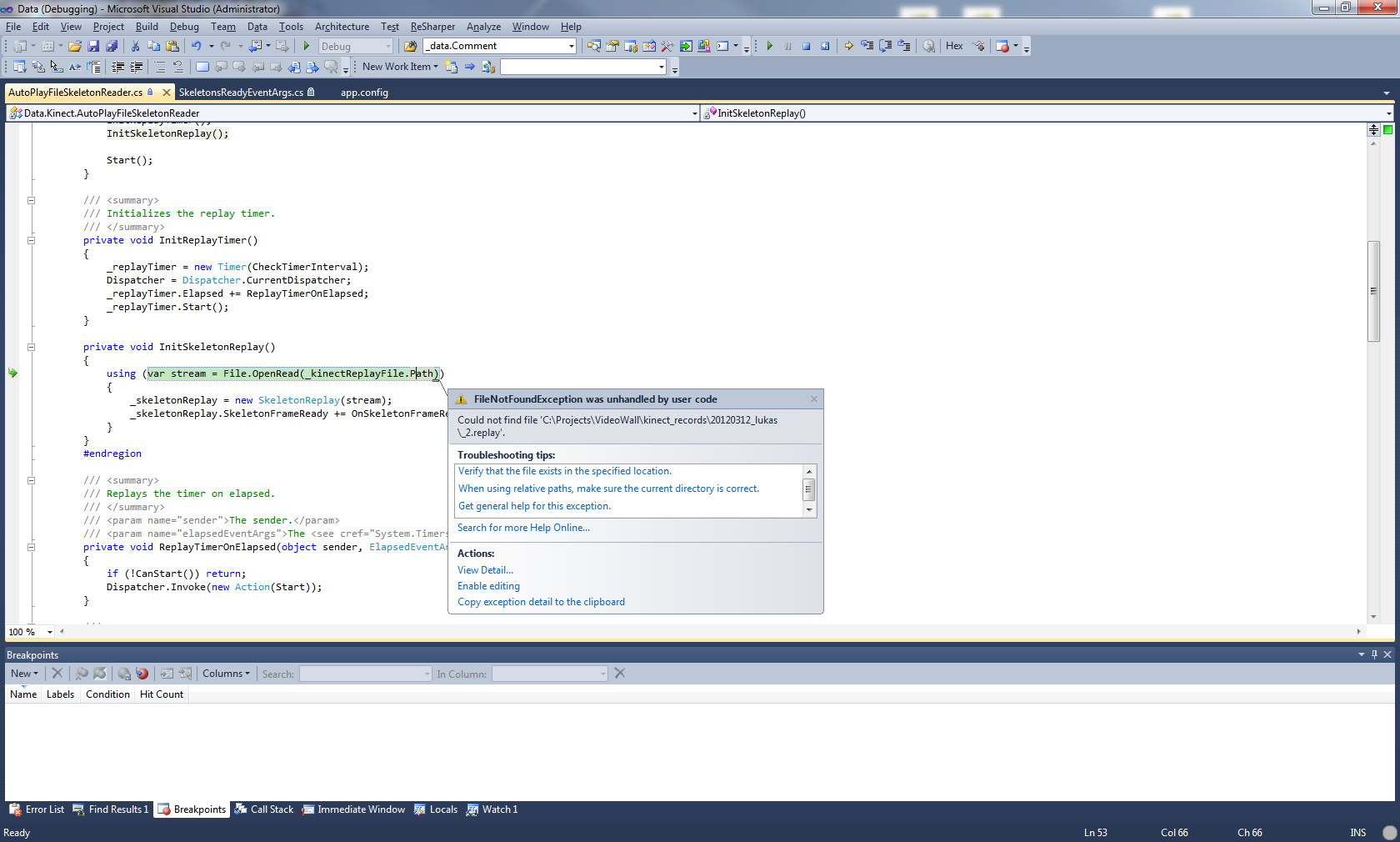
# Code Review vom 03.05.2012

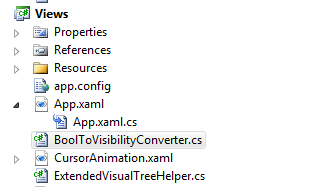
## Exception bei Startup



Could not find file 'C:\\Projects\\VideoWall\\kinect\_records\\20120312\_lukas\\\_2.replay'.":"C:\\Projects\\VideoWall\\kinect\_records\\20120312\_lukas\\\_2.replay

