



# 欢迎参与微软下一代人机互动 HACKATHON

## More Personal Computing

Cortana/Windows Hello/Ink/Kinect/Cognitive Services

# 猜电影

片段1



# 无密码的未来世界

- Microsoft Passport和Windows Hello实现安全和便利的结合

余泽鹏 Zepeng She 技术顾问

微软中国开发体验与合作事业部



# Microsoft Passport以及Windows Hello 方便的多因素认证

## Microsoft Passport

企业级的双因素认证

Device + biometric或者PIN

## Windows Hello

生物识别鉴权技术，为终端用户而生

支持人脸识别，虹膜识别以及指纹识别



A close-up photograph of a person's brown eye, looking directly at the camera. The lighting is dramatic, highlighting the iris and eyelashes.

# Windows Hello 带动全新身份认证的革命

- ✓ **Windows Hello** - 系统级的生物特征认证技术
  - 脸部, 虹膜, 指纹
- ✓ 一致用户注册 , 验证体验

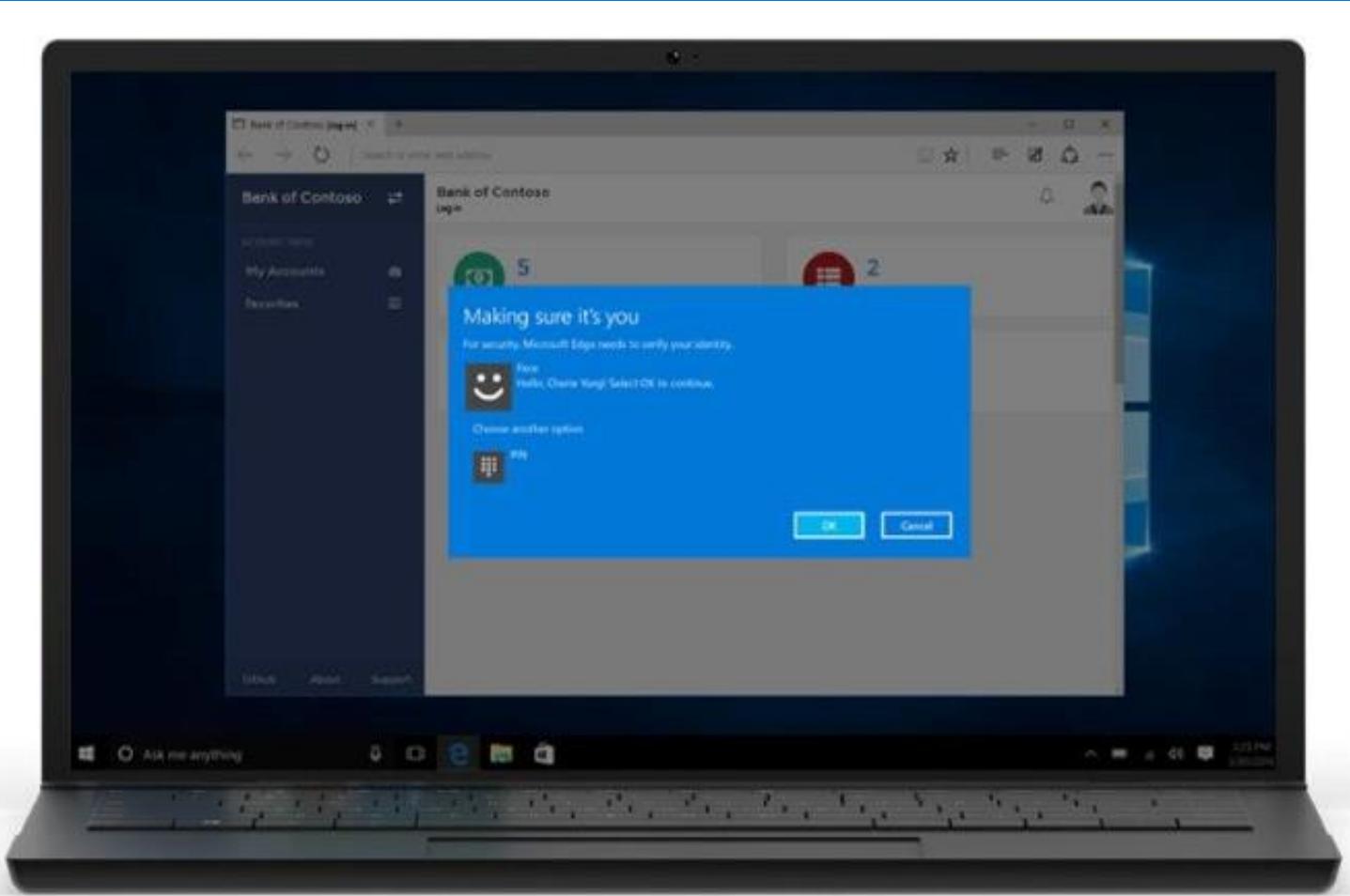


- ✓ 无可挑剔安全性 - 企业级的安全级别
  - 集成防止物理攻击的全新策略
  - 超越现代安全标准

# 为什么一个4位数字的PIN码更加安全？

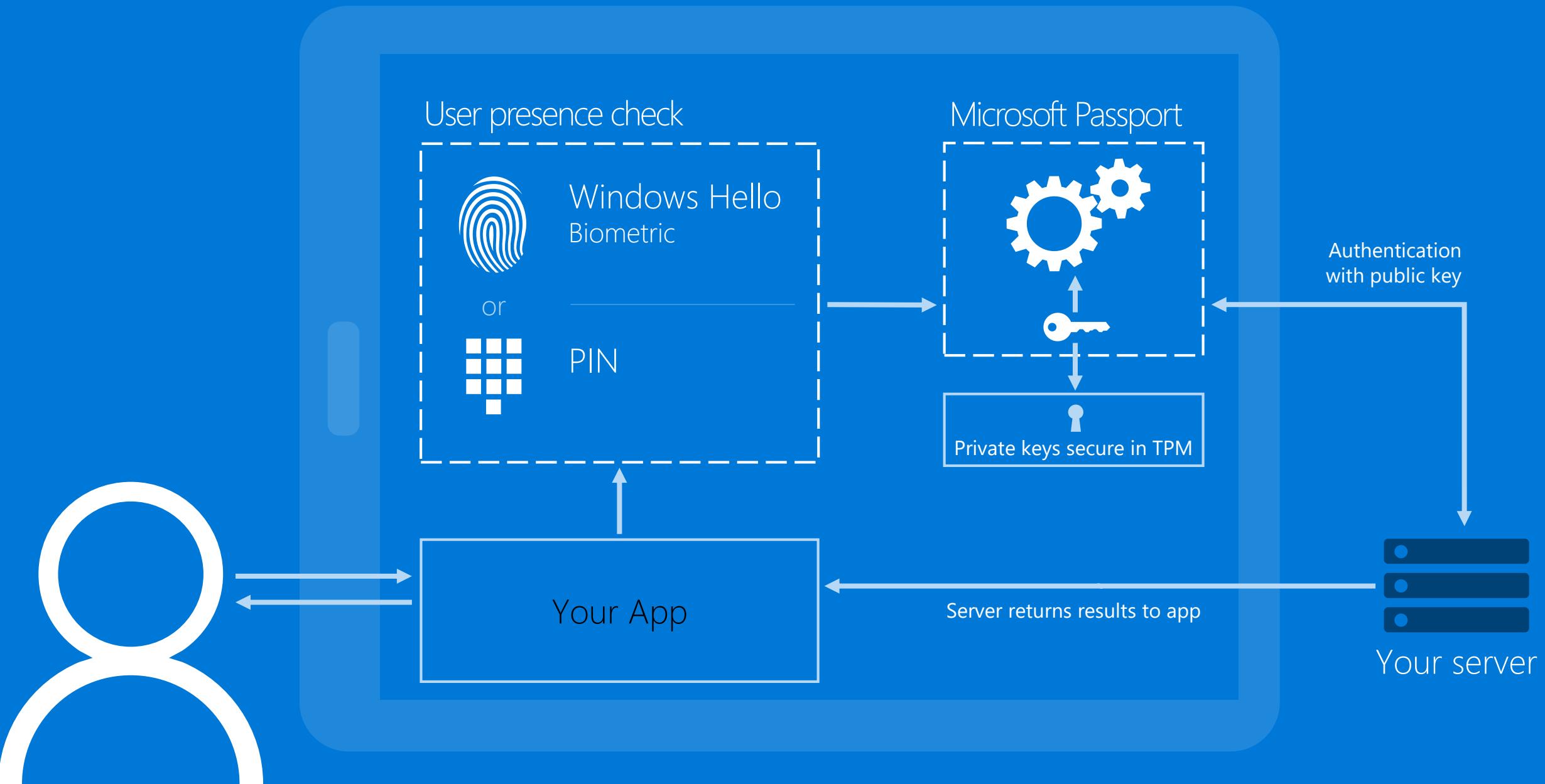
- 网络黑客需要知道你PIN码，而且需要持有你的设备
- TPM提供了反恶意攻击的技术来阻止线下的恶意破解
- 硬件绑定Key无法被窃取或者重放
- PIN码不会存储在设备上，也不会被发送到服务器

