

Get your head in the cloud

Billy Hollis

Next Version Systems

Nashville, Tennessee, USA

@billyhollis

nextver.com

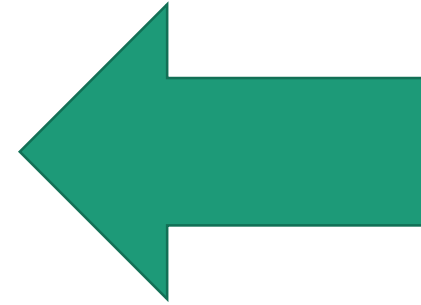
What are you more excited about?

The Cloud

The latest superhero movie



Immediate
Near Term



Medium Term

Long Term

A warning

This is all just my opinion.

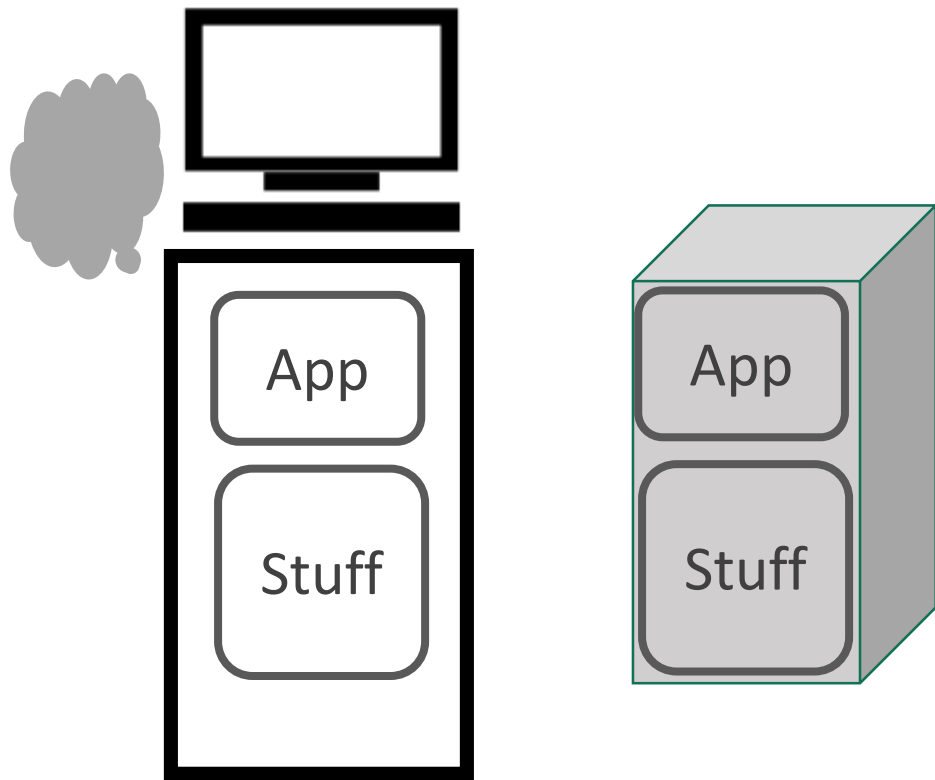
In the South, we have a saying: “If you’re so smart, why ain’t you rich?”

I’m not rich....

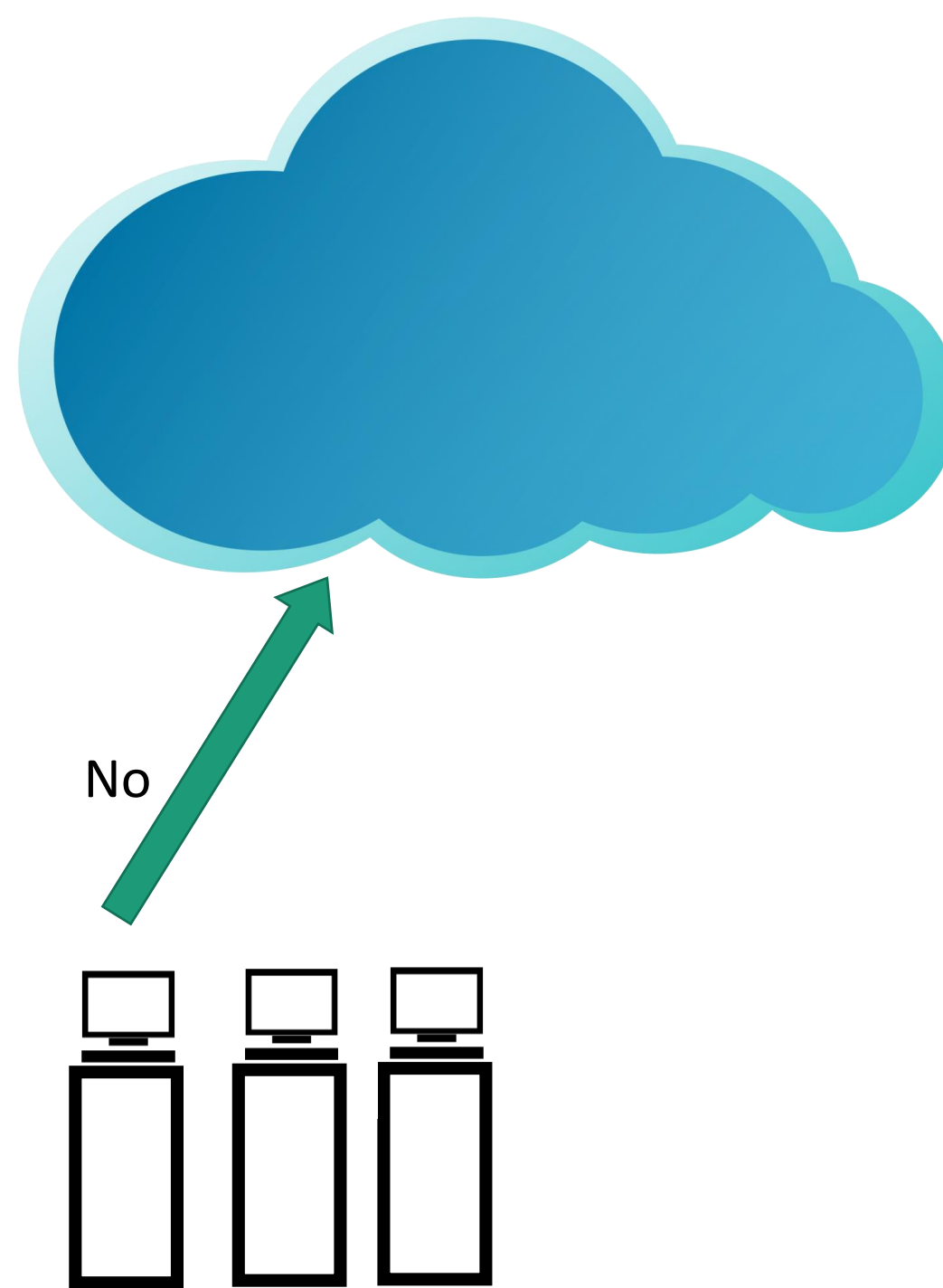
But, I am independent in my opinions

- I don't represent Microsoft, Amazon, Google, or anyone else
- I lead a team that does real work
- I first started writing code for money in 1978 so I have seen many waves of change
 - Why do you think my hair is this white?

Using cloud for infrastructure replacement



Can we host in existing server?



