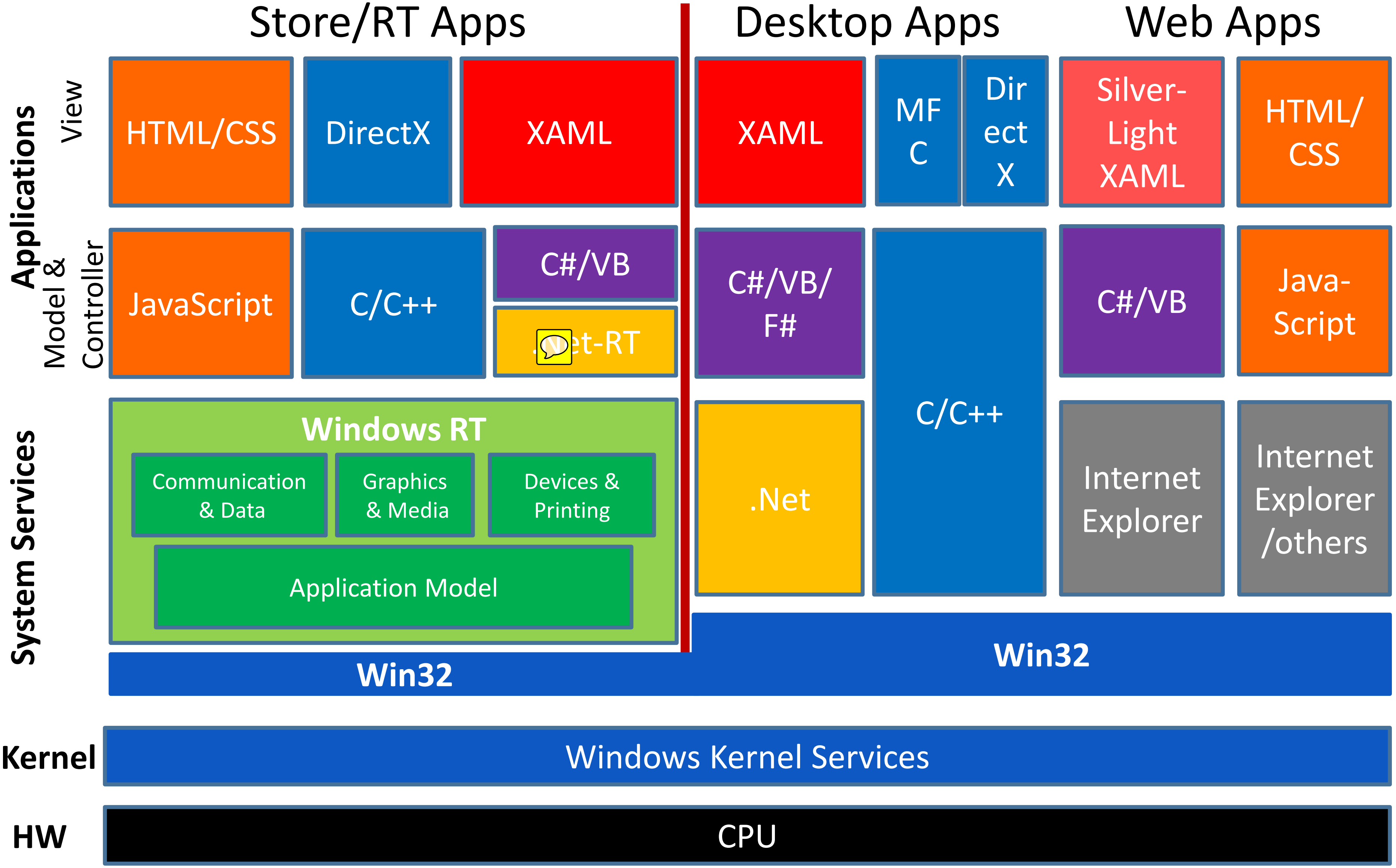
Windows 8 Store App Development

AARHUS UNIVERSITY
SCHOOL OF ENGINEERING

Windows 8 Application Types



Store/RT Apps

Desktop Apps

Web Apps

Applications

View

Model & Controller

HTML/CSS

DirectX

XAML

XAML

MFC

DirectX

Silver-Light XAML

HTML/CSS

JavaScript

C/C++

C#/VB

.Net-RT

C#/VB/F#

C/C++

C#/VB

Java-Script

System Services

Windows RT

Communication & Data

Graphics & Media

Devices & Printing

Application Model

.Net

Internet Explorer

Internet Explorer /others

Win32

Win32

Kernel

Windows Kernel Services

HW

CPU

Windows Store Apps

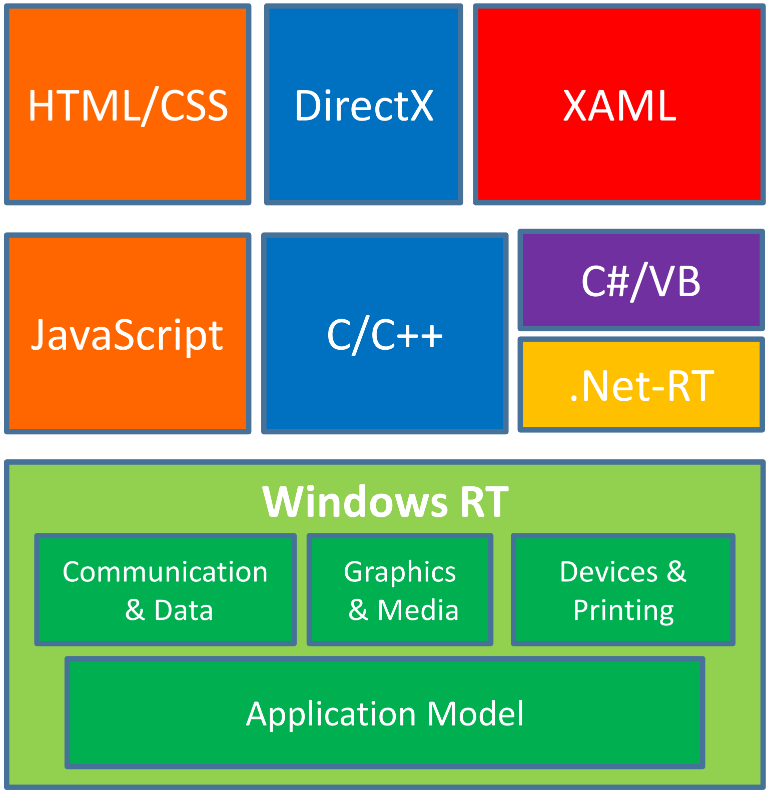
- Windows 8 introduces a new type of application:
  - the **Windows Store App**.
- Windows Store apps:
  - have a brand new look and feel,
  - run on a variety of devices, and
  - you sell them on the Windows Store.

Store vs. L-O-B

- Windows apps can be very broadly classified into consumer applications and enterprise line-of-business applications.
  - But there are lots of gray areas in between.
- Consumer apps can only be distribute through the Windows Store.
  - It provides developers with a channel through which their Windows Store apps can be discovered by users.
- Enterprise L-O-B apps can be distribute through side-loading or an enterprise app store.
- The Windows Store accepts only Windows Store apps.
- Because each Windows Store app declares its prerequisites, the Windows Store can inform users about system and software requirements so that they do not purchase apps that they won't be able to run.

