 30 Jahre zurück im Smalltalk als Model-View-Controller (MVC) und fand über die Weiterentwicklung Model-View-Presenter (MVP) zur aktuellen Ausprägung. Dabei bezeichnet MVVM eine spezielle Form des Presentation-Pattern durch Microsoft für deren neueren Technologien WPF und Silverlight. Die Idee dahinter ist die Kapselung der Geschäftslogik in einem eigenen Konzept (Model), deren Daten mittels eines Vermittlers (ViewModel) im Frontend (View) dargestellt werden. Diese Art der Aufteilung adressiert gleich mehrere Aufgaben

* **Fokussierung der technischen Verantwortlichkeiten** - Das Model kümmert sich ausschließlich um das Management (Lesen, Bearbeiten, Speichern) von Daten, das ViewModel dient ausschließlich zur visuellen Aufbereitung von Daten und steuert die Logik der UI. Der View ist lediglich zur Anzeige und Manipulation von Daten verantwortlich. Damit wird auch das Architekturpattern SOC (separation of concerns) umgesetzt.
* **Fokussierung der organisatorischen Verantwortlichkeiten** - Während der Entwickler die eigentliche Programmlogik programmiert, kann das Frontend völlig losgelöst von einem UI-Experten gestaltet werden.
* **Flexibilität** - Eine derartige Architektur erlaubt das schnelle Auswechseln und Anpassen von Komponenten. Was heute noch auf dem Desktop ist, kann morgen einfach ins Web portiert werden. Dazu müssen entsprechend nur der View mit seinem zugehörigen ViewModel implementiert werden. Die meist aufwändige Businesslogik bleibt unangetastet.
* **Testbarkeit** - Durch einen fehlenden UI-Bezug ist die Geschäftslogik relativ einfach Unit-testbar.

Das Besondere an MVVM ist das sogenannte DataBinding. Dabei werden die UI-Elemente vom Designer einfach auf entsprechende Anknüpfpunkte des ViewModels geklemmt. Die Technologie sorgt dann völlig eigenständig dafür, dass die Daten korrekt im Frontend angezeigt oder in der anderen Richtung im ViewModel aktualisiert werden. So kann beispielsweise ein Integerwert mit einem Slider, Spinedit oder Eingabefeld verbunden werden. Der Entwickler muss sich keine Gedanken mehr machen, den Slider oder das Spinedit korrekt zu setzten - das passiert automatisch im Hintergrund durch das Binding.  
  
~~MVVM ist universell gegenüber der Plattform und neben den genannten Plattformen und Technologien gibt es unzählige weitere Implementierungen (z.B. KnockoutJS für JavaScript und .~~

**ArcGIS Runtime SDK for WPF**

ArcGIS Runtime for Desktops ist ein neues Produkt von Esri für die Entwicklung von leichtgewichtigen und funktional reichhaltigen GIS-Anwendungen für verschiedene Desktop-Plattformen (siehe Artikel in arcAKTUELL 4/2011, 2/2012 und diese Ausgabe). Dabei kann der Entwickler zwischen verschiedenen nativen SDKs wählen. Diese SDKs sind im Rahmen einer Subskription des Esri Developer Network (EDN) erhältlich. Eines davon ist das ArcGIS Runtime SDK for WPF. Bei der Erstellung von Applikationen auf dieser Basis sollte natürlich auch das bewährte MVVM-Pattern angewendet werden. Ein möglicher Ansatz dazu wird im Folgenden beschrieben.

**Werkzeuge**

**Entwicklungsumgebungen**

Neben Visual Studio bietet Microsoft mit Expression Blend ein ausgezeichnetes Programm zum Designen von Oberflächen an. Die anvisierte Zielgruppe wird schon auf dem ersten Blick klar: Designer. Visuell orientiert sich Expression Blend an den Adobe-Klassikern - ein dunkles UI um die Augen nicht abzulenken und unzählige kleine Panels zur Parametrierung. Aber nicht nur dem Designer wird die Arbeit erleichtert, auch dem Entwickler kann Expression Blend eine gute Hilfe sein. Unter anderem lässt sich DataBinding über diverse Menüs sehr komfortabel umsetzen.

**MVVM Light Toolkit**

Darüber hinaus gibt es für die Entwicklung mit MVVM unzählige kleine und große Helferlein. Ein elegantes und gerne von uns verwendetes Framework ist das kostenfreie MVVM Light Toolkit von Galasoft. Die Vorteile sind:

* Verwendbarkeit für WPF- und Silverlight-Projekte
* Projekt- und Item-Templates
* Viele Nützliche Hilfen für DataBinding (Blendability)
* Messaging-Unterstützung
* Umfangreiche Dokumentation

Mittels NuGet lässt sich das Toolkit einfach in eigene Projekte einbinden. Dabei wird neben den Einbetten der eigentlichen Bibliotheken auch der DevTree für die Verwendung angepasst.

**All together - unser Demo**

Motivation, Beschreibung unseres Projektes