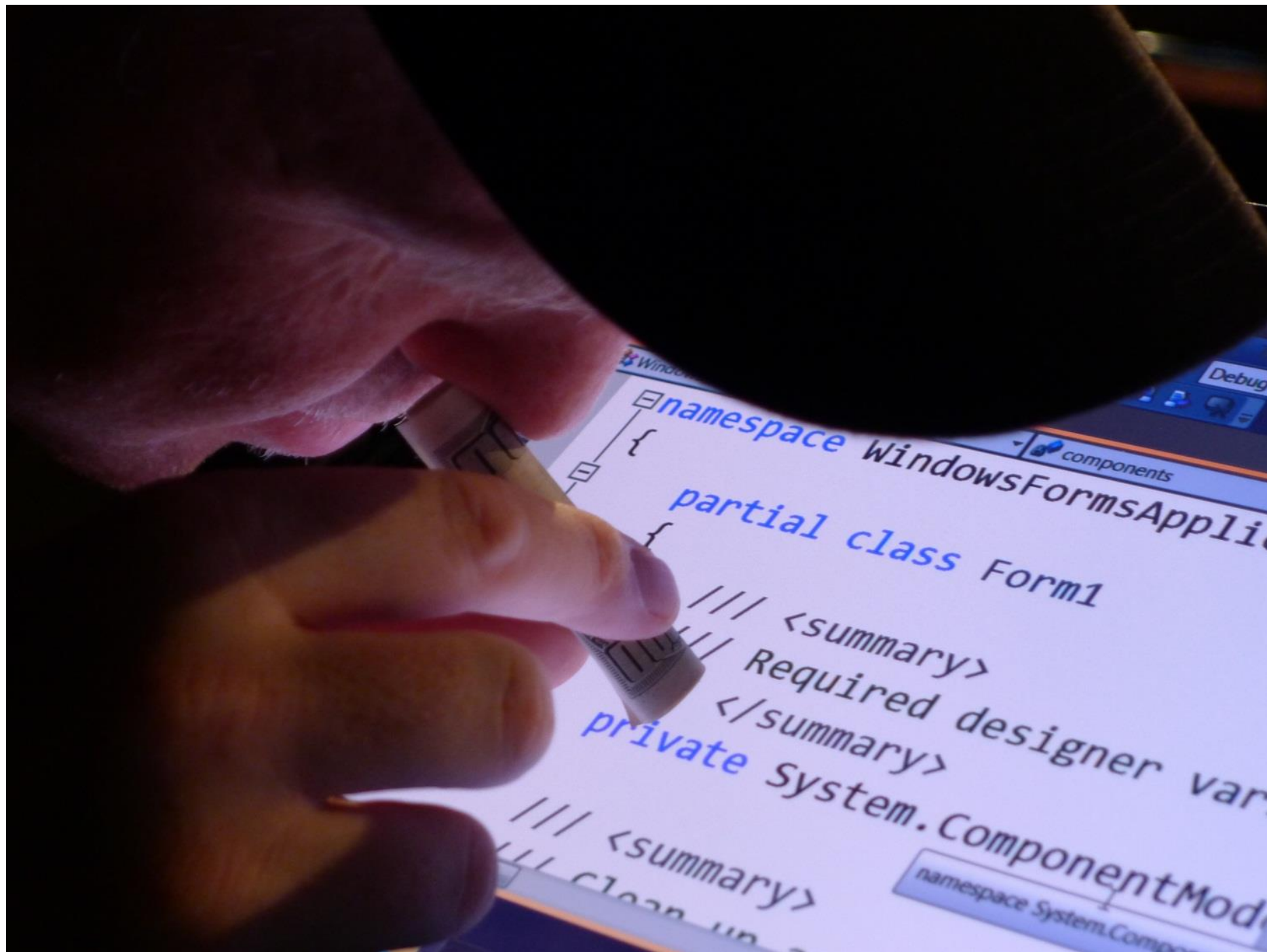
This is IaaS – Infrastructure as a Service.
This activity is sometimes called “Lift and Shift”.
Cloud also offers **scalability** for these apps
just by turning up capacity. Or you can stand
up a web site very quickly.

Why do you care about this?

You don't.

Only IT people care. Doesn't involve serious coding.

My observation: If there's no coding, developers don't care. That's because many of them are substance abusers - addicts.



But there is an important use of IaaS for developers

- Some development teams are already moving to virtual machines in the cloud as their developer box

Why use VMs for development?

- Isolation – Can use VM for only one project
- Especially helpful for consultants who work with multiple clients
- Extra power without buying hardware
- Only pay for what you use
 - But you do need a Windows license
- Get to it from anywhere

Create a resource

All services

FAVORITES

Dashboard

All resources

Resource groups

App Services

Function Apps

SQL databases

Azure Cosmos DB

Virtual machines

Load balancers

Storage accounts

Virtual networks

Azure Active Directory

Monitor

Home > Virtual machines > Compute

Virtual machines
(Default Directory)

Edit columns More

name...

alDev

Compute

Filter

Search Compute



Windows Server
2016 Datacenter
Microsoft



Teradata
Viewpoint
Teradata



Data Science
Virtual Machine
Microsoft



CIS Windows
Server 2016
Center For Internet...



Docker on Ubuntu
Server
Canonical + Micros...

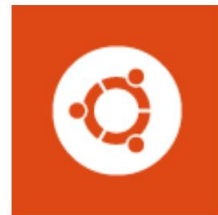


Carto Builder 2.1
CARTO

Operating Systems



Windows Server
Microsoft



Ubuntu Server
Canonical



Red Hat
Enterprise Linux
Red Hat



Windows Client
Microsoft



CentOS-based 7.3
Rogue Wave Softw...



SLES 12 SP3
SUSE

Solution Templates



Another scenario – you work on two different apps that require incompatible frameworks

App A requires FX version
1.2.3.456

App B requires FX version
1.2.3.789

But these don't run side by side, so...



Why do you care about using cloud VMs for dev?

- Less time juggling stuff on your machine, more time coding
- Still pretty boring, though
- Certainly not a game changer
- But IaaS can affect your strategy for some desktop apps

Cloud-based desktop applications

- Visual Studio is a complex desktop application, and runs well from the cloud for most projects
- Other complex desktop apps may also be candidates
 - Applications done on Citrix today are good candidates
- Can help provide more reach for complex desktop apps
 - Alternative to web-based front ends for complex apps
- Still not a game changer – just offers more reach and new deployment options

Superficial analysis

“There’s no such thing as the cloud.
It’s just someone else’s computer.”

Equivalent logic

“There’s no such thing as a restaurant. It’s just someone else’s kitchen.”

Superficial analysis

“There’s no such thing as the cloud.
It’s just someone else’s computer.”

So where will the cloud lead to real changes for you?

- Lots and lots of APIs to leverage
 - Mapping
 - Transportation routing
 - Voice recognition
 - Chat bots (with caution – I think these are over-hyped)
 - Service bus (remote events and notifications)
 - SMS Texting
 - Data analytics
 - Etc. Etc. Etc. Etc. Etc.....
- Basically, if there is a package for it today, you can probably expect a cloud version of it available now or coming soon



Plumbing layer 4

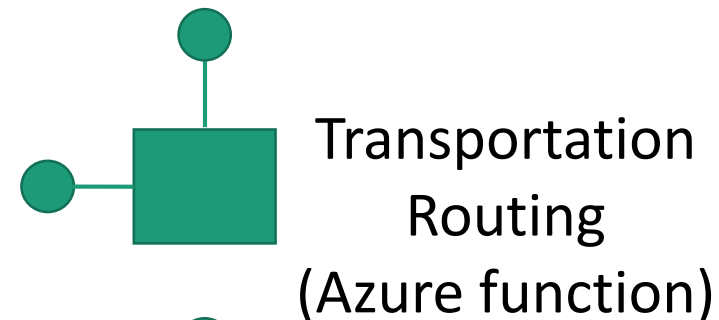
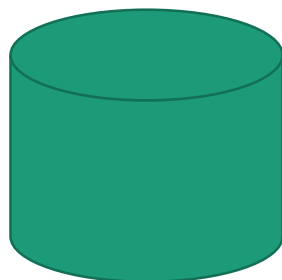
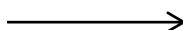
Plumbing layer 3

Plumbing layer 2

Plumbing layer 1

Data access and modelling

Maybe some
business logic buried
in these layers
somewhere



Cloud APIs

WPF application with advanced mapping

DISPATCH MANAGER

Filter Deliveries... search

13 Degree Day 0 Emergency 0 High Priority 0 Medium Priority 13 Low Priority 13 Unassigned

Brians Tickets
Joe Johnson
Mogo Demo
Start First Customer
End Last Customer
0 out of 13 delivered
Remaining time 0d 18h 44m
Total qty 3,856
Remaining qty 3,856
3,200

Load 1
Joe Johnson
Mid.Com'98 Ford 350
Start Willmar Fill Point
End Willmar Fill Point
0 out of 28 delivered
Remaining time 0d 0h 0m
Total qty 13,048
Remaining qty 13,048
3,500

Load 2
Larry Sampson
LC 90 Ford/2600 gal/ATR-591
Start Bird Island Fill Point
End Bird Island Fill Point
0 out of 48 delivered
Remaining time 0d 0h 0m
Total qty 21,009
Remaining qty 21,009
3,600

Nate test
Joe Johnson
Mid.Com'98 Ford 350
Start First customer
End Last customer
0 out of 0 delivered
Remaining time 0d 0h 0m
Total qty 0
Remaining qty 0
3,500

Create Delivery... Synchronize All Create Load... Optimize Load... More options...