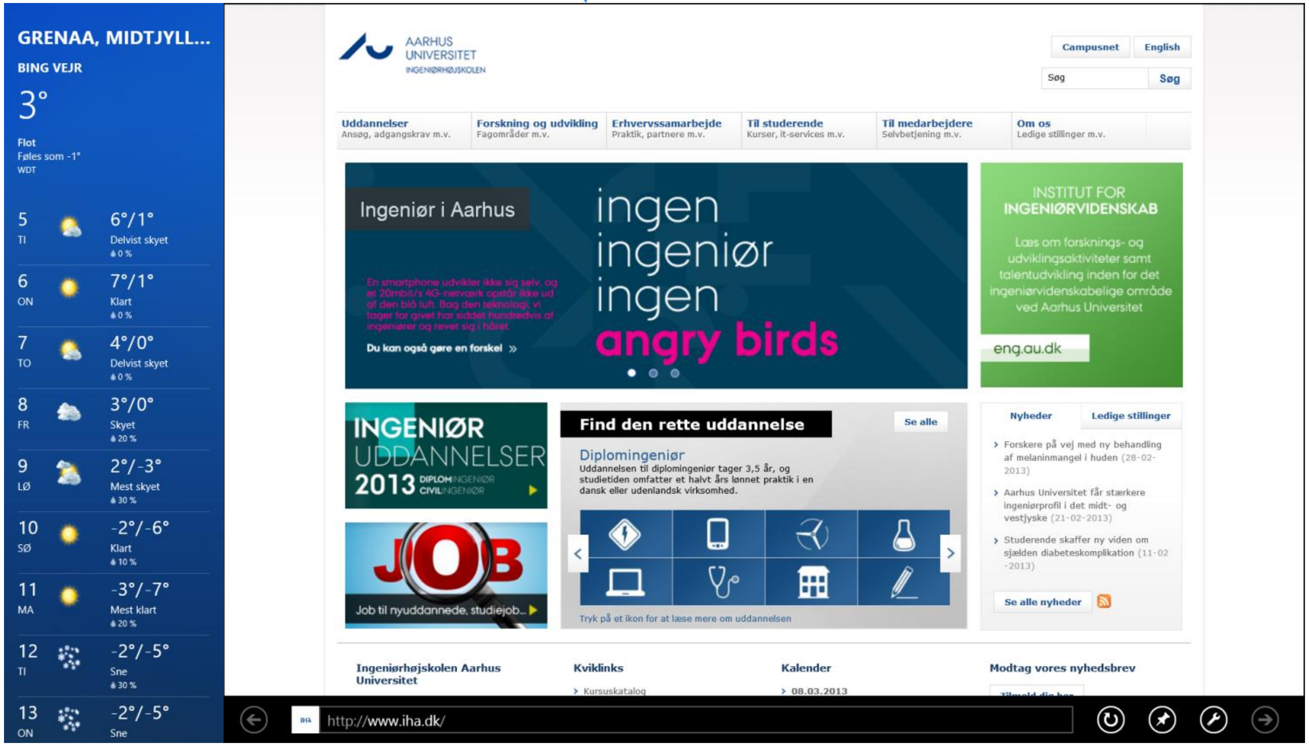
Language Choices

- You can develop Windows Store apps in a variety of languages:
  - Using XAML, with code-behind in C++, C#, or Visual Basic.
  - Using HTML5, Cascading Style Sheets, Level 3 (CSS3), and JavaScript.
  - Using DirectX and native C++ to take full advantage of graphics hardware.



One Window

- Apps have one window that supports multiple views.
- Apps have a single, chromeless window that fills the entire screen by default, so there are no distractions.
- A window may be snapped, filled or in portrait mode.



GRENAA, MIDTJYLL...

BING VEJR

3°

Flot
Føles som -1°
WDT

5	TI	6°/1°	Delvist skyet	0 %
6	ON	7°/1°	Klart	0 %
7	TO	4°/0°	Delvist skyet	0 %
8	FR	3°/0°	Skyet	20 %
9	LØ	2°/-3°	Mest skyet	30 %
10	SØ	-2°/-6°	Klart	10 %
11	MA	-3°/-7°	Mest klart	20 %
12	TI	-2°/-5°	Sne	30 %
13	ON	-2°/-5°	Sne	

En smartphone udvikler ikke sig selv, og et 20mbit/s 4G-netværk opstår ikke ud af den blå luft. Bag den teknologi, vi tager for givet har siddet hundredvis af ingeniører og revet sig i håret.

Du kan også gøre en forskel »

INSTITUT FOR INGENIØRVIDENSKAB

Læs om forsknings- og udviklingsaktiviteter samt talentudvikling inden for det ingeniørvidenskabelige område ved Aarhus Universitet

eng.au.dk
INGENIØR UDDANNELSER 2013 DIPLOMINGENIØR CIVILINGENIØR