# 集成 Microsoft Passport & Windows Hello

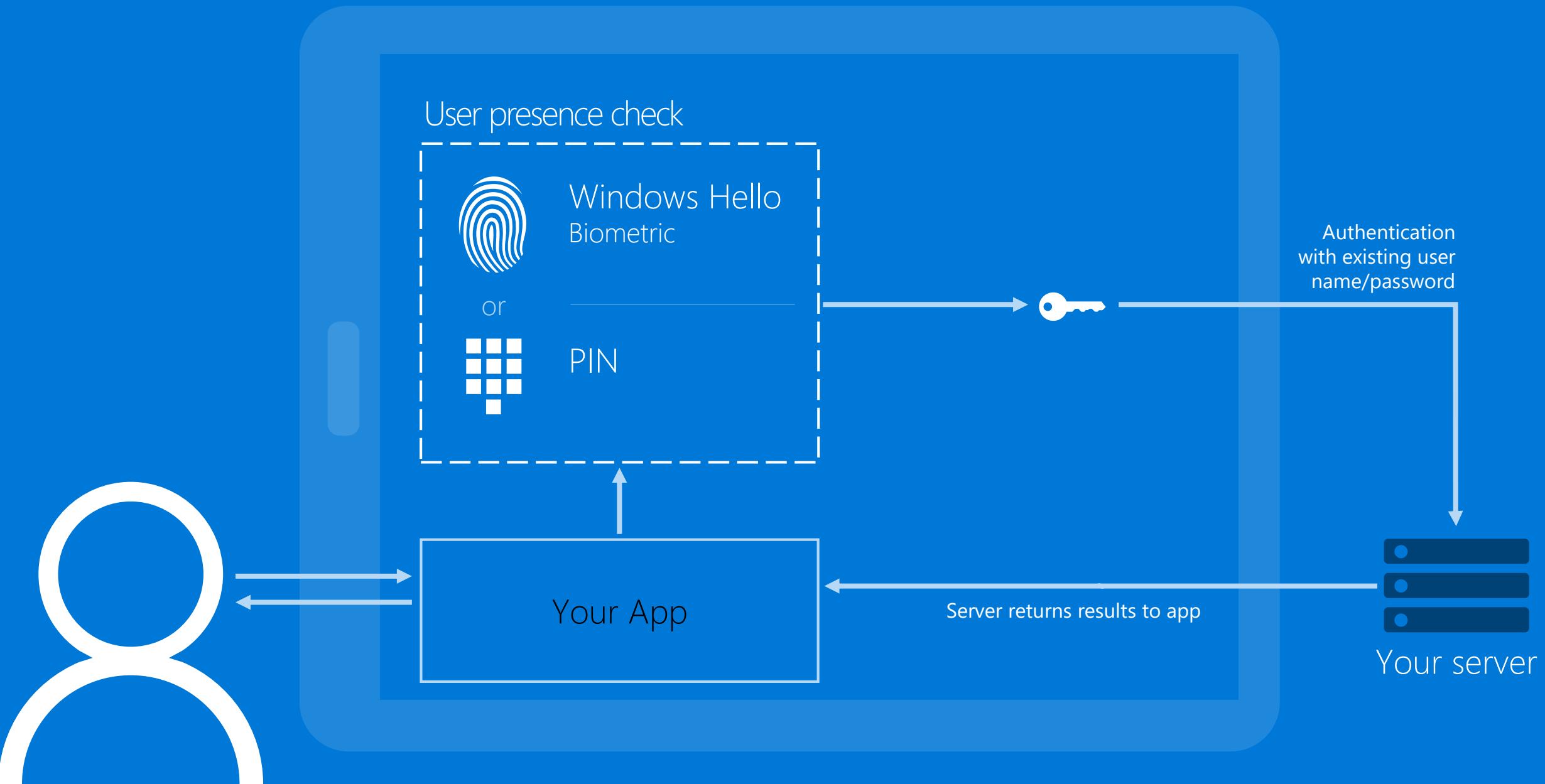


Edge-friendly websites

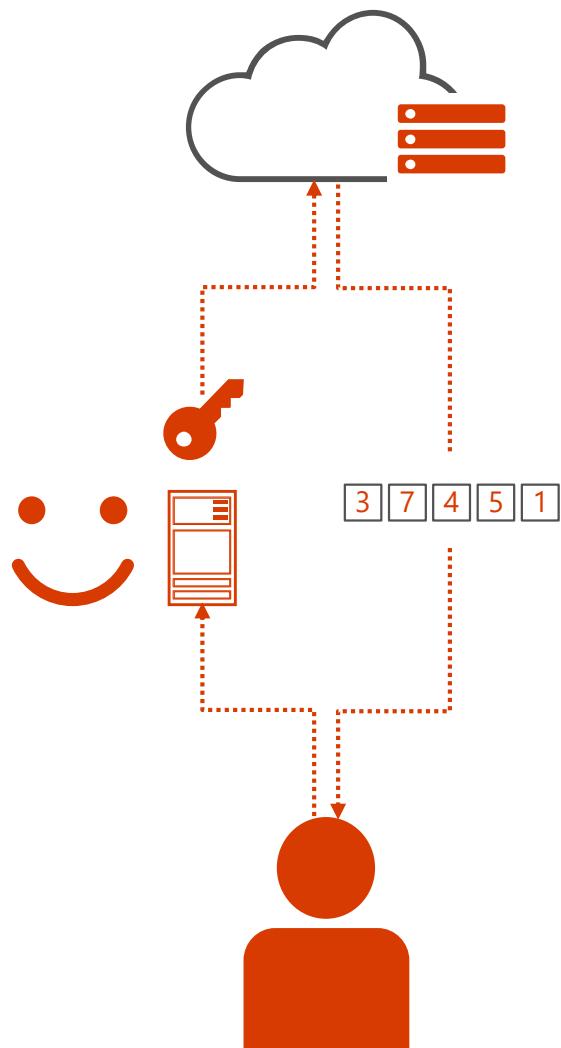
# 完整地集成 Microsoft Passport & Windows Hello