These APIs add great new capabilities, but they are just bolted on to what you have now

- That's good because you can plug it in and use it now
- But it's not the endpoint for using the cloud

IoT demonstration from Microsoft Build 2018

- Isolated device
- Everything deploys from cloud (Azure IoT in this case)
- Device talks directly to logic in the cloud
- Video at <http://bit.ly/AzureIoTBuild>
- End result: You press a button and get a pointless tweet

At first glance, this looks like a pointless capability

I could have hooked a button to my device and caused it to generate a pointless tweet ten years ago



thisispointless @thisispointless · Aug 11

How do you lower the difficulty settings in Visual Studio?

Asking for a friend.



thisispointless

@thisispointless

Follow



My standard response when someone suggests that I do something "for exposure":
"What's the current exchange rate between exposure and dollars?"

9:43 AM - 23 Mar 2018



thisispointless @thisispointless · Jul 17

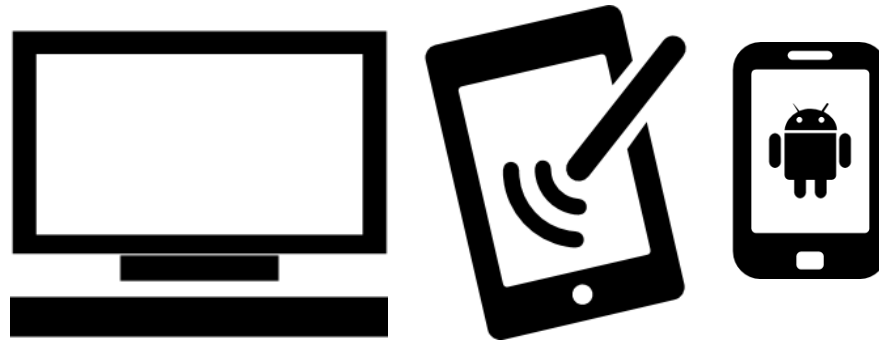
Exclusive photo of Microsoft's new Azure data switch



So why was this demo in the keynote at Build?

- Presenter focused on how easy it is to get IoT devices up and running
- This is important – without a way to deploy, say, ten thousand sensors, the Internet of Things would be quite constrained
- But he missed a larger point for developers:

This button device is an example of how the cloud standardizes plumbing and deployment code, and reduces the need for you to write it



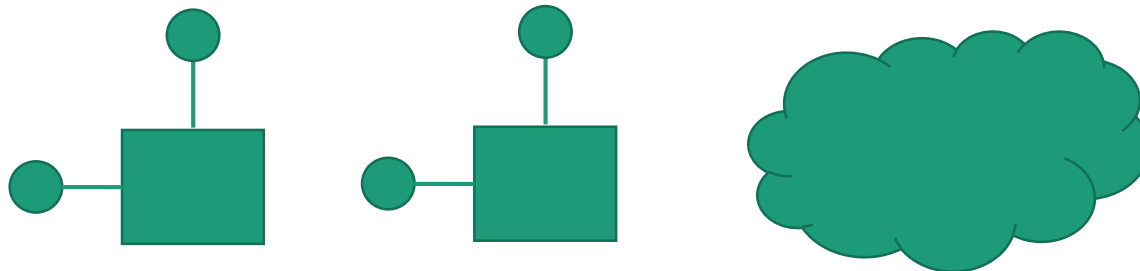
Plumbing layer 2

Plumbing layer 1

Data access and modelling

Cloud APIs

Transportation
Routing
(Azure function)



Data and
business logic in
the cloud
instead of on
your servers

Do you write lots of plumbing or deployment code?

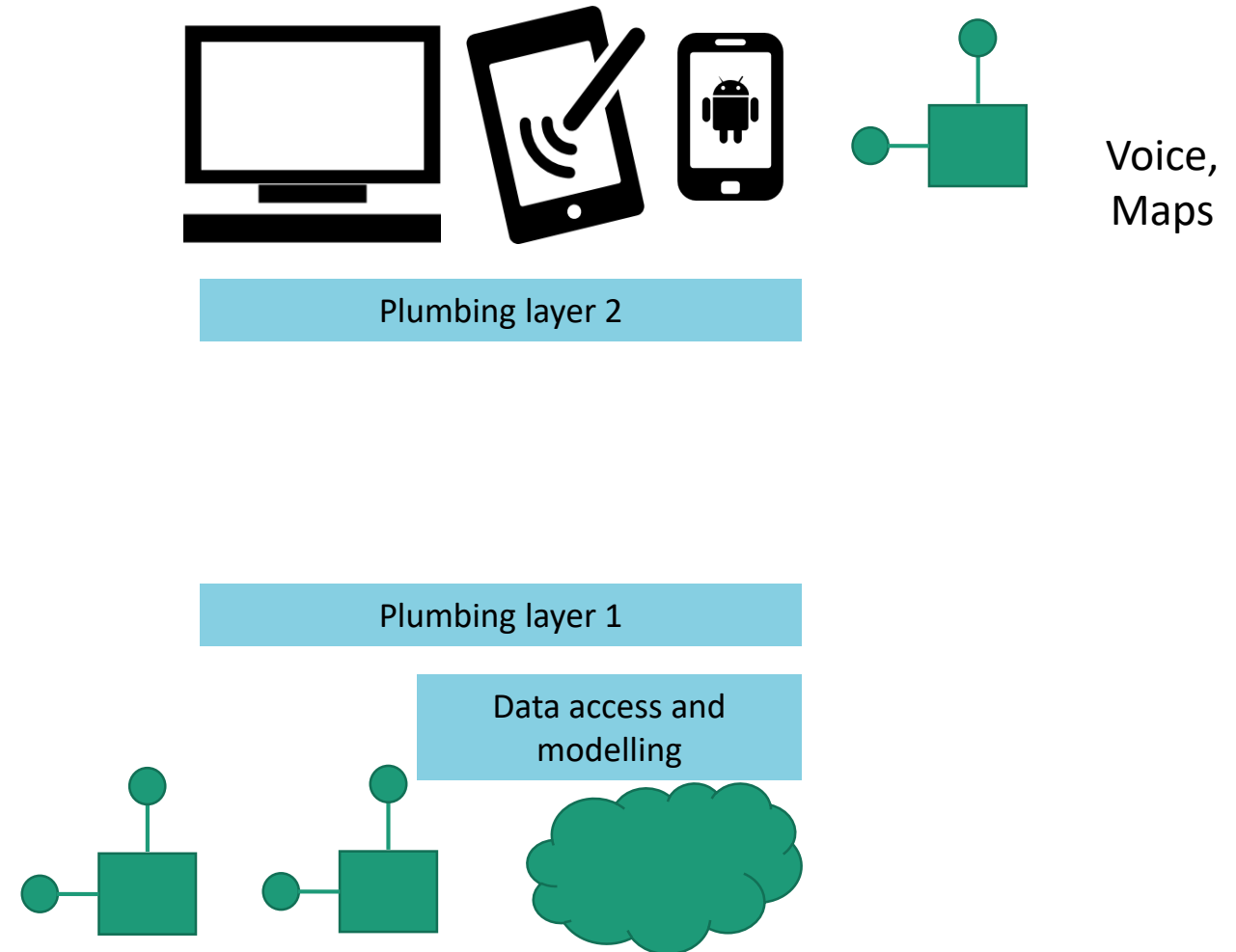
Cloud-based apps probably won't need as much.