JOB
Job til nyuddannede, studiejob...
Find den rette uddannelse
Se alle

Diplomingeniør
Uddannelsen til diplomingeniør tager 3,5 år, og studietiden omfatter et halvt års lønnet praktik i en dansk eller udenlandsk virksomhed.

Tryk på et ikon for at læse mere om uddannelsen

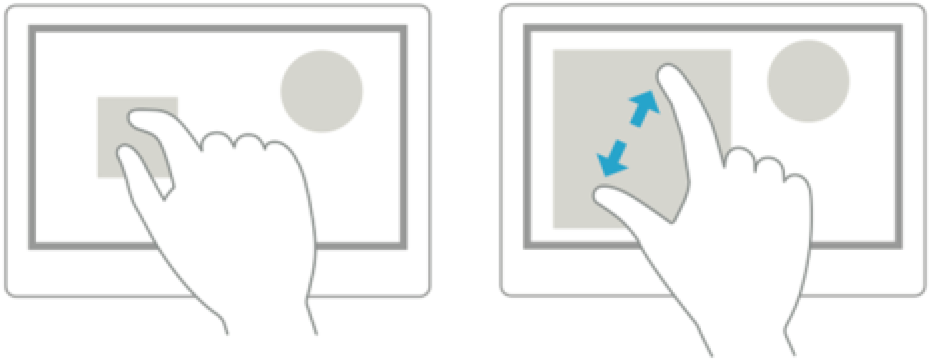
- Forskere på vej med ny behandling af melaninmangel i huden (28-02-2013)
- Aarhus Universitet får stærkere ingeniørprofil i det midt- og vestjyske (21-02-2013)
- Studerende skaffer ny viden om sjælden diabeteskomplikation (11-02-2013)

Different Form Factors

- A Windows Store app can support different layouts and views to create a fluid and harmonious experience across a variety of form factors and display sizes.

Touch And Pen Input

- Windows Store apps work smoothly with a variety of input sources, including touch, pen, mouse, and keyboard input.
- You can use a single set of events that work for all these input sources.

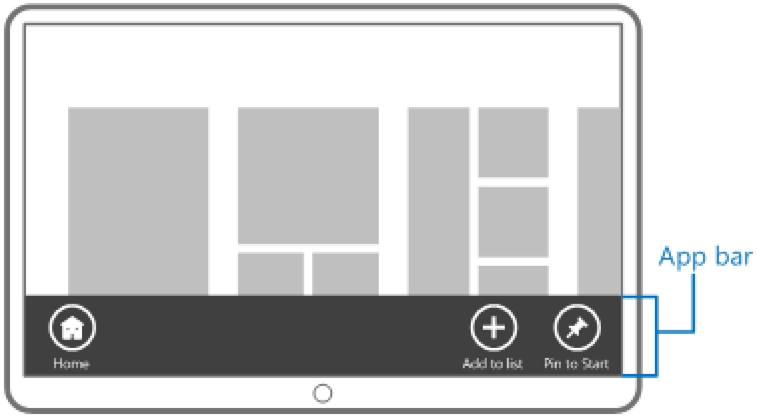


Apps can talk to each other

- App contracts are a way for users to seamlessly search across and share content between different apps.
- They extend the usefulness of your app by eliminating the need to work with varying standards or app-specific APIs to access data stored or created by another app.