# 使用Windows Hello进行用户校验



# Microsoft Passport 鉴权



1. 应用启动并且从服务器端请求数据
2. 服务器要求用户进行鉴权并且向用户发送一个 challenge消息
3. 应用需要拿自己的私钥把challenge消息进行签名

```
var signResult = await  
userKey.RequestSignAsync(message);
```
4. 用户会被提示要求输入PIN码或者生物识别信息
5. Challenge消息被成功签名并且发送回服务器端
6. 服务器使用用户的public key验证该签名
7. 如果签名正确，那么授权该用户并且返回数据

# 微软认知服务 (前“牛津计划”)

微软中国开发体验与平台推广部  
刘士君, 平台技术顾问





## 理解您应用中的数据

认知服务的API帮助您理解并与包括语音，文字，图片，及视频数据进行交互



## 强大的模型

认知服务的模型充分利用了构建微软多项产品线的同样的深度学习与机器学习技术



## 容易使用

认知服务让您可以跨平台轻松调用这些  
REST API，使您可以更加专注您的应用

# 微软认知服务概览

视觉	语音	语言	知识	搜索
计算机视觉API	自定义智能语音识别服务	必应拼写检查API	学术搜索API	必应网络搜索API
情感识别API	声纹识别API	语言理解智能服务	实体链接智能服务	必应视频搜索API
人脸识别API	语音识别API	文本分析API	推荐API	必应图片搜索API
视频检测API		网络级语言模型API	知识搜索服务	必应新闻搜索API
		语言分析API		必应自动建议API



# 人脸识别 API



## 检测

```
"faceRectangle": {"width": 193, "height": 193, "left": 326, "top": 204}
```

...

## 属性特征 : 27个

```
"attributes": { "age": 42, "gender": "male",  
"headPose": { "roll": "8.2", "yaw": "-37.8", "pitch": "0.0" }}
```

## 分组



## 辨识

Jasper Williams



# 人脸识别演示



# 人脸识别 API

## 检测

必要的步骤，获得FaceID

## 查找相似/分组/验证

输入不同数量的FaceID

## 认证

建组 -> 建人 -> 加人脸 -> 训练 -> 认证

## 持久化的FaceList

在选定区域内识别人脸，持久化，仅用于相似查找



# 情绪识别 API



## 人脸检测

```
"faceRectangle": {"width": 193, "height": 193, "left": 326, "top": 204}
```

...

## 情感指数

```
"scores": { "anger": 5.182241e-8,  
           "contempt": 0.0000242813,  
           "disgust": 5.621025e-7,  
           "fear": 0.00115027453,  
           "happiness": 1.06114619e-8,  
           "neutral": 0.003540177,  
           "sadness": 9.30888746e-7,  
           "surprise": 0.9952837}
```



# 情绪识别演示



# Kinect for Windows

## Face API

Lilith Liu | 刘颖  
[liliu@microsoft.com](mailto:liliu@microsoft.com)



Microsoft

# Kinect Fusion

Provides a rich mesh reconstruction of environment and objects

Generate a 3D model of an object or person in real time

Surface reconstruction done using voxels (=volume + pixel)



## 1 Process Function

- Locates the camera in 3D space
- Can be rendered and used to begin the Fusion

## 3 Other considerations

- Can enable color to be lifted from the scan
- Experimental pose finders (another 6DoF tracker) also included to play with

## 2 Create Reconstruction Function

- Pass in a set of parameters defining the environment and processing type
- Outputs the reconstruction

## 4 Flexibility to

- Customize elements of the Fusion data pipeline
- Start with a raw depth map and do your own calculations to align camera and scan, calculate voxels and create mesh

# Face APIs: 2D face model

Offers a great entry-point into easy to use face attributes

Operations and outputs are returned in 2D space



## Detection

- Outputs a bounding box around the face
- Can be visualized in color or IR

## Orientation

- Returns a quaternion (4-D vector) of the head joint with respect to the sensor
- Quaternion prevents gimbal lock

## Alignment

- Identifies 5 facial landmarks on the face
- Operation performed in color or IR

## Expressions

- Provides classifiers for happy, left / right eye open, engagement, mouth open and mouth moving

# Microsoft.Kinect.Face



<http://kinect.github.io/tutorial/lab08/index.html>

```
public MainPage()
{
    this.multiSourceFrameReader.MultiSourceFrameArrived += this.Reader_MultiSourceFrameArrived;

    // specify the required face frame results
    // init with all the features so they are accessible later.
    this.faceFrameFeatures =
        FaceFrameFeatures.BoundingBoxInColorSpace
        | FaceFrameFeatures.PointsInColorSpace
        | FaceFrameFeatures.BoundingBoxInInfraredSpace
        | FaceFrameFeatures.PointsInInfraredSpace
        | FaceFrameFeatures.RotationOrientation
        | FaceFrameFeatures.FaceEngagement
        | FaceFrameFeatures.Glasses
        | FaceFrameFeatures.Happy
        | FaceFrameFeatures.LeftEyeClosed
        | FaceFrameFeatures.RightEyeClosed
        | FaceFrameFeatures.LookingAway
        | FaceFrameFeatures.MouthMoved
        | FaceFrameFeatures.MouthOpen;

    this.faceManager = new FaceManager(
        this.kinectSensor,
        this.faceFrameFeatures);

    // set IsAvailablechanged event notifier
    this.kinectSensor.IsAvailableChanged +=
        this.Sensor_IsAvailableChanged;
    //...
    this.InitializeComponent();

    this.Loaded += MainPage_Loaded;
}

void MainPage_Loaded(object sender, RoutedEventArgs e)
{
    SetupCurrentDisplay(DEFAULT_DISPLAYFRAMETYPE);
}
```

# HD Face: 3D face model

Provides a high-definition mesh of a face

Reconstruction is done by deforming “shape units”

Operations and outputs are returned in 3D space



1

## Face Model Builder

- Captures the face data
- Interactive API
- Provides collections status and evaluation of frames collected

2

## Face Model

- Set of 94 shape units which represent scale, hair color and skin color

3

## Face Mesh

- The face mesh can be used in most animation applications
- Mesh topology is the same for all faces represented in a standard way (numTriangles, numVertices, etc.)

# HD Face (Microsoft.Kinect.Face)



```
private void FaceReader_FrameArrived(object sender, HighDefinitionFaceFrameArrivedEventArgs e)
{
    using (var frame = e.FrameReference.AcquireFrame())
    {
        if (frame != null && frame.IsFaceTracked)
        {
            frame.GetAndRefreshFaceAlignmentResult(_faceAlignment);
            UpdateFacePoints();
        }
    }
}

private void UpdateFacePoints()
{
    if (_faceModel == null) return;

    var vertices = _faceModel.CalculateVerticesForAlignment(_faceAlignment);
}
```

<https://github.com/Vangos/kinect-2-face-hd>

# Windows Hello with the Kinect v2

1. The first step is to opt-in to driver flighting. You can follow the instructions [here](#) to set up your registry by hand, or you can use the following text to create a .reg file to right-click and import the settings:

Windows Registry Editor Version 5.00

[HKEY\_LOCAL\_MACHINE\SOFTWARE\Microsoft\DriverFlighting\Partner]

"TargetRing"="Drivers"

2. Next, you can use Device Manager to update to the preview version of the Kinect driver and runtime:

1. Open Device Manager (Windows key + x, then m).
2. Expand "Kinect sensor devices".
3. Right-click on "WDF KinectSensor Interface 0".
4. Click "Update Driver Software..."
5. Click "Search automatically for updated driver software".
6. Allow it to download and install the new driver.
7. Reboot.