Adjust your expertise accordingly. Cloud-based apps have other areas that deliver higher value.



But this is not where we're going to end up.

This is just the beginning of the changes in construction of cloud based apps.



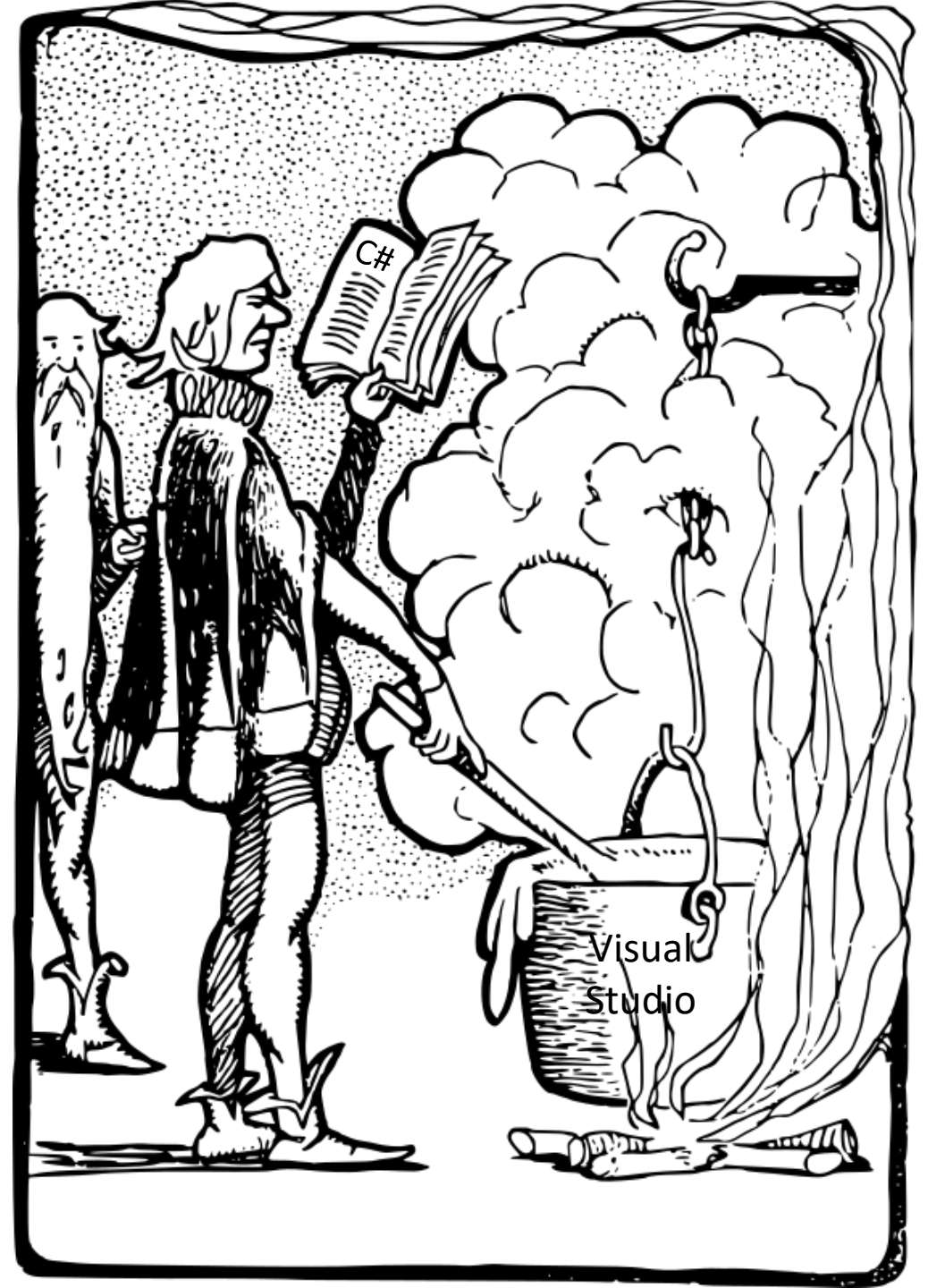
A question:

And I need an honest answer.

How many of you feel overwhelmed by the complexity of today's application development and the constant, unrelenting change in development technologies?

If you are, I guarantee that the decision makers in your company are too

Most don't understand what you do. You might as well be stirring a cauldron and saying magic spells.

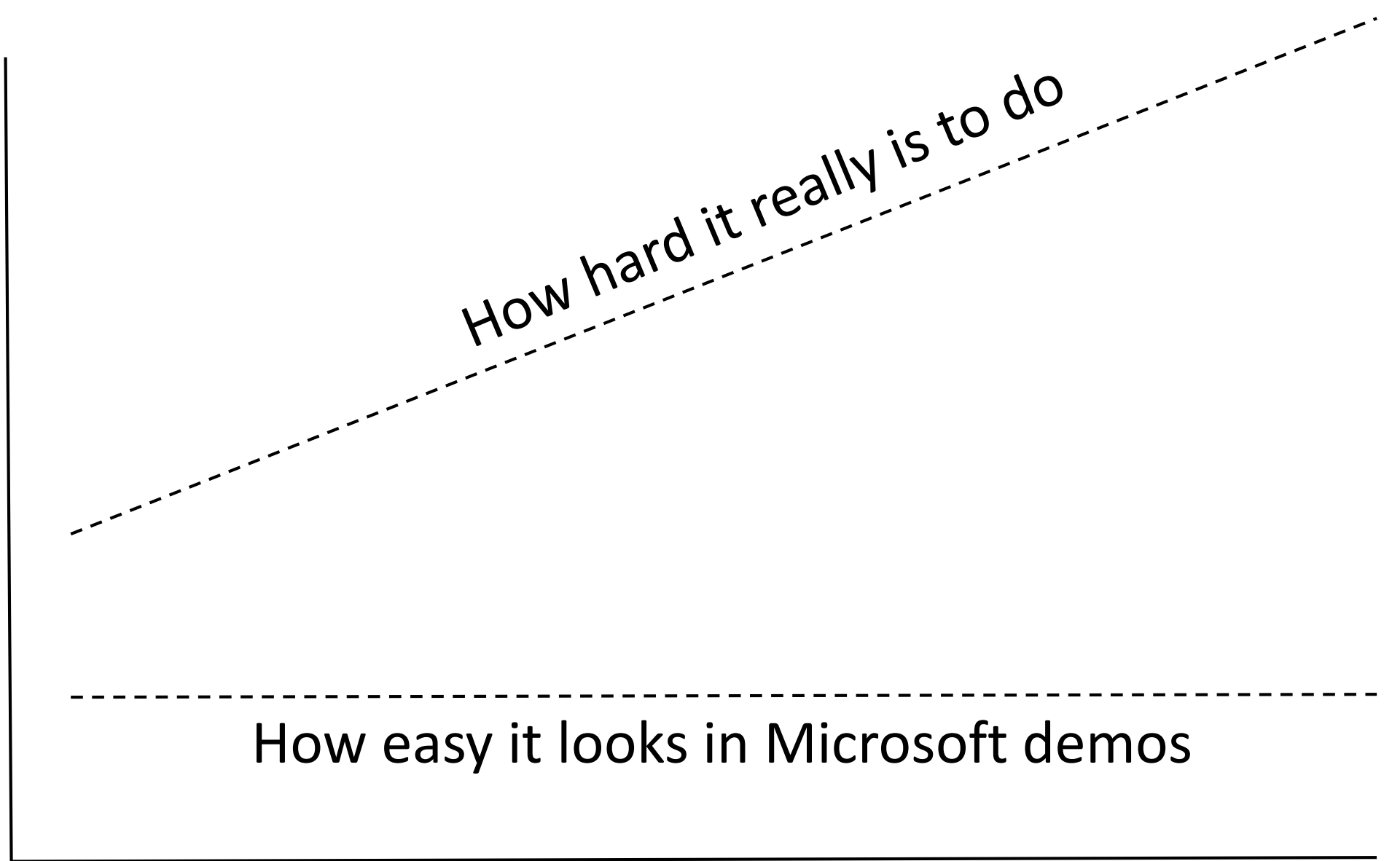


Difficulty

How hard it really is to do

How easy it looks in Microsoft demos

1998 2000 2002 2004 2006 2008 2010 2012 2014 2016 2018



Angular

Dojo

Spry

Kendo

Breeze

React

Knockout

JQuery

Bootstrap

Socket.IO

Wakanda

Polymer

This has problems

- You would not be laughing at all this if it did not contain some truth
- Still skeptical? Here's more evidence.