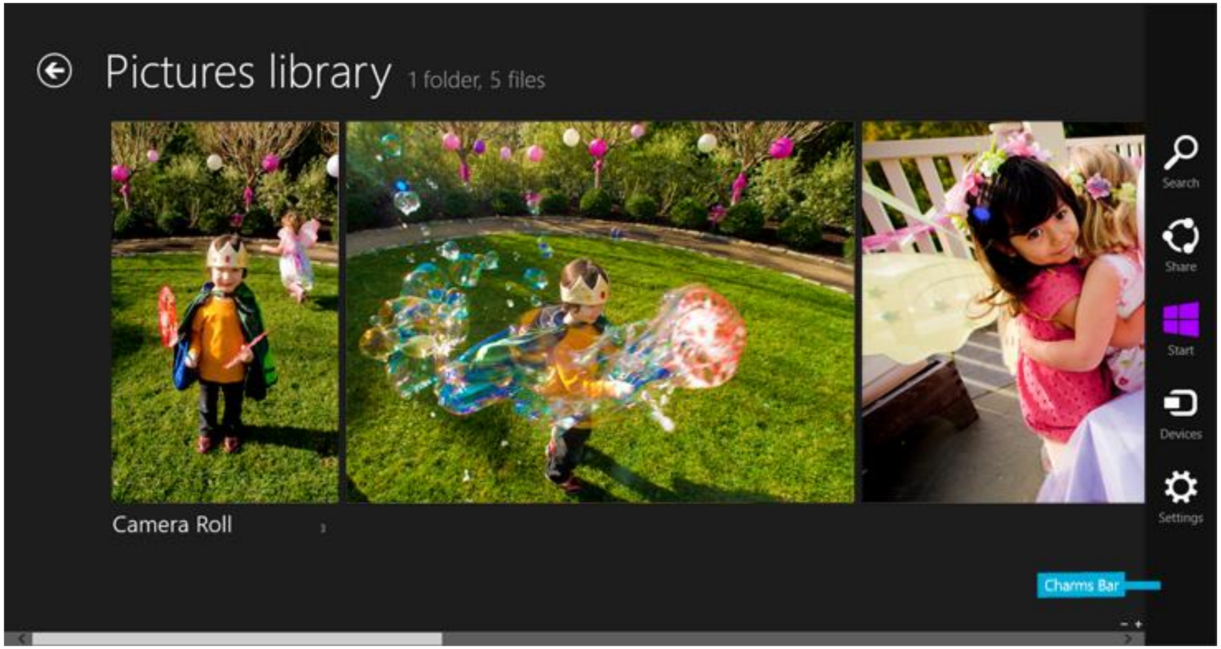
The App Bar

- Outside of the app window, the app bar is the primary command interface for your app.
- Use the app bar to present navigation, commands, and tools to users.
- The app bar is hidden by default and appears when users swipe a finger from the top or bottom edge of the screen.
- It covers the content of the app and a user can dismiss it with an edge swipe, or by interacting with the app.