<https://channel9.msdn.com/coding4fun/kinect/Windows-Hello-with-the-Kinect-v2>

# Kinect Resources

Kinect for Windows website <http://kinectforwindows.com>

Download the V2 SDK <http://aka.ms/K4WSDK>

Purchase a V2 Sensor <http://aka.ms/OrderK4W>

Kinect for Windows Virtual Academy Training  
<http://channel9.msdn.com/Series/Programming-Kinect-for-Windows-v2>

Developer Forums <http://aka.ms/k4wv2forum>

Twitter @KinectWindows

# 猜电影

## 片段2



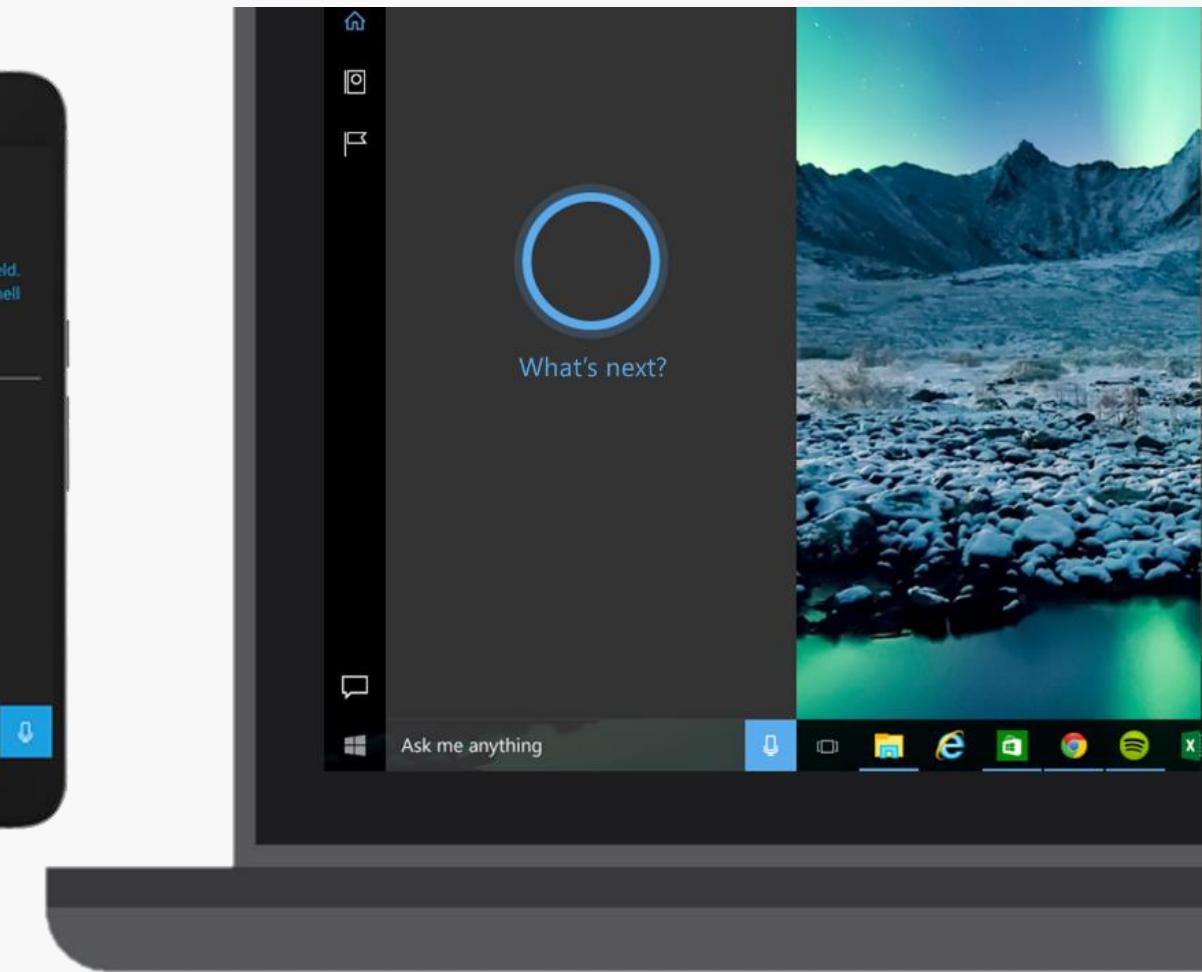
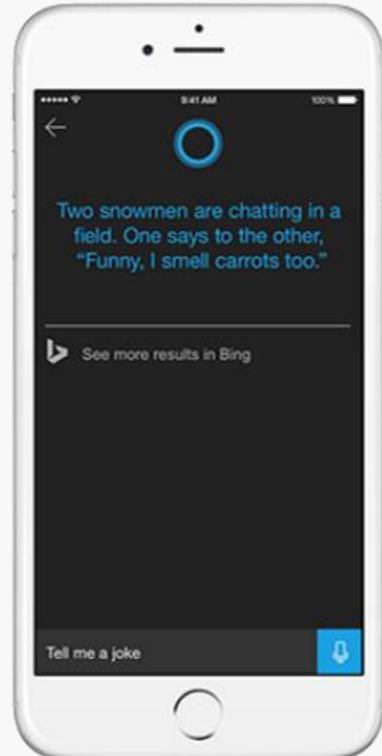
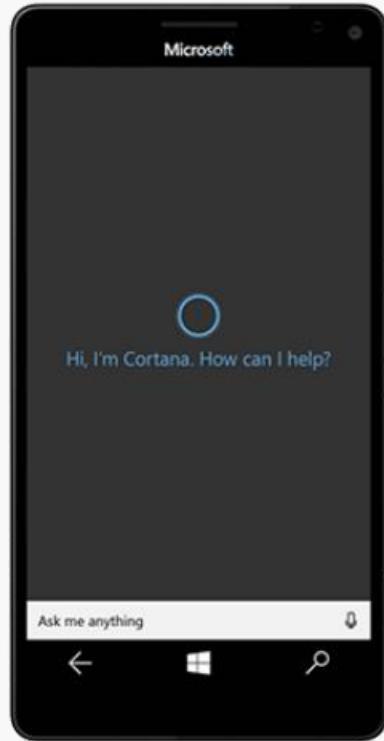