From my buddy David
Neal in Atlanta

He sells these on Amazon.
He's selling a lot of them.



Or you can buy this
coffee cup



“Javascript at a point becomes kind of unmaintainable. The fault basically lies in the language itself.”

John Robertson, Microsoft executive, spoken in a session at Build 2015

Other indications

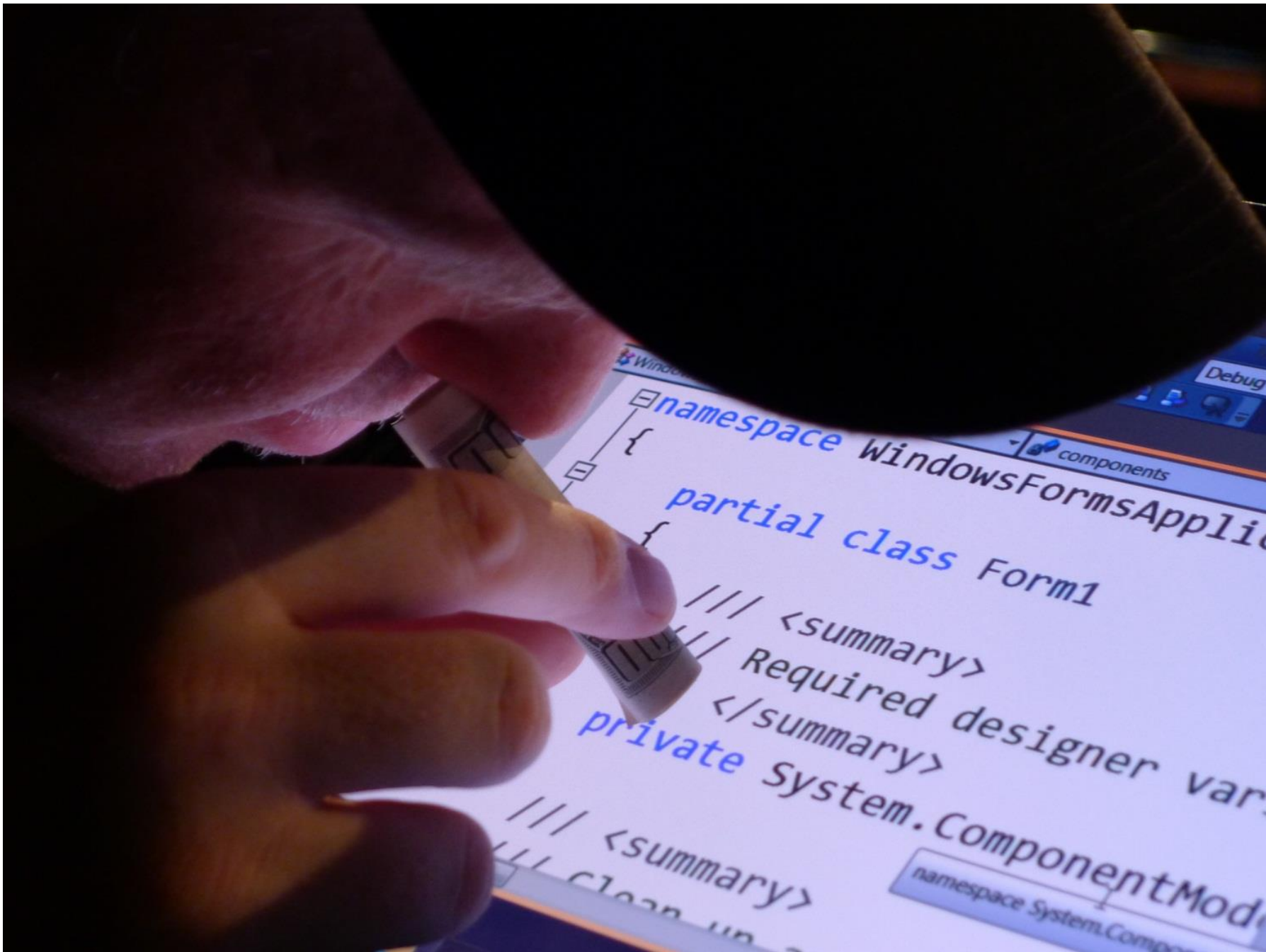
- “Page rot” – web pages created ten years ago with simpler technologies still work fine, but pages created two years ago don’t work because one of the n libraries they use has changed
- Lack of entry point for new developers
 - They have to learn too much just to get started in today’s HTML5/JS world
- Browser incompatibility is a constant struggle
- Decision maker disenchantment increasing

In a cloud world, is the answer to layer on more complexity?

- AI APIs
- Voice APIs
- Bot APIs
- 3D APIs for virtual and mixed reality
- Distributed events (Azure Message Bus, Signal R, etc.)
- Telepathic APIs (OK, that one's not really available. Yet.)

You may not see these problems as clearly as your management

- I talk to development managers and higher executives, and I can tell you that they are more dissatisfied each year
 - They just don't know what to do about it yet
- Many of them feel they are close to the breaking point on handling complexity
 - That's why so many of them are still on fifteen year old technologies
- For many developers, as long as someone is paying them to write code all day, they don't care about anything else



```
namespace WindowsFormsApplic
{
    partial class Form1
    {
        /// <summary>
        /// Required designer var
        </summary>
        private System.ComponentModel
        /// <summary>
        /// Clean up
    }
}
```

Right now we're patching our way around some of the worst problems

- TypeScript, for example

But the cloud is a platform change, and that could lead to a completely new way to do software development

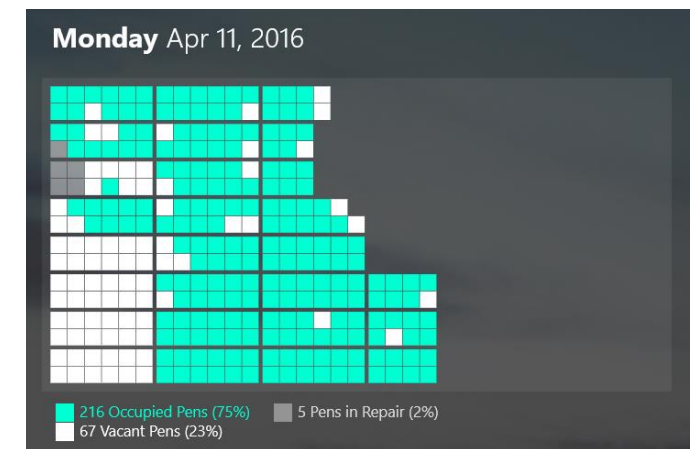
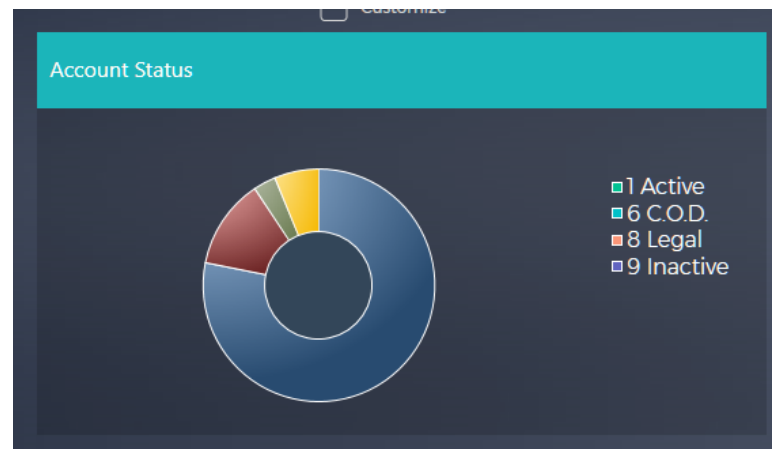
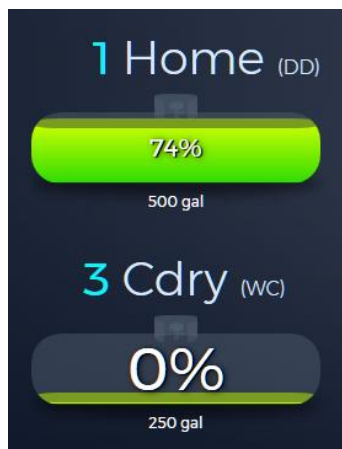
- Example from the past: Visual Basic, Foxpro, and Access in the 1990s
- All of them covered up the complexity of the Windows API