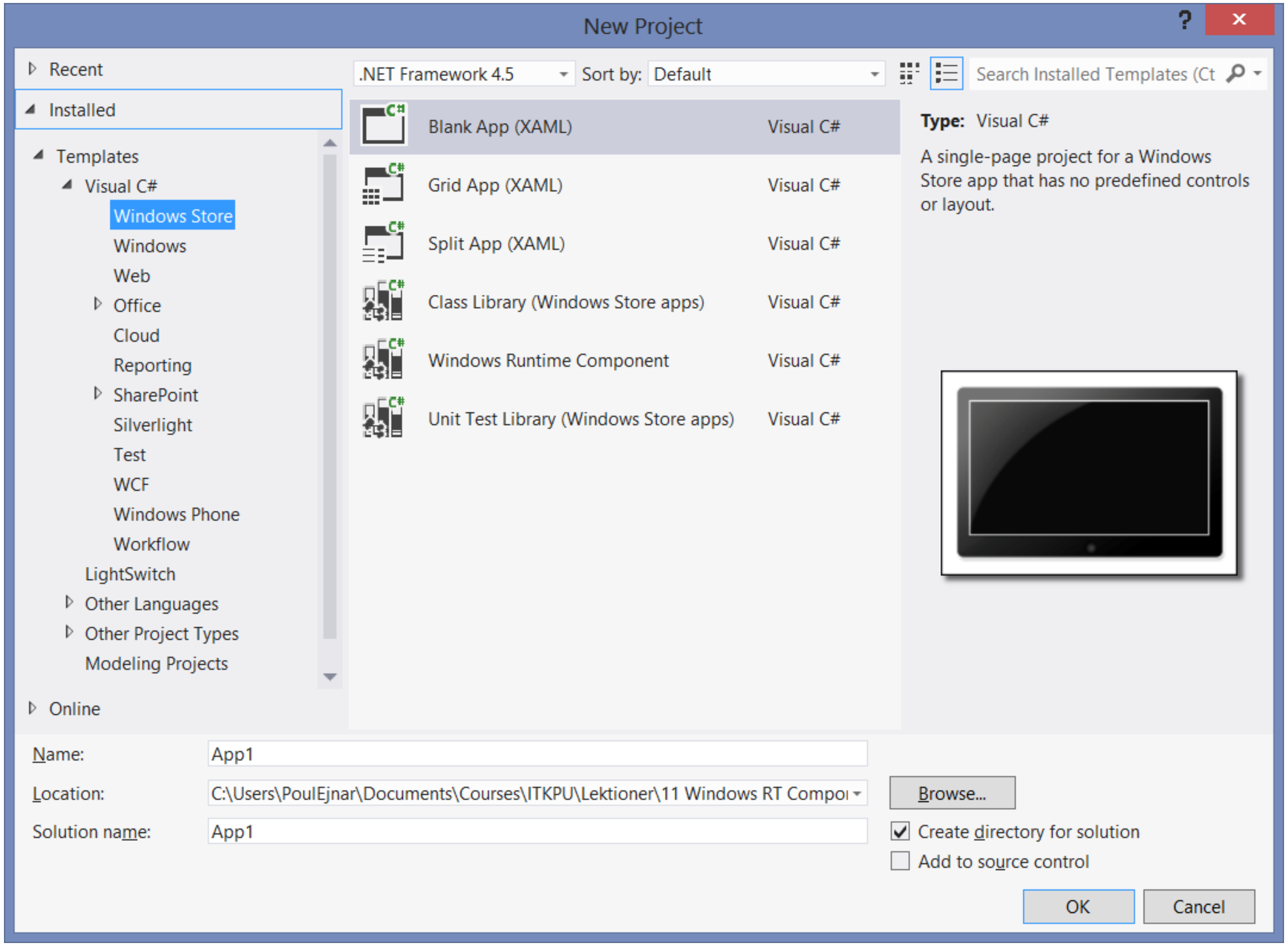
The Charms

- The charms are a specific and consistent set of buttons in every app:
  - search
  - share
  - connect
  - settings
  - start



Preconditions

- Windows 8 Modern apps can only be developed on Windows 8!
- And with Visual Studio 2012.



New Project

.NET Framework 4.5 Sort by: Default

Blank App (XAML)	Visual C#
Grid App (XAML)	Visual C#
Split App (XAML)	Visual C#
Class Library (Windows Store apps)	Visual C#
Windows Runtime Component	Visual C#
Unit Test Library (Windows Store apps)	Visual C#

Type: Visual C#

A single-page project for a Windows Store app that has no predefined controls or layout.

Name: App1

Location: C:\Users\PoulEjnar\Documents\Courses\ITKPU\Lektioner\11 Windows RT Compo

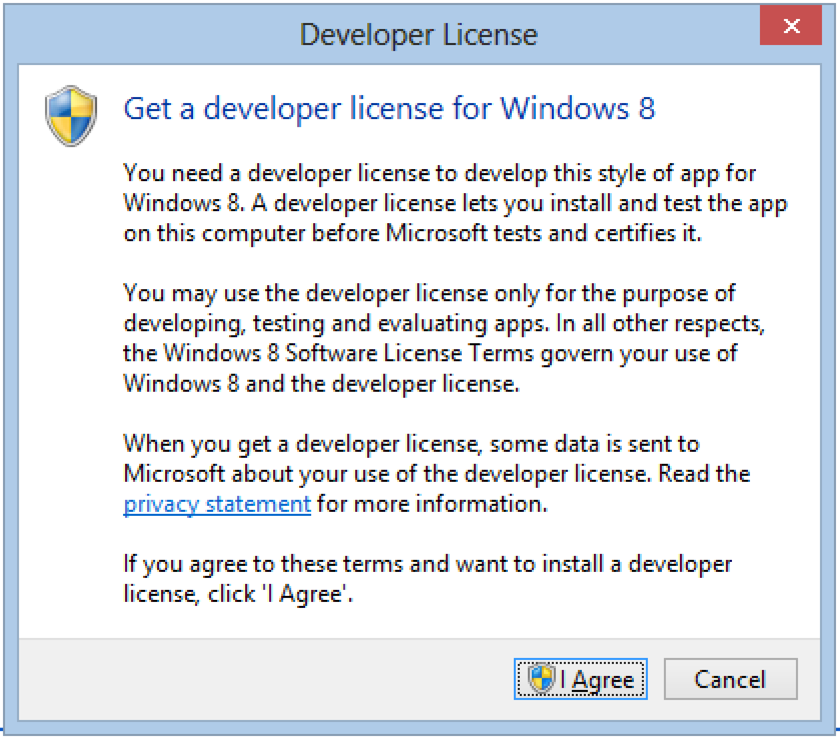
Solution name: App1

☑ Create directory for solution

☐ Add to source control

Microsoft Account

- The first time you create a Windows 8 app on your machine, you'll be prompted to create a developer license.
- This is free, and is granted for 30 days for non-store use and 90 days if you have a Windows Store account.
- Renewing is painless as long as you have an internet connection.