# MPC: Cortana

丁煜恒  
微软开发者体验与平台合作事业部  
平台技术顾问

# Cortana—跨平台的个人助手，如你所愿

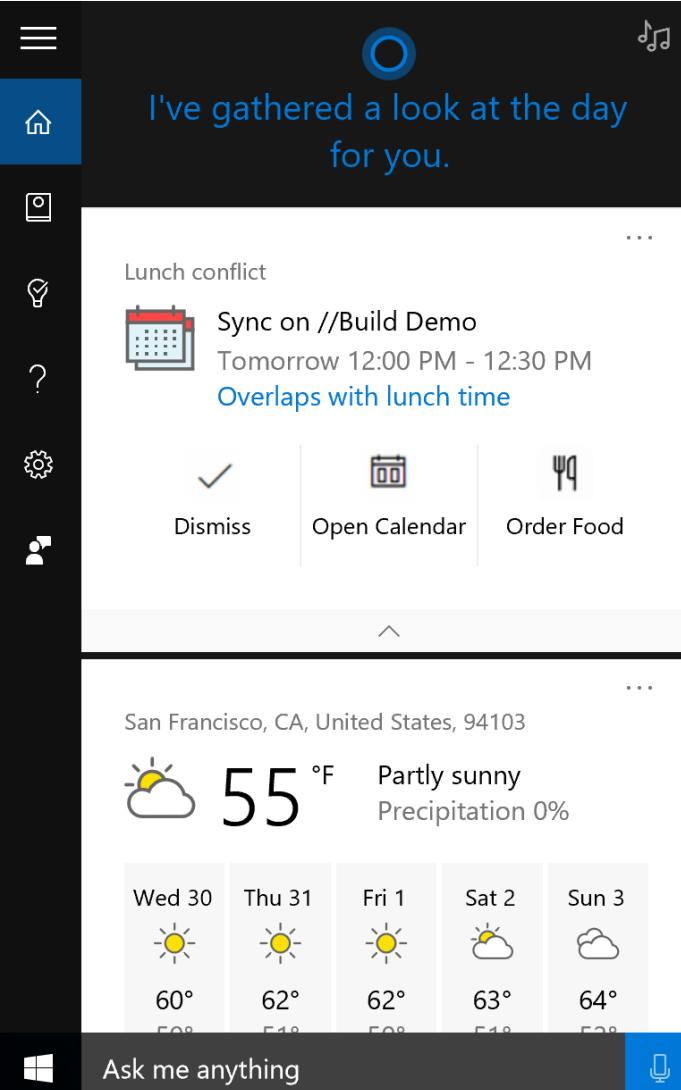
Cortana is available on Windows 10 desktops, tablets, and phones – as well as iOS, Android and Cyanogen phones.



# Cortana 设计准则：

Personal & trusted	Keeps one shared notebook with what she's learned about me; I can access the notebook from any Cortana experience and Cortana is always aware of its contents
Productive	Helps me get things done by completing tasks or bringing me a step closer
Natural interaction	Lets me interact using natural language in a consistent and contextual way
Proactive	Always looks out for me (e.g. using contextual awareness and her notebook)
Personality	Consistent visual identity (O), personality traits, voice (e.g. Jen Taylor in the US); Adapts continually to be globally inclusive and culturally relevant

# Cortana 更新: Proactive Actions



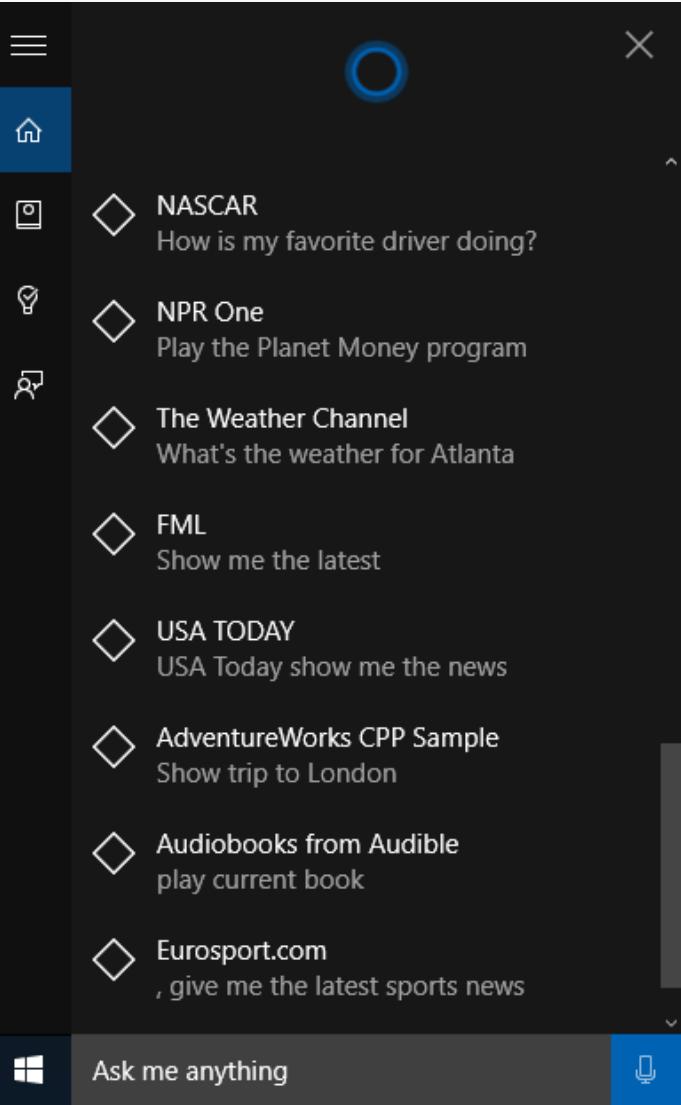
Proactively suggests actions

to users, at just the right time, to drive usage of your apps and websites

Easy to register actions  
in Cortana portal

Works everywhere Cortana is  
starting with Windows 10 and Android

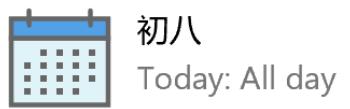
# 通用应用(UWP)的语音命令



- Apps can register voice commands for users to say to Cortana
- Cortana launches the app to complete the task
- App can use Cortana's canvas and voice to present feedback to the user
- Available for Universal Windows Apps built with XAML and JavaScript



Need anything, YH?



Microsoft Corp (NASDAQ: MSFT)

45.66 ▼ 0.95 (2.04%)

SHANGHAI COMPOSITE: Shanghai SE Composite

3,601.24 ▼ 62.88 (1.72%)

Data from Morningstar

Popular now

你好 小娜， 提请我在  
xx 点钟 做做XX

你好 小娜， 给 Jonas  
发邮件

你好 小娜， 给 Jonas  
打电话

你好 小娜， 给 Jonas  
发送消息

|Ask me anything



# 前台语音命令架构





Need anything, YH?



Microsoft Corp (NASDAQ: MSFT)

45.66 ▼ 0.95 (2.04%)

SHANGHAI COMPOSITE: Shanghai SE Composite

3,601.24 ▼ 62.88 (1.72%)