Other forces driving change

- The changing role of data
- The emergence of AI
- Cloud based engines

Changing role of data

- The cloud makes it easier to bring together data from different sources
- Some are traditional databases, with text-based data about entities
- Others are based on sensors, measurements, activity counts, and other ways that data can be automatically created and stored



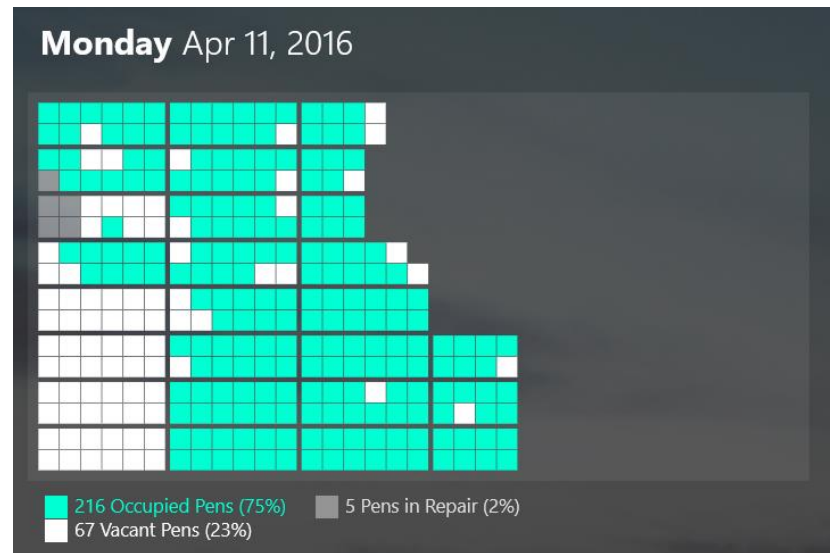
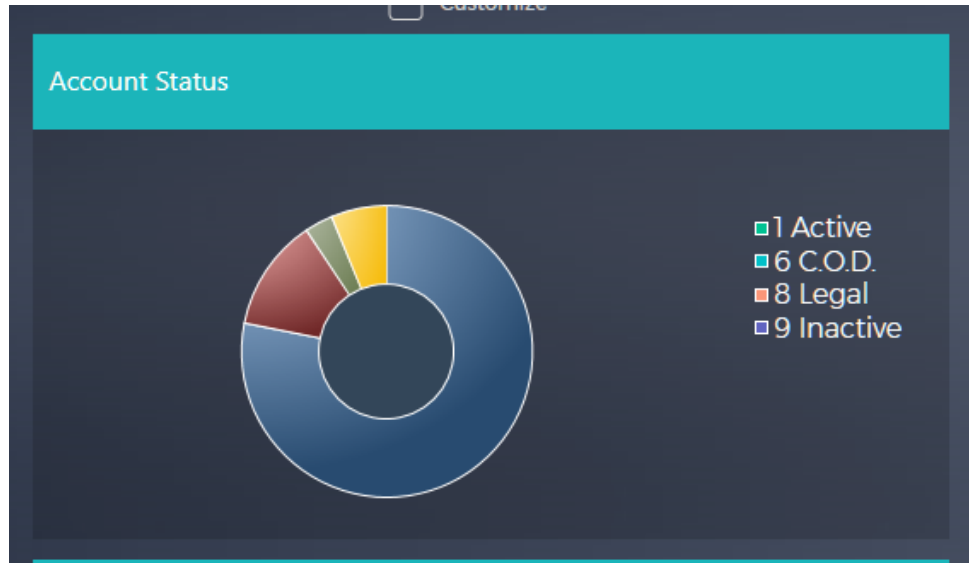
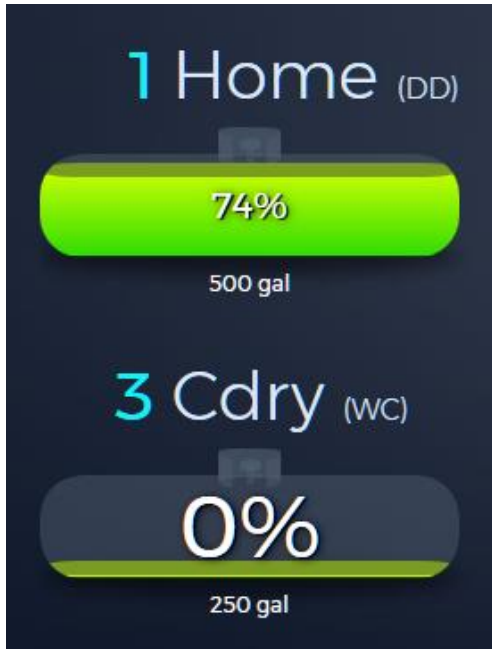
Working with data in business apps

Up to now

- Lots of screens to enter and edit data
 - Up to 60-70% of pages/views are “CRUD”
- Data often used in searches to find a master record and its detail records
- Data commonly displayed in data grids

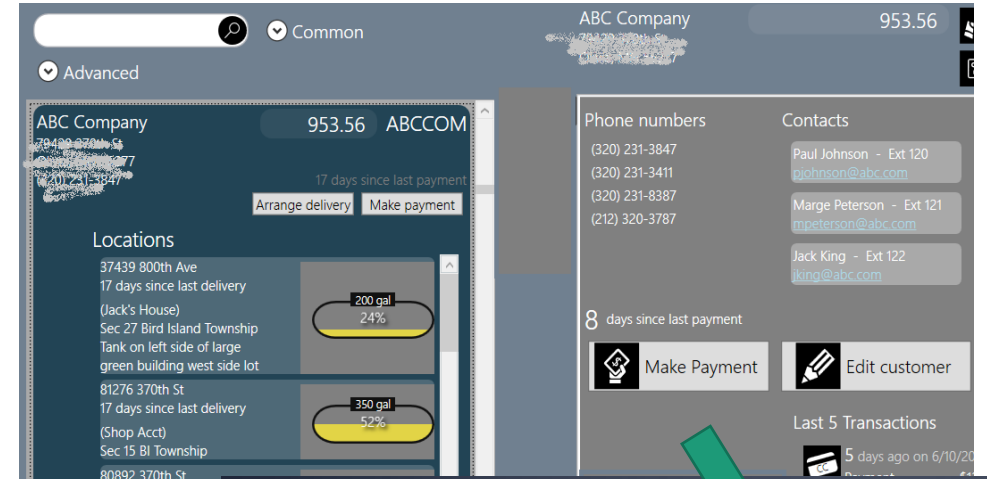
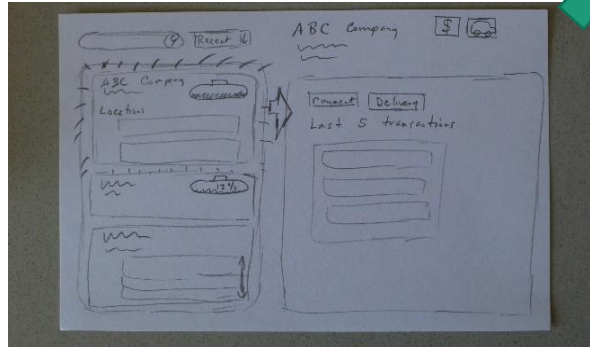
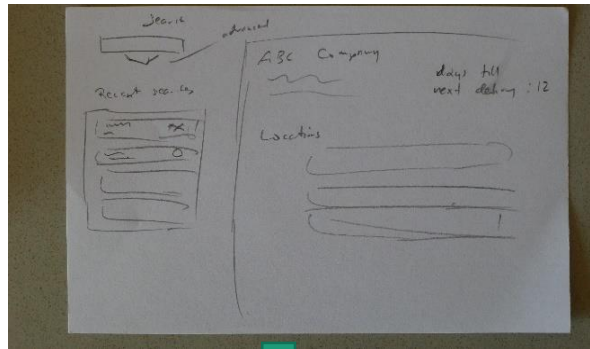
Going forward