Developer License

Get a developer license for Windows 8

You need a developer license to develop this style of app for Windows 8. A developer license lets you install and test the app on this computer before Microsoft tests and certifies it.

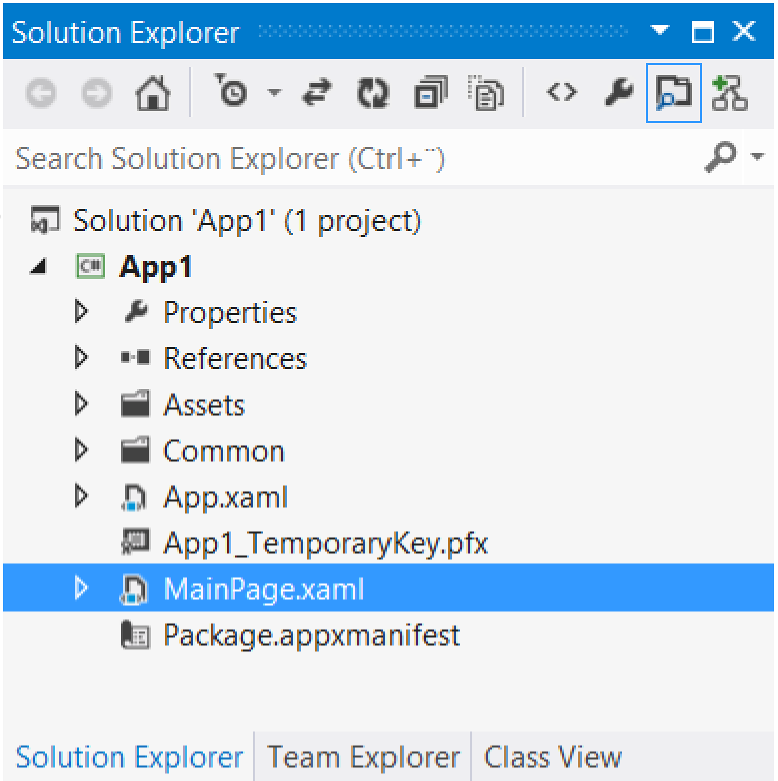
You may use the developer license only for the purpose of developing, testing and evaluating apps. In all other respects, the Windows 8 Software License Terms govern your use of Windows 8 and the developer license.

When you get a developer license, some data is sent to Microsoft about your use of the developer license. Read the privacy statement for more information.

If you agree to these terms and want to install a developer license, click 'I Agree'.

