



Reactive Extensions



Martin Konôpka
Microsoft Student Partner
PhD student @ FIIT STU

A scene from the movie Toy Story featuring Woody and Buzz Lightyear. Woody, on the left, is a cowboy doll with a yellow and red plaid shirt, a black and white cow-print vest, and a large gold belt buckle. He has a worried expression. Buzz Lightyear, on the right, is a space ranger action figure in his iconic green, white, and purple suit. He is gesturing with his right hand, showing three fingers. The background is a simple room with a door and a window with yellow stars.

DATA

DATA EVERYWHERE

```
[{ x: 322, y: 445 }, { x: 314, y: 442 },  
 { x: 313, y: 441 }, { x: 313, y: 437 },  
 { x: 315, y: 435 }, { x: 319, y: 432 },  
 { x: 333, y: 426 }, { x: 345, y: 423 },  
 { x: 375, y: 417 }, { x: 393, y: 414 },  
 { x: 413, y: 411 }, { x: 455, y: 407 },  
 { x: 478, y: 404 }, { x: 501, y: 402 },  
 { x: 548, y: 398 }, { x: 571, y: 397 },  
 { x: 593, y: 396 }, { x: 634, y: 397 },  
 { x: 652, y: 399 }, { x: 683, y: 404 },  
 { x: 696, y: 408 }, { x: 708, y: 412 },  
 { x: 728, y: 420 }, { x: 736, y: 425 },  
 { x: 744, y: 429 }, { x: 754, y: 437 },  
 { x: 955, y: 486 }, { x: 995, y: 486 }]
```

```
[{ title: "A Tale of Two Cities", year: 1998 },  
 { title: "Whatever the Case May Be", year: 2000 },  
 { title: "Some Like It Hoth", year: 2003 },  
 ...  
 { title: "Happily Ever After", year: 2010 }]
```

```
[{ x: 322, y: 445 }, { x: 314, y: 442 },  
 { x: 313, y: 441 }, { x: 313, y: 437 },  
 { x: 315, y: 435 }, { x: 319, y: 432 },  
 { x: 333, y: 426 }, { x: 345, y: 423 },  
 { x: 375, y: 417 }, { x: 393, y: 414 },  
 { x: 413, y: 411 }, { x: 455, y: 407 },  
 { x: 478, y: 404 }, { x: 501, y: 402 },  
 { x: 548, y: 398 }, { x: 571, y: 397 },  
 { x: 593, y: 396 }, { x: 634, y: 397 },  
 { x: 652, y: 399 }, { x: 683, y: 404 },  
 { x: 696, y: 408 }, { x: 708, y: 412 },  
 { x: 728, y: 420 }, { x: 736, y: 425 },  
 { x: 744, y: 429 }, { x: 754, y: 437 },  
 { x: 955, y: 486 }, { x: 995, y: 486 }]
```

```

var movies = await GetMoviesAsync();
for (int i = 0; i < movies.Count; i++)
{
    ...
}

```

```

[ { title: "A Tale of Two Cities", year: 1998 },
  { title: "Whatever the Case May Be", year: 2000 },
  { title: "Some Like It Hoth", year: 2003 },
  ...
  { title: "Happily Ever After", year: 2010 } ]

```

```

[ { x: 322, y: 445 }, { x: 314, y: 442 },
  { x: 313, y: 441 }, { x: 313, y: 437 },
  { x: 315, y: 435 }, { x: 319, y: 432 },
  { x: 333, y: 426 }, { x: 345, y: 423 },
  { x: 375, y: 417 }, { x: 393, y: 414 },
  { x: 413, y: 411 }, { x: 455, y: 407 },
  { x: 478, y: 404 }, { x: 501, y: 402 },
  { x: 548, y: 398 }, { x: 571, y: 397 },
  { x: 593, y: 396 }, { x: 634, y: 397 },
  { x: 652, y: 399 }, { x: 683, y: 404 },
  { x: 696, y: 408 }, { x: 708, y: 412 },
  { x: 728, y: 420 }, { x: 736, y: 425 },
  { x: 744, y: 429 }, { x: 754, y: 437 },
  { x: 955, y: 486 }, { x: 995, y: 486 } ]

```

```

var movies = await GetMoviesAsync();
for (int i = 0; i < movies.Count; i++)
{
    ...
    if (movies[i].Year > 2000)
    ...
}

```

```

[ { title: "A Tale of Two Cities", year: 1998 },
  { title: "Whatever the Case May Be", year: 2000 },
  { title: "Some Like It Hoth", year: 2003 },
  ...
  { title: "Happily Ever After", year: 2010 } ]

```

```

[ { x: 322, y: 445 }, { x: 314, y: 442 },
  { x: 313, y: 441 }, { x: 313, y: 437 },
  { x: 315, y: 435 }, { x: 319, y: 432 },
  { x: 333, y: 426 }, { x: 345, y: 423 },
  { x: 375, y: 417 }, { x: 393, y: 414 },
  { x: 413, y: 411 }, { x: 455, y: 407 },
  { x: 478, y: 404 }, { x: 501, y: 402 },
  { x: 548, y: 398 }, { x: 571, y: 397 },
  { x: 593, y: 396 }, { x: 634, y: 397 },
  { x: 652, y: 399 }, { x: 683, y: 404 },
  { x: 696, y: 408 }, { x: 708, y: 412 },
  { x: 728, y: 420 }, { x: 736, y: 425 },
  { x: 744, y: 429 }, { x: 754, y: 437 },
  { x: 955, y: 486 }, { x: 995, y: 486 } ]

```

```
var movies = await GetMoviesAsync();  
movies.Where(m => m.Year > 2000)  
    .Select(m => m.Title)  
    .Take(page);
```