- Data comes from many sources
- Most data no longer entered by a human
 - “CRUD” drops to smaller fraction of pages/views
- Data often used for analysis and decision support
- Data commonly displayed in ways that make it easy to consume

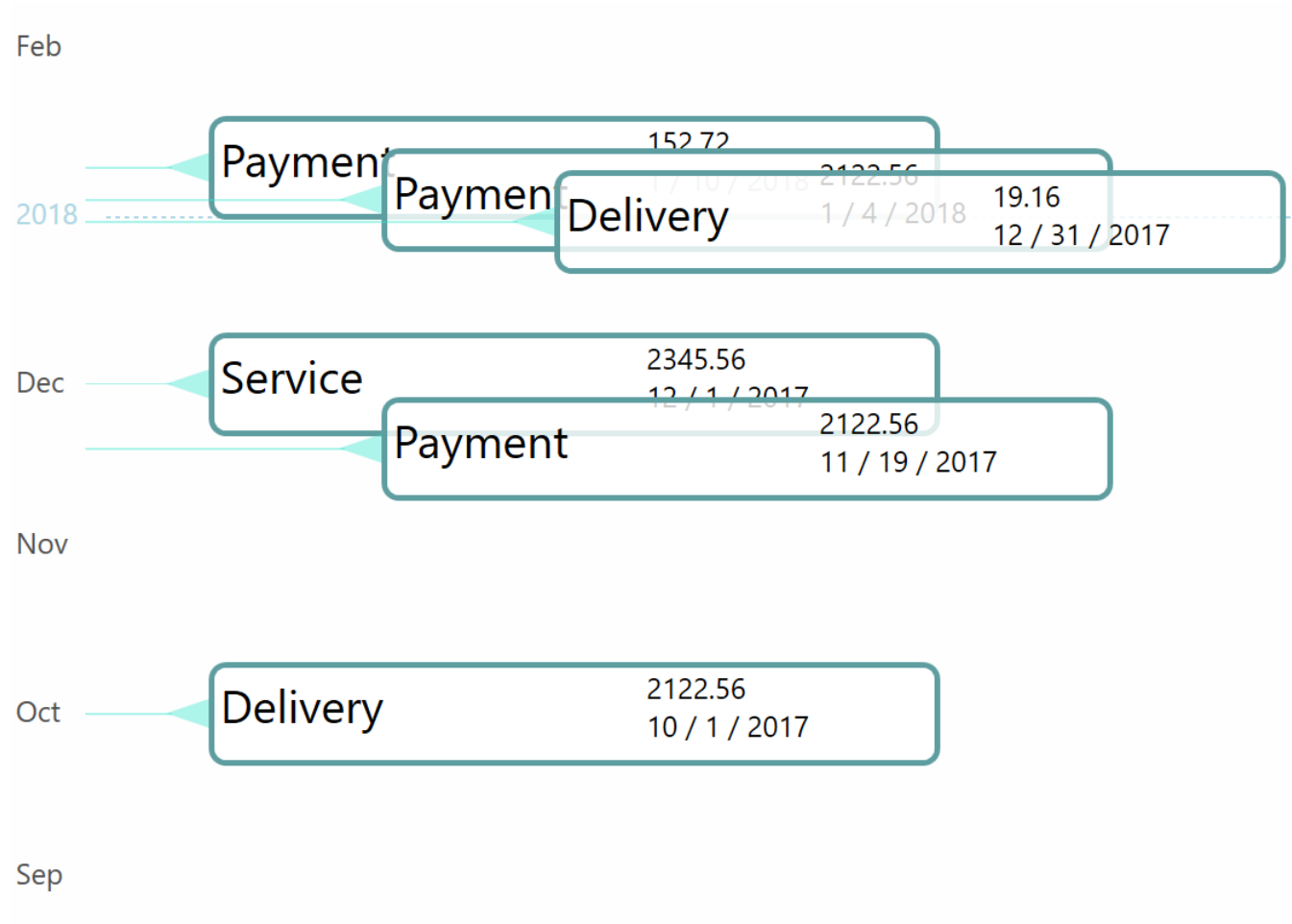


User Experience design becomes more important

With much more data to consume, good UX design is essential to consume it quickly and effectively



Event date	Description	Amount	
1/10/2018	Payment	\$152.72	
1/4/2018	Payment	\$2,122.56	
12/31/2017	Delivery	\$19.16	
12/1/2017	Service	\$2,345.56	
11/19/2017	Payment	\$2,122.56	
10/1/2017	Delivery	\$2,122.56	
1/10/2016	Payment	\$1,934.56	
10/1/2015	Service	\$1,934.56	



Using AI in business apps

- Image analysis and recognition
- Facial recognition
- Language translation
- Detecting customer dissatisfaction or unprofitable customers
- Calculation of risk
- Shopping recommendations
- Fault prediction and maintenance management

Commercial AI still in early stages

We are still learning the pitfalls and limitations of AI

For example, early image recognition intended to detect nudity gave false positive for sand dunes



Building your own AI

- Given enough data, many real world problems can be attacked by building a custom AI for the problem
 - Typical example – detecting objects shown in a video stream
- The cloud facilitates *machine learning* (ML), which builds AI models for various tasks
- Consuming the model is easy
 - That's what you're doing when you use packaged AI services
- Creating the model is hard, and requires many skills

What makes it hard?

- Much of the work is done in Python or R – languages you probably don't know
- It will take you weeks or months to learn to do this if you focus on it, months or years if you are working on it in your spare time
 - A serious math background helps – statistics, linear algebra, etc. are somewhere between useful and essential
- There are multiple techniques for different kinds of AI problems
- There are many libraries that may be needed to attack a real problem
- If you want to learn more, there's a Reddit thread:
<http://bit.ly/AIReddit>

