



Angriff
der
KILLERBUGS

**Angriff
der**

KILLERBUGS



Das Problem: Die vier apokalyptischen Reiter betreten die Bühne

interne Rahmenbedingungen



- Kenntnisse der verfügbaren Technologien
- verfügbare Technik
- weitere Projekte
- Technik-Trouble



Implementierung und Deployment



- Suche nach geeignetem Entwicklungswerkzeug
- Aneignung der spezifischen Kenntnisse
- Prototyping
- Prototyp wegwerfen
- Neuen Prototyp erstellen
- u.s.w.u.s.f.
- Auslieferung über PlayStore

Kundenanforderungen

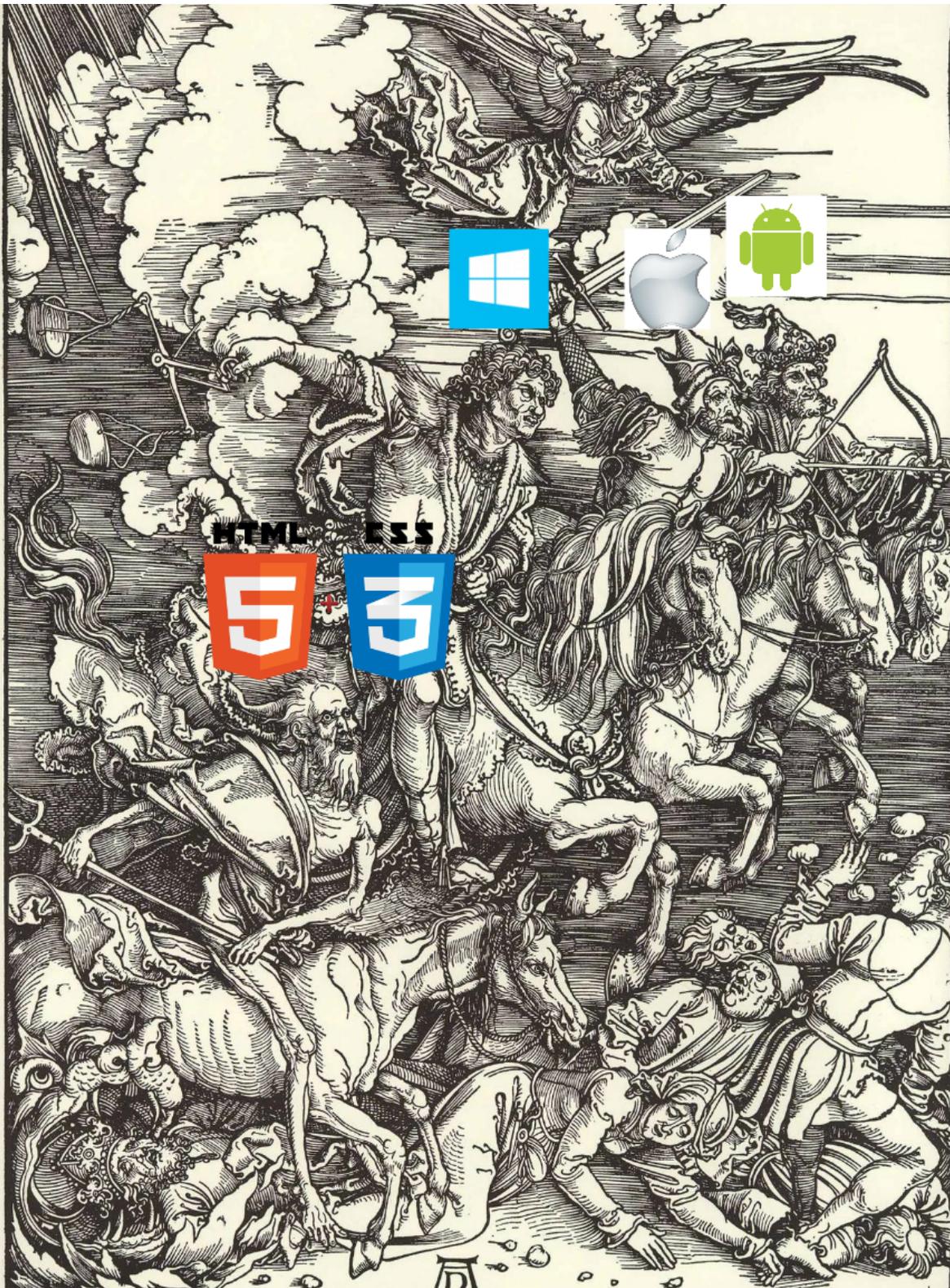


- einfache Nutzeroberfläche
- alle nur denkbaren Funktionen
- intuitive Bedienbarkeit
- Fachexpertenunterstützung
- Offline-Fähigkeiten
- Online-Fähigkeiten
- spritzwassergeschützte GUI
- Toleranz gegenüber grob fahrlässiger Bedienung