Data from Morningstar

Popular now

你好 小娜， 帮我在大麦上搜索周杰伦的演唱会

你好 小娜，在爱奇艺上搜索暴走大事件

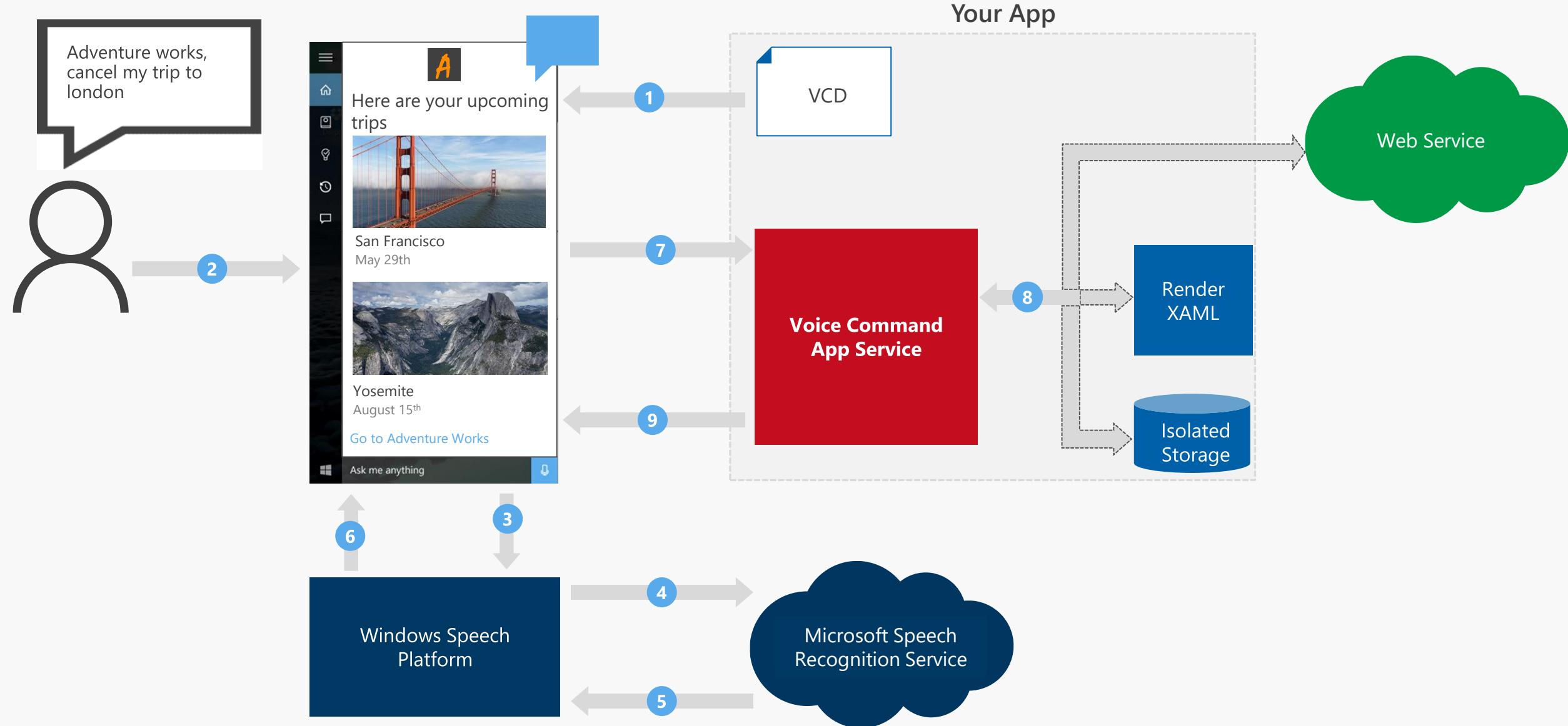
你好 小娜，发微信给豆豆

Next ?

Ask me anything



# 后台语音命令架构



Cortana is your **truly personal** digital assistant working  
across all your devices to help you **get things done**.

- 奇幻旅行相关代码及白皮书：  
[https://github.com/DXChinaTE/More-Personal-Computing/tree/master/MPC Cortana Whitepaper](https://github.com/DXChinaTE/More-Personal-Computing/tree/master/MPC_Cortana_Whitepaper)
- Join the Cortana actions developer preview：  
<https://developer.microsoft.com/en-US/cortana/signup/showform>
- UWP 应用中的 Cortana 交互：  
<https://msdn.microsoft.com/zh-cn/windows/uwp/input-and-devices/cortana-interactions>
- Cortana 设计指南：  
<https://msdn.microsoft.com/zh-cn/windows/uwp/input-and-devices/cortana-design-guidelines>
- 在 Cortana 中支持自然语言形式的语音命令：  
<https://msdn.microsoft.com/zh-cn/windows/uwp/input-and-devices/support-natural-language-voice-commands-in-cortana>

# 语音控制的未来世界

—— Windows 10 Speech Platform

余泽鹏  
技术顾问  
微软中国开发体验与合作事业部



# In-App Speech Integration

Speech Synthesis  
(aka Text-to-Speech)



# Speech Synthesis (aka Text-to-Speech)

- Speak with default speech settings
- Speak with any installed speech language
- Customize TTS voice with Speech Synthesis Markup Language (SSML)
- New since Windows & WP 8.1: Audio streams!
- Supported Languages in WP 8.1 & Windows 10
  - English (US, UK, India)
  - German
  - Spanish (Spain, Mexico)
  - French
  - Italian
  - Polish
  - Portuguese
  - Russian
  - Japanese
  - Chinese (Traditional, Simplified)



# In Application Speech Synthesis

```
// Universal Windows 10, Windows & Windows Phone 8.1 Store App (UWP/WinRT)

// Speech Synthesis
<!--MediaElement in xaml file-->
<MediaElement Name="audioPlayer" AutoPlay="True" .../>

// C# code behind
// Function to speak a text string
private async void SpeakText(MediaElement audioPlayer, string textToSpeak)
{
    SpeechSynthesizer synthesizer = new SpeechSynthesizer();

    SpeechSynthesisStream ttsStream = await synthesizer.SynthesizeTextToStreamAsync(textToSpeak);

    audioPlayer.SetSource(ttsStream, ""); // This starts the player because AutoPlay="True"
}
```



# In Application Speech Synthesis

```
// Universal Windows 10, Windows & Windows Phone 8.1 Store App (UWP/WinRT)
```

```
// Speech Synthesis
<!--MediaElement in xaml file-->
<MediaElement Name="audioPlayer" AutoPlay="True" ... />
```

```
// Synthesis
<!--MediaElement in xaml file-->
<MediaElement Name="audioPlayer" AutoPlay="True" ... />
```

Speech synthesis uses a MediaElement control to speak text

```
SpeechSynthesizer tts = new SpeechSynthesizer();
SpeechSynthesizerResult result = await synthesizer.SynthesizeTextToStreamAsync(textToSpeak);

audioPlayer.SetSource(ttsStream, ""); // This starts the player because AutoPlay="True"
}
```



```
// C# code behind
// Function to speak a text string
private async void SpeakText(MediaElement audioPlayer, string textToSpeak)
{
    SpeechSynthesizer synthesizer = new SpeechSynthesizer();

    SpeechSynthesisStream ttsStream
        = await synthesizer.SynthesizeTextToStreamAsync(textToSpeak);

    audioPlayer.SetSource(ttsStream, "");
    // This starts the player because AutoPlay="True"
}
```

Speak text by setting the stream as source on the MediaElement

```
audioPlayer.SetSource(ttsStream, ""); // This starts the player because AutoPlay="True"
}
```

# In-App Speech Integration

Speech Recognition &  
Grammars





# In-App Speech Recognition

- Check if the interaction was input through voice or text
- For voice, use Speech to continue the user interaction in the app
- Use `Windows.Media.SpeechRecognition` for speech recognition
- Use `Windows.Media.SpeechSynthesis` APIs to reply/talk to the user



# On-Demand Speech Recognition

- Use `SpeechRecognizer.RecognizeWithUIAsync` to present the user with System listening GUI that can be customized
- Use `SpeechRecognizer.RecognizeAsync` for on demand recognition and provide a listening UI experience
- System predefined grammars: Dictation, Web Search, Form Filling
- App grammars: a list of strings or SRGS file



# Continuous Speech Recognition

- `SpeechRecognizer.ContinuousRecognitionSession` enables listening continuously
- Can be used with Dictation grammar for scenarios where users speak for long periods
- Can be used with app-provided grammars for always-listening in-app commands

# Demo: Speech

In-App Speech Recognition & Grammars



# 语音API

语音识别 (语音转文本)  
将语音转换为文本

语音输出(文本转语音)  
将文本合成语音



# 语音识别演示



# 语音识别

## REST 服务

## 客户端SDK

支持平台

任意

Windows, Android, iOS

数据支持

支持

支持

麦克风支持

不支持

支持

麦克风沉默识别

不支持

支持

音频长度

短

短与长

响应数结果数量

n-best模式

多个部分结果，较短语音n-best模式以及较长语音的多短语模式

与微软的Xbox, Windows, Cortana采用相同技术



# 语音识别SDK

## 短模式

音频时长 < 15秒

最终结果 n-best模式

部分结果 支持

## 长模式

< 2分钟

Best模式并在语音结束时输出

支持



### \*\*\*\*\* 最终 N-BEST 结果 \*\*\*\*\*

- [0] Confidence=Normal Text="450 six St San Francisco."
- [1] Confidence=Normal Text="For 50 six St San Francisco."
- [2] Confidence=Normal Text="456th St San Francisco."
- [3] Confidence=Normal Text="450 six St in San Francisco."
- [4] Confidence=Normal Text="456 St San Francisco."