AI needs balance between cloud and edge

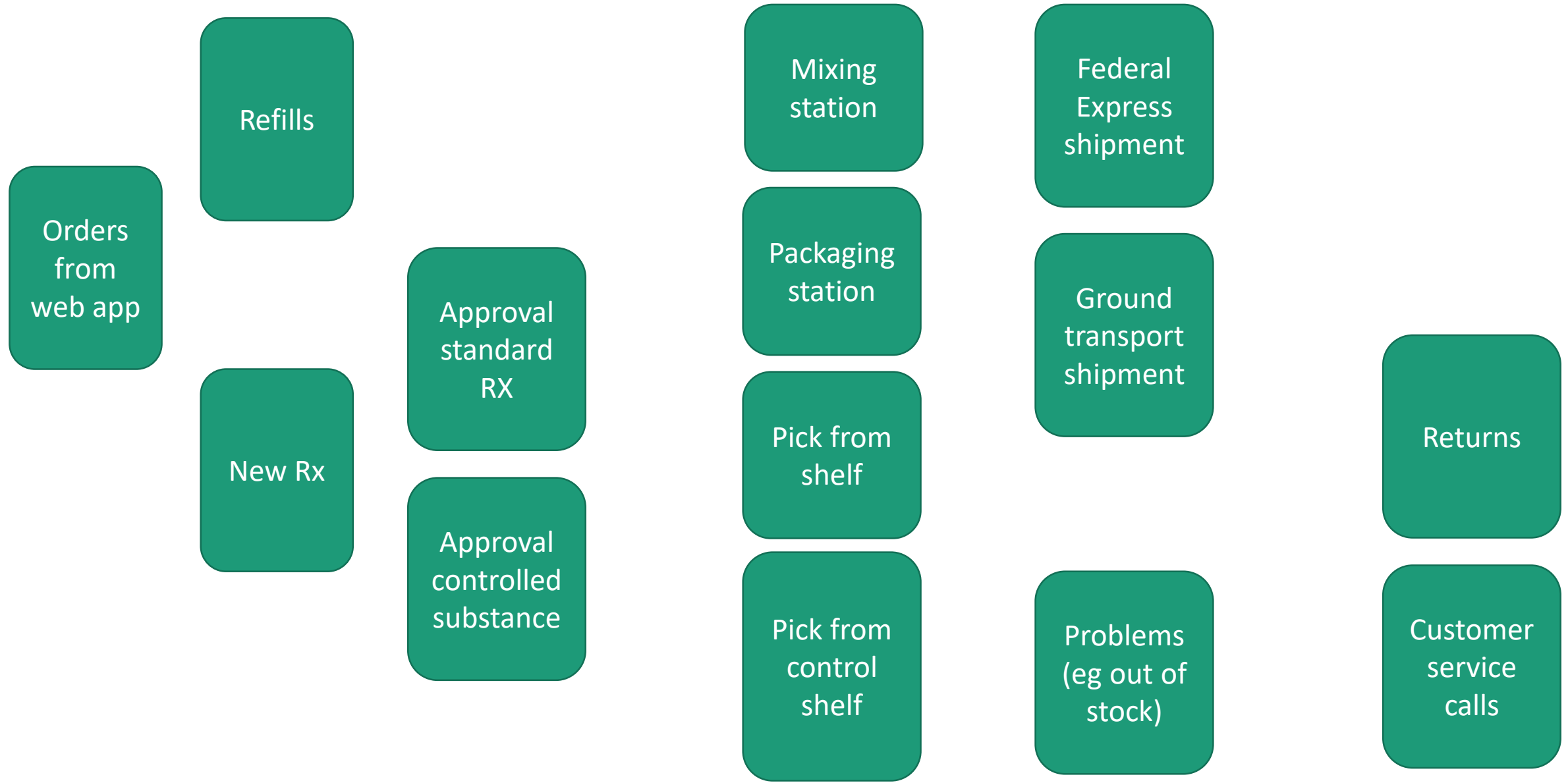
- Deriving AI models using machine learning is computation intensive, and likely to be done in the cloud
- Running the models can be done in the cloud for web apps
- For native apps requiring low latency or with limited bandwidth, models can be run on the local device
 - Example: object detection on a Hololens



Cloud-based engines

- Besides AI services and engines, other engines with more traditional logic become more feasible with the cloud
 - Tax calculation
 - Medical diagnosis
 - Drug interactions
 - Malware detection
 - Block spam calls
 - Data validation of various kinds

Let's drill in on workflow as an example



This could easily be extended to distributed workflow in the cloud

- Create REST API
- Use Azure Message Bus or Signal R to transport workflow events
- Someone will make a billion dollars with workflow in the cloud
- Based on past performance, I don't think it will be Microsoft
- There are some limited or proprietary systems in the cloud now, but I see the need for something more general

So how do we manage all this complexity?

- 1990: Windows development was expensive and complex to develop with C++ and Windows API
- In 1991 came Visual Basic – 8 years later it was the most popular development tool on the planet
 - It hid the complexity of Windows
- I expect something like that to happen now, because the complexity now is way more crushing than it was in 1990
- It doesn't mean traditional code-centric development goes away, but it will probably be supplemented by something faster and more approachable

Putting all the pieces together: A new concept for developing cloud apps

- Our existing development tools are not ideally suited for cloud-based apps
 - They are optimized for writing code, not for assembling pieces
- We need automatically generated UI, with layout determined via a metadata markup
- We need tools that are more approachable by non-professional developers and by those just starting as developers
- We are seeing the early, primitive versions of such a tool now

Current attempts at a new tool

Google Flutter

- Native for iOS and Android
- Composition based design (“everything’s a widget”)
- Mobile only
- Intended for people who are quite comfortable writing code

Microsoft PowerApps

- Very easy for non-developers
- Web interface only
- Data philosophy tied too closely to Microsoft Dynamics
- Formulas in Excel format

Others: Uno Platform, Zoho Creator, Kony AppPlatform, new ones appearing regularly

Where will we end up?

- My vision is a development environment created from the ground up for cloud-based apps
- It would generate UI automatically for web and devices, based on generalized markup that was mostly filled in via templates
- It would integrate cloud capabilities transparently
- It would be compositional – putting pieces together would be a primary activity
- A subset of functionality could be used with little or no code
- Coding would be in isolated segments associated with small pieces

App starts with data, either defined or imported

Traditional DB schema

- Data type
- Field name
- Nullable flag
- Etc

Plus metadata for app generation

- Validation rules (with states such as initial, final)
- Label to use in UI
- Order/priority in UI
- Role required to access
- Etc.

Starter kit schemas for common business entities available to choose

UI layout metadata - Vertical stacks, panels, etc.

Compositional designer to let user assemble the app by plugging pieces together and writing small amounts of glue code. (works with everything in diagram)

Engine for generating user screens (data and actions)

User-written components that access engine APIs – minimal code

Web UI

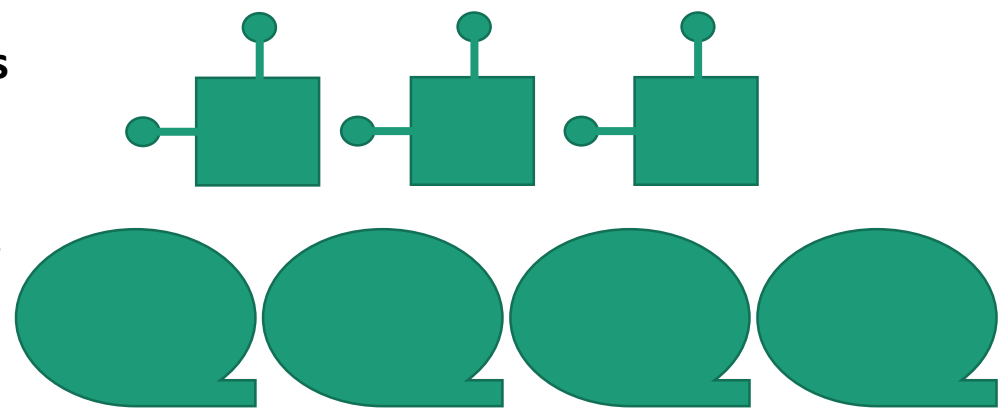
iOS app

Android app

Windows 10 app

Cloud engines

- Voice
- Photo recog
- Route optimiz
- Pricing
- Workflow
- Taxes, etc.



Will this really happen?

- While I can't say my vision will end up close to reality, I do think we will see a radical change in the way software is development at some point in the evolution of the cloud
- So you should be ready to leverage that when it happens
 - Don't blindly stick to the way you've always developed apps

Recap – changes for you

- VMs for development
- New capabilities with cloud APIs
- Less plumbing / deployment code to write
- More data, but less data entry – more attention to visualizing data
- AI and cloud engines allow dramatically new functionality
 - Some kinds of AI will drive execution towards the edge
- Current crushing complexity plus need to accommodate new APIs will likely mean new and different development environments
 - Probably start as an entry point for simple apps, with early versions of this now available
 - Possibly a new environment for more advanced and professional development

- User interface design and prototyping
- Corporate design facilitation and strategy
- Native app development – UWP/WPF/XAML
- Training on user experience design
- Training on XAML
 - Windows 10 / WPF / mobile / touch
 - Beginning through advanced

www.nextver.com for samples and videos

billyhollis@live.com or billyhollis@gmail.com

Billy Hollis