äußere Rahmenbedingungen



- verfügbare Frameworks
- Terminvorgaben
- wankelmütiger Kunde
- Kundenmotivation



HTML5



Kundenanforderungen



- einfache Nutzeroberfläche
- alle nur denkbaren Funktionen
- intuitive Bedienbarkeit
- Fachexpertenunterstützung
- Offline-Fähigkeiten
- Online-Fähigkeiten
- spritzwassergeschützte GUI
- Toleranz gegenüber grob fahrlässiger Bedienung

äußere Rahmenbedingungen



- verfügbare Frameworks
- Terminvorgaben
- wankelmütiger Kunde
- Kundenmotivation

interne Rahmenbedingungen



- Kenntnisse der verfügbaren Technologien
- verfügbare Technik
- weitere Projekte
- Technik-Trouble

Implementierung und Deployment



- Suche nach geeignetem Entwicklungswerkzeug
- Aneignung der spezifischen Kenntnisse
- Prototyping
- Prototyp wegwerfen
- Neuen Prototyp erstellen
- u.s.w.u.s.f.
- Auslieferung über PlayStore

Technische Ansätze bisher



- Nummer sicher: Android-Entwicklung mit C#
- Xamarin for Visual Studio
- Problem: Verfügbarkeit von Dritthersteller-Komponenten
- Abhängigkeit von Xamarin



Technische Grundlagen

- dynamische Anpassung der Oberfläche ohne Quellcodeanpassung
- Datenerfassung offline möglich
- Datenübertragung live in zentrale Kunden-DB möglich
- Kartenbasis (noch) GoogleMaps

PRODUCT DEMO
NEXT EXIT ↗



Android fights back!



Android fights back!





Stadler
elisker vithy

len

L.

A

dbo.REF_SchaderregerAttribute

Spalten

- AttributID (PS, uniqueidentifier, Nicht NULL)
- SchaderregerID (FS, uniqueidentifier, NULL)
- Attributname (nvarchar(255), NULL)
- ControlType (nvarchar(255), NULL)
- Sort (int, NULL)

dbo.REF_SchaderregerMonitoring

Spalten

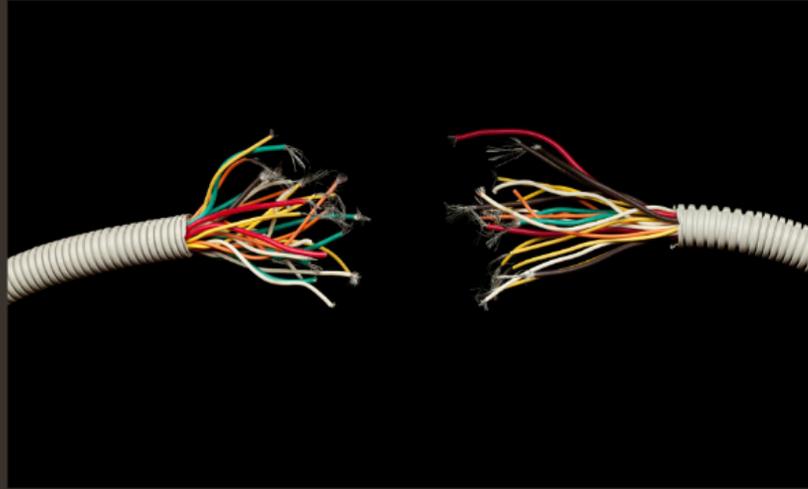
- ID (PS, uniqueidentifier, Nicht NULL)
- Name (nvarchar(255), NULL)
- Sort (int, Nicht NULL)

dbo.REF_SchaderregerAttributWerte

Spalten

- ID (PS, int, Nicht NULL)
- AttributID (FS, uniqueidentifier, Nicht NULL)
- Wert (nvarchar(255), NULL)
- Sort (int, NULL)