Template Solution Items

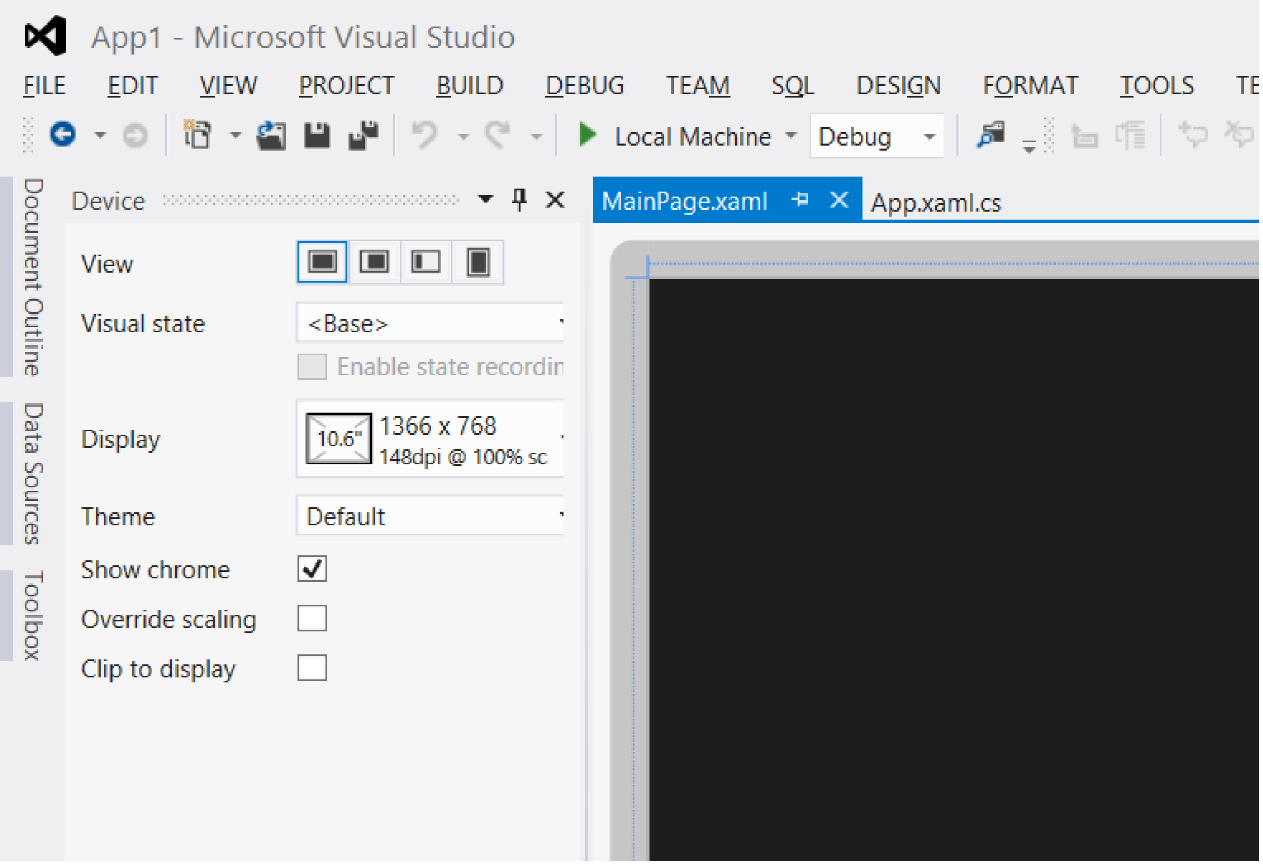
- The Common folder contains the very important StandardStyles.xaml file.
  - This is where the Windows 8 modern app styles reside.
  - Don’t Mess with the Standard Styles.
- The Assets folder contains the various application images.
- The Package.appxmanifest” file is where you specify the runtime configuration for your application, including the logos, supported orientations, background color, capabilities and declarations.



Solution 'App1' (1 project)
- App1
  - Properties
  - References
  - Assets
  - Common
  - App.xaml
  - App1_TemporaryKey.pfx
  - MainPage.xaml
  - Package.appxmanifest

The Device Pane

- Open MainPage.xaml and you’ll see the design surface and the usual developer window panes.
- The Device window panel is new.



View

Visual state <Base>

□ Enable state recordin

Display 10.6" 1366 x 768 148dpi @ 100% sc

Theme Default

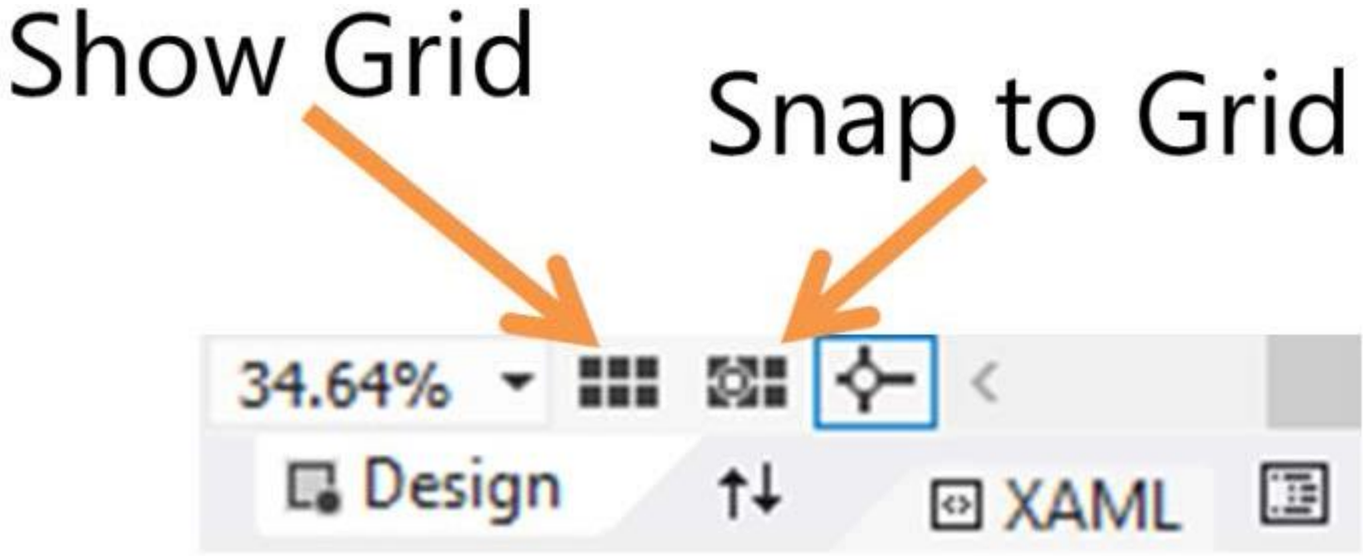
Show chrome ☑

Override scaling □

Clip to display □

The Layout Grid

- Modern Style apps are laid out on a 20 pixel grid.
  - Each 20 pixel square is called a “unit”.
- The grid is further subdivided into 5x5 pixel squares to assist with spacing of elements.