# 语音输出



通过POST请求将文本合成为语音  
最长支持15秒  
支持17种语言

```
<speak version="1.0"  
xmlns="http://www.w3.org/2001/10/synthesis"  
xmlns:mstts="http://www.w3.org/2001/mstts"  
xml:lang="en-US">  
<voice name="Microsoft Server Speech Text to Speech  
Voice (en-US, ZiraRUS)">
```

Synthesize audio from text, to speak to your users.

```
</voice></speak>
```

由文本合成为  
语音并传达给  
你的用户听



## 语言理解智能服务

理解用户所说的话  
使用来自Bing和Cortana的预构的模型或者  
自己创建的模型

# 语言理解智能服务

用交互的特征来减少贴标签的工作

使用可视化来测量和提升性能

与语音识别服务的无缝集成

几个样例就足以搭建自己的应用，并自主学习

定义概念

提供样例

部署

主动学习

# 语言理解智能服务



```
{  
  "entities": [  
    {  
      "entity": "flight_delays",  
      "type": "Topic"  
    }  
  ],  
  "intents": [  
    {  
      "intent": "FindNews",  
      "score": 0.99853384  
    },  
    {  
      "intent": "None",  
      "score": 0.07289317  
    },  
    {  
      "intent": "ReadNews",  
      "score": 0.0167122427  
    },  
    {  
      "intent": "ShareNews",  
      "score": 1.0919299E-06  
    }  
  ]  
}
```