Offline-Datenerfassung

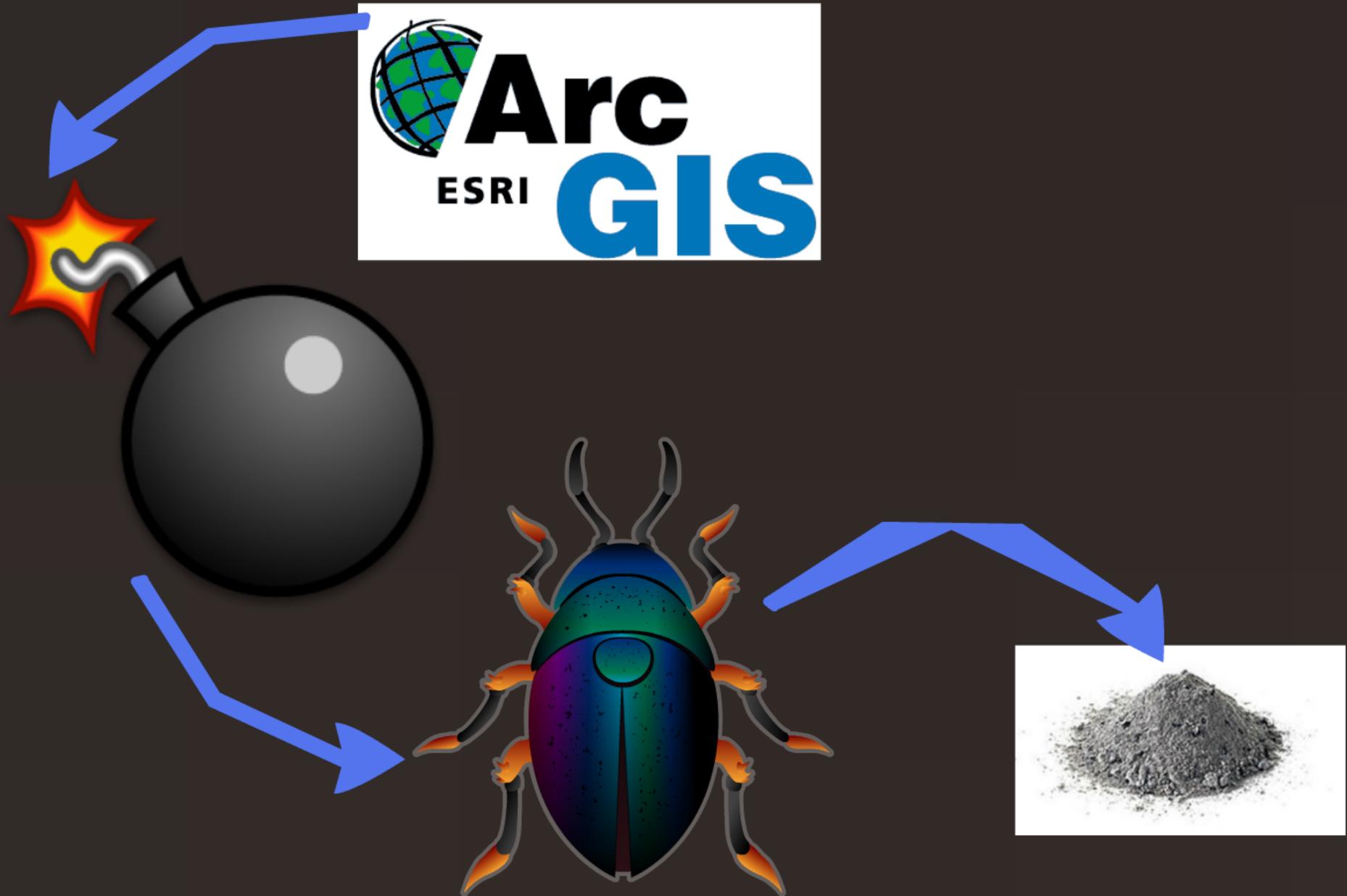


- lokale Zwischenspeicherung XML-filebasiert
- Übertragung der Daten per FTP wenn Datenverbindung verfügbar ist
- FTP-Server mit dienstbasierter Prüfung auf neue Daten

Datenfluss



Datenfluss (2)