PACKAGING FOR WINDOWS STORE APPS

App Package

- A Windows Store app is deployed in an app package,
  - which is a compressed file that contains the files that make up the app,
  - along with a manifest that describes the app to Windows.
- An app package has an identity that includes:
  - name,
  - publisher,
  - version, and
  - other information.
- The app package is the unit of deployment and servicing.
- You write the app, and Visual Studio and Windows take care of packaging and deploying it.
  - All files that your app needs ship with your app.

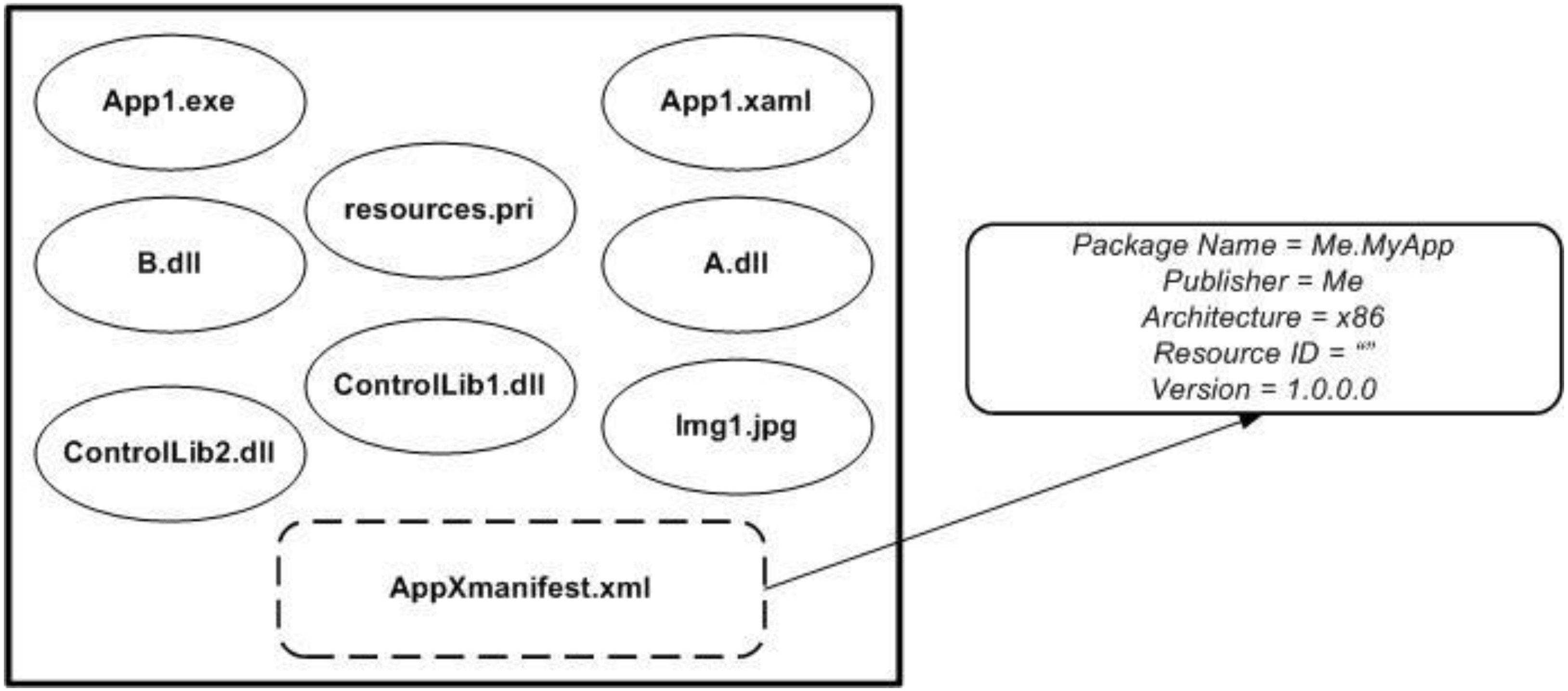
App Manifest

- The manifest explicitly declares the package's:
  - prerequisites,
  - dependencies on other app packages,
  - requested capabilities, and
  - OS integration points.

- The manifest enables Windows to understand the software's footprint on the system and to cleanly remove the software on behalf of the developer.

- Frictionless deployment and the ability to uninstall software easily brings Windows client apps closer to the cleaner, low-impact world of web apps.

App Package Example


App1.exe
App1.xaml
resources.pri
B.dll
A.dll
ControlLib1.dll
ControlLib2.dll
Img1.jpg
AppXmanifest.xml
Package Name = Me.MyApp
Publisher = Me
Architecture = x86
Resource ID = ""
Version = 1.0.0.0