* ZIP-Extract notwendig (mit Lukas besprochen?), ist dieses Vorgehen auch in der Doku beschrieben?
  + Problem: Nicht eingeweihte bzw. neu zum Projekt stossende Programmierer werden mit Garantie in das Problem laufen.
  + Daher: Bitte Deployment anders lösen (z.B. mit Resources), der Code darf (auch beim Entwicklungszeitpunkt) keine Pfade enthalten (DITO in LunchMenuReader).

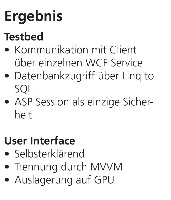
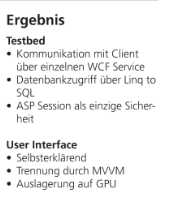
## Organisation der Namspaces



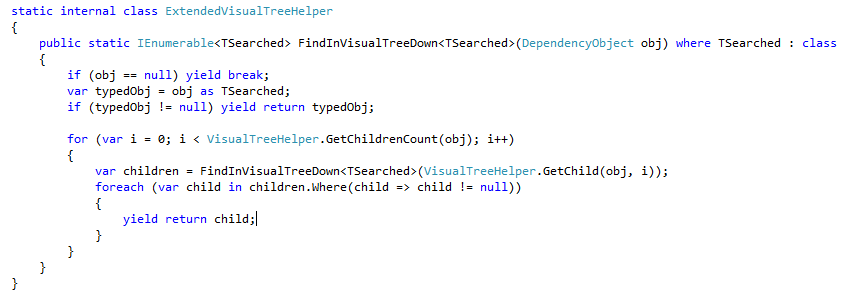
Converter(s) bitte in eigenes Package / Ordner.

## Image

Tipp: RenderOptions.BitmapScalingMode="HighQuality"  
Sieht besser aus und es Laggt weniger (!)



## Scope & Utils



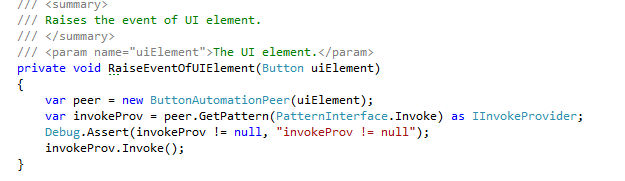
static internal class / public static method -> Semantik bewusst eingesetzt?

Util’s bitte in eigenes Package, evtl. sogar in Common GUI-Library?

Kommentare im Beispiel fehlen.

## Extension Methods

Bitte Information Expert einhalten: Das folgende Code-Stück wäre ein gutes Beispiel für eine Extension-Method auf Button->Click() :



## MVVM

UI Elemente im ViewModel ist unschön. Zerstört die Testbarkeit.

## Logger

Logger (log4net) fehlt (noch)? Console.WriteLine() im Code sehr unschön, sobald der Code auf Build Servern ausgeführt wird.

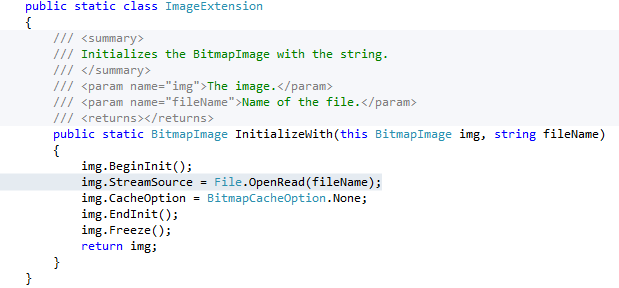


## Memory-Problem

OutOfMemoryProblem Exception ist beim Rumspielen aufgetreten. Vorsicht: Mengengerüst besagt, dass viel mehr Daten (Posters) geladen werden müssen. Ist dies in der Risiko-Analyse beschrieben?