GIS-Kopplung



**Fach-IS
Schlüssel-
übergabe**

**Wertanalyse
(XML-Webservice)**

**Zoom und
Selektion
(ArcGIS for
Server)**



**Schlüsselübergabe
via XML-Webservice**

Selektion im Fach-IS

Aufbereitung mobile Daten

1



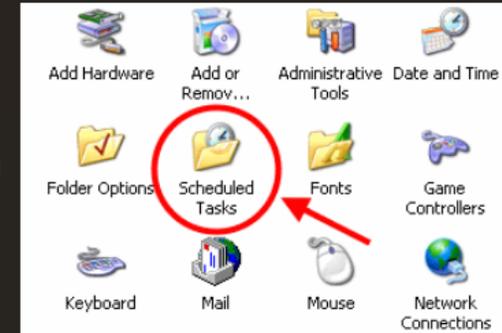
XML-File

2



FTP-Transfer

3



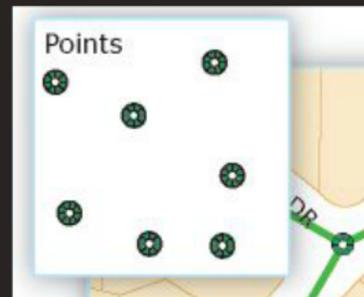
Task-Ausführung

4



Python-Script

5



Featureclass
Import

6



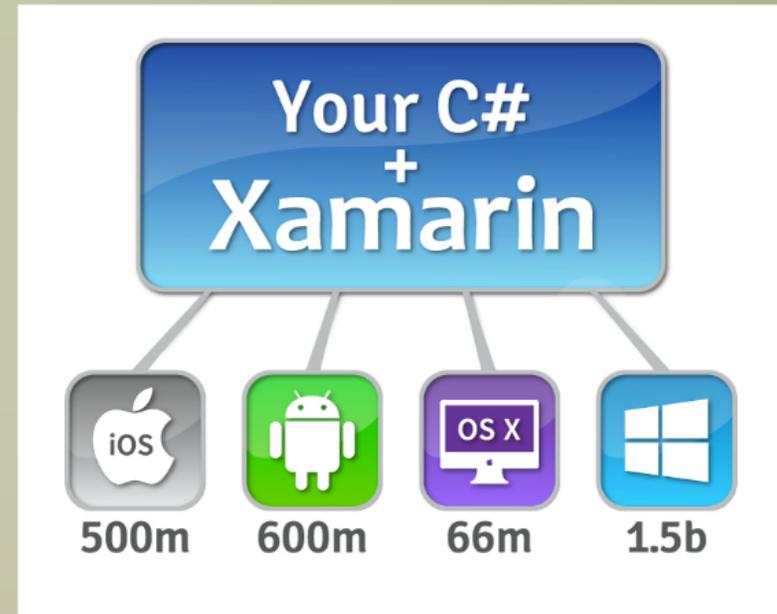
GIS-Darstellung

Blick in die Zukunft



- mehr Unabhängigkeit von Drittanbietern
- mehr Funktionen in der Karte
- Nutzung der App als Template für neue Anwendungsfälle

Künftige Lösungsansätze: Entwicklungsplattform



ArcGIS API for JavaScript

AppStudio for
ArcGIS



Der native Ansatz für Android: Java



- Voller Funktionsumfang
- geringe Abhängigkeiten von Drittanbietern
- UI 1:1 wie Zielplattform
- gute Performance
- schnelle Anpassung bei Änderungen der Zielplattform
- keine Zusatzkosten für IDE

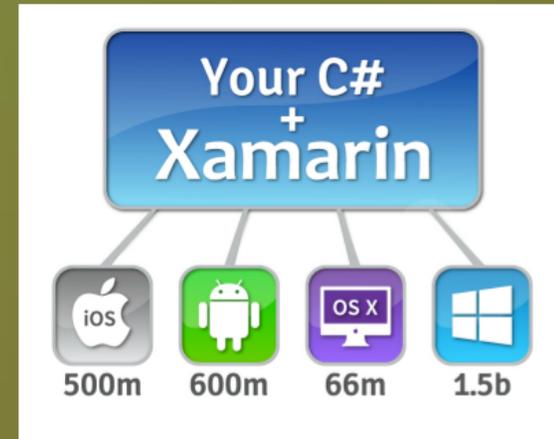


- Volle Abhängigkeit von Zielplattform
- Portierbarkeit der Lösung schwierig



Cross-Plattform für C# -Jünger

- Portierung für verschiedene Zielplattformen unterstützt
- Integration in VisualStudio
- Nachnutzung bestehenden Quellcodes möglich
- gute Performance



- Abhängigkeit von Xamarin bei Aktualisierungen der Zielplattform
- Nutzung von Drittanbieter-Komponenten tlw. problematisch
- Kosten für Lizenz und Pflege
- Anpassungen für Zielplattformen trotzdem tlw. notwendig



Der Web-Ansatz



- Portierbarkeit für verschiedene Zielplattformen relativ einfach
- keine Kosten für IDE
- viele Plugins verfügbar
- umfangreiche Unterstützung durch Esri
- große Community



- suboptimale Performance
- Einschränkungen bei Offline-Szenarien
- Einschränkungen in der GIS-Funktionalität

Das Rundum-Sorglos-Paket



- Portierbarkeit für verschiedene Zielplattformen gegeben
 - einfache Bedienung
 - Templates für "schnelle App für zwischendurch"
 - Build-Prozess automatisiert durch Esri
-
- kein voller Funktionsumfang
 - Abstriche bei nativer UI-Abbildung
 - Abhängigkeit von Drittanbietern



**AppStudio for
ArcGIS**



Fazit?

es kommt immer darauf an...

Eigene
Kenntnisse

Anforderungen
der
[preliminären]
Kunden

Budget

Eigene Kenntnisse

Anforderungen der (potenziellen) Kunden

Budget



WAKE

UP

WE'RE

ALMOST

DONE