# 猜电影

## 片段3

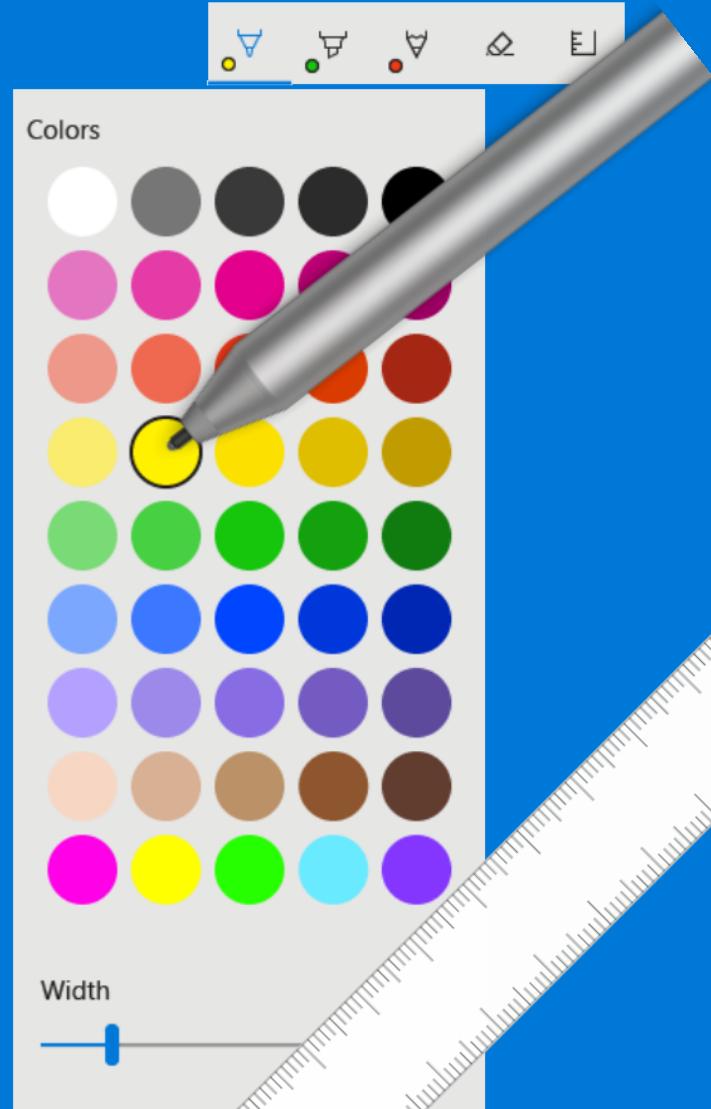




# 电子墨水 Windows Ink

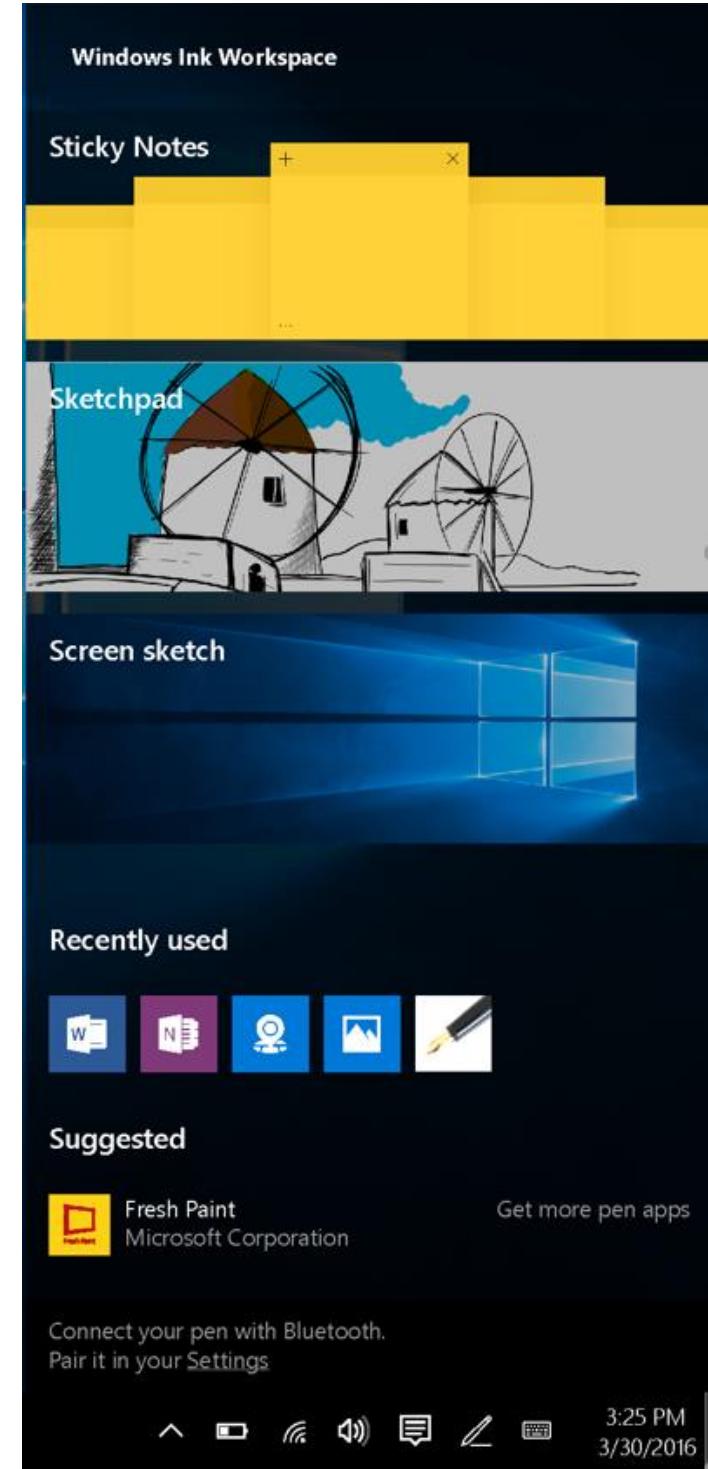
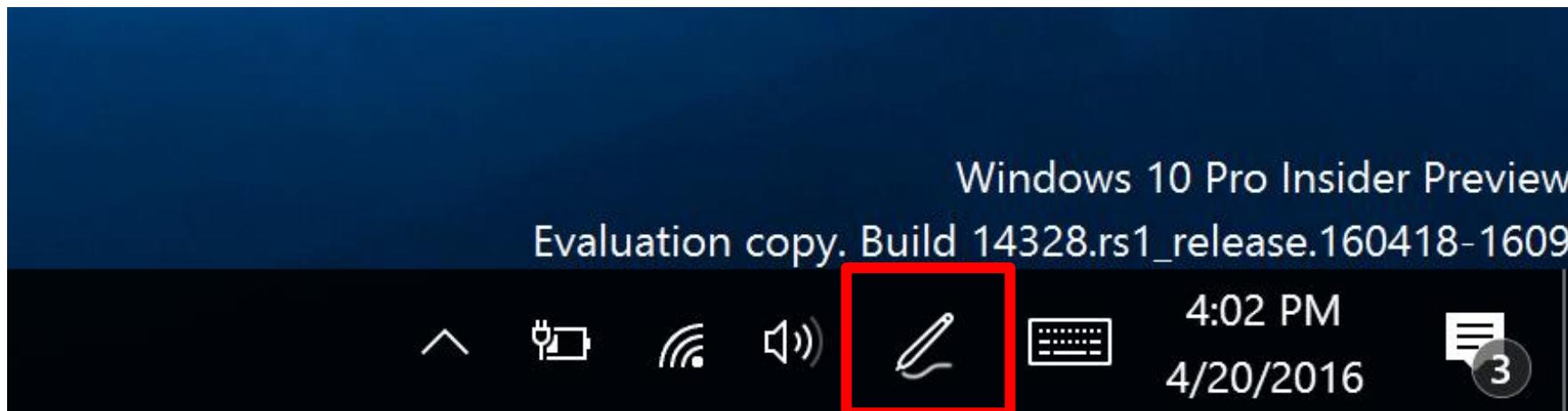
王博 技术顾问

微软中国开发体验与合作事业部

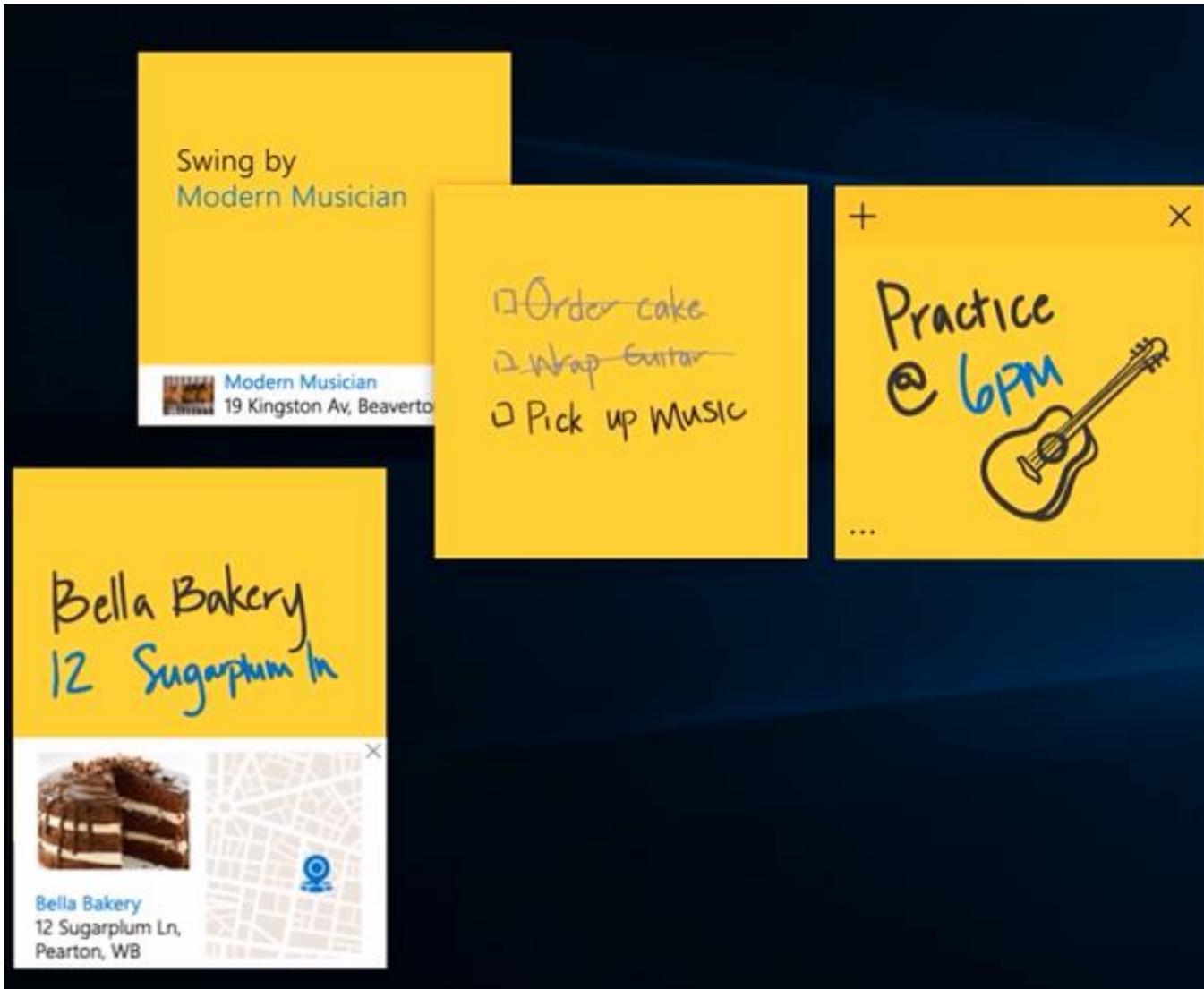


# Windows Ink Workspace

- Improved Sticky Notes
- Sketchpad
- Screen sketch



# Sticky Notes



# InkToolbar

Direct Ink: Low latency, high quality ink canvas

Easy: One line of code to add an ink experience

```
<InkCanvas x:Name="m_inkCanvas"/>
```

Consistent: One line of code to add ink tools\*

```
<InkToolbar TargetInkCanvas="{x:Bind m_inkCanvas}" />
```

\*If you did the above, you get the ruler for "free"