Die Posters werden auf einem Intel Core i5 660 / 4GB Ram nur mit „LAG“ geladen. Dies könnte ebenfalls ein erweitertes Problem darstellen, falls die Daten übers Netzwerk geladen werden (dann z.B. lokal kopieren)?

An diversen Stellen werden die Bilder mit einem File-Stream geöffnet:



Wer schliesst diesen Stream? Verursacht dies ein Memory-Leak? Before/After Pattern nicht angewendet/eingehalten?  
Unschön: Konzept des Memory-Managmenents (Caller / Callee konsumiert Stream) nicht in der Code-Dokumentation der Methode festgehalten. Dies stellt ein potentielles Problem für Memory-Leaks dar.

Besser direkt als Initialisierungs-Methode: LunchMenu::CreateFrom(string fileName) ?

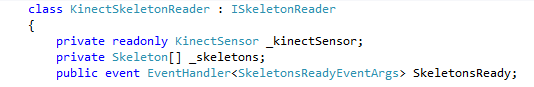


## Domain Objekte / Business Layer

Wo befindet sich das ServiceModel (bzw. die Domain Objekte)?  
Nun greift das ViewModel direkt auf die DatenServices (DL) zu. Betreffend Nomenklatur ist dies irreführend oder gewollt?

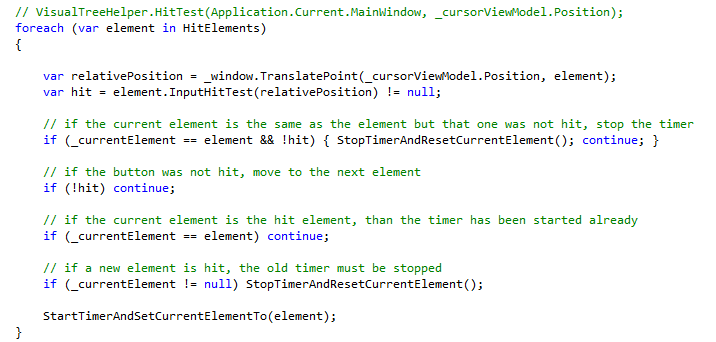
## Dispose() Pattern

Dieses wurde an diversen Stellen nicht vollständig implementiert, z.B.:



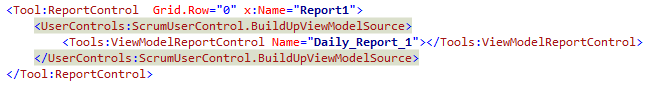
ISkeletonReader ist zwar IDisposable, allerdings fehlt die komplette Implementation des Dispose-Patterns (siehe <http://msdn.microsoft.com/en-us/library/b1yfkh5e(v=vs.71).aspx> )

## WPF



Weshalb ein Loop und nicht die HitTest Methode (1. Linie)?

ViewModel Verschachtelung, MenuViewModel zu mächtig bzw. weiss mehr als MainWindowViewModel? Warum so gelöst?



PropertyChange vs Changes mit Events.

Warum Thread based Timer und nicht DispatcherTimer?

## Debugging

Wozu dieser Assert hier? Muss die Funktionalität des .NET Frameworks nochmals überprüft werden?



## Dokumentation

Entwurf.docx -> Converter -> Converter (beispielsweise zur Konvertierung von bool zu visbility)  
-> Ist im View?

Technische Dokumentation nicht bis zum Ende aufschieben. Auch Erkenntnisse von den Prototypen?

## Generelle Anmerkungen zum Programm-Code

Regionen abgeschafft / nicht geplant?

Reihenfolge und Strukturierung von Properties / Methoden / Construktoren ist inkonsistent.

Code-Guidelines vorhanden & eingehalten?

Keine PreConditions eingesetzt (haben wir doch besprochen)?