IEnumerable<T> LINQ

```
[{ title: "A Tale of Two Cities", year: 1998 },  
 { title: "Whatever the Case May Be", year: 2000 },  
 { title: "Some Like It Hoth", year: 2003 },  
 ...  
 { title: "Happily Ever After", year: 2010 }]
```

```
[{ x: 322, y: 445 }, { x: 314, y: 442 },  
 { x: 313, y: 441 }, { x: 313, y: 437 },  
 { x: 315, y: 435 }, { x: 319, y: 432 },  
 { x: 333, y: 426 }, { x: 345, y: 423 },  
 { x: 375, y: 417 }, { x: 393, y: 414 },  
 { x: 413, y: 411 }, { x: 455, y: 407 },  
 { x: 478, y: 404 }, { x: 501, y: 402 },  
 { x: 548, y: 398 }, { x: 571, y: 397 },  
 { x: 593, y: 396 }, { x: 634, y: 397 },  
 { x: 652, y: 399 }, { x: 683, y: 404 },  
 { x: 696, y: 408 }, { x: 708, y: 412 },  
 { x: 728, y: 420 }, { x: 736, y: 425 },  
 { x: 744, y: 429 }, { x: 754, y: 437 },  
 { x: 955, y: 486 }, { x: 995, y: 486 }]
```

[...] vs. ⚡

.....{ x: 22, y: 45 }.....{ x: 14, y: 42 }.....{ x: 13, y: 41 }.....
{ x: 13, y: 37 }...{ x: 15, y: 35 }.....{ x: 19, y: 32 }...{ x: 333, y:

..“A Tale of Two Cities”....“Whatever the Case May Be”...“Some Like It
Hoth”...“Happily Ever After”.....”Fire + Water”.....”Catch-22”.....

IEnumerable<T> vs. ⚡ event

- ⚡ Pohyb myši
- ⚡ Dáta zo senzorov
 - ⚡ GPS
 - ⚡ Teplota
- ⚡ Prúdy dát
 - ⚡ Twitter Streaming APIs

```
w.MouseMoved += (s, e) => { ... }  
et.GazeChanged += (s, e) => { ... }  
...
```

Your Mouse is a Database

Web and mobile applications are increasingly composed of asynchronous and realtime streaming services and push notifications.

Erik Meijer

Among the hottest buzzwords in the IT industry these days is "big data," but the "big" is something of a misnomer: big data is not just about volume, but also about velocity and variety.⁴

```
class Sensor
{
    public event EventHandler<Measurement> TemperatureChanged;
    private void OnTemperatureChanged(Measurement measure)
    {
        TemperatureChanged?.Invoke(this, measure);
    }
}

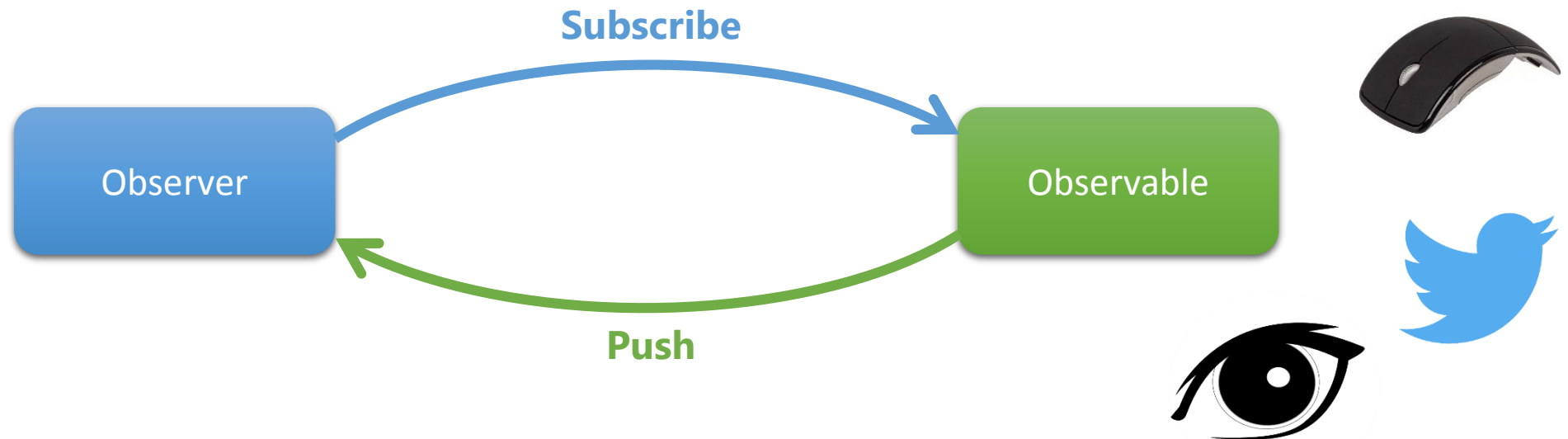
class Client
{ ...
    sensor.TemperatureChanged += Sensor_OnTemperatureChanged;
    ...
    void Sensor_OnTemperatureChanged(object sender, Measurement args) { ... }
    ...
    sensor.TemperatureChanged -= Sensor_OnTemperatureChanged;
}
```

Prečo nestačí ⚡ event

```
sensor.TemperatureChanged += (s, e) =>
{
    if (e.Temperature > MAX_TEMP)
    {
        // ... Pridať TimeStamp
        // ... Zalogovať
        // ... Odoslať na server
        // ... Skontrolovať s iným senzorom
    }
}
sensor.TemperatureChanged -= /* ? */;
// sensor.TemperatureChanged.Where(e => e.Temperature > MAX_TEMP)
```

IObservable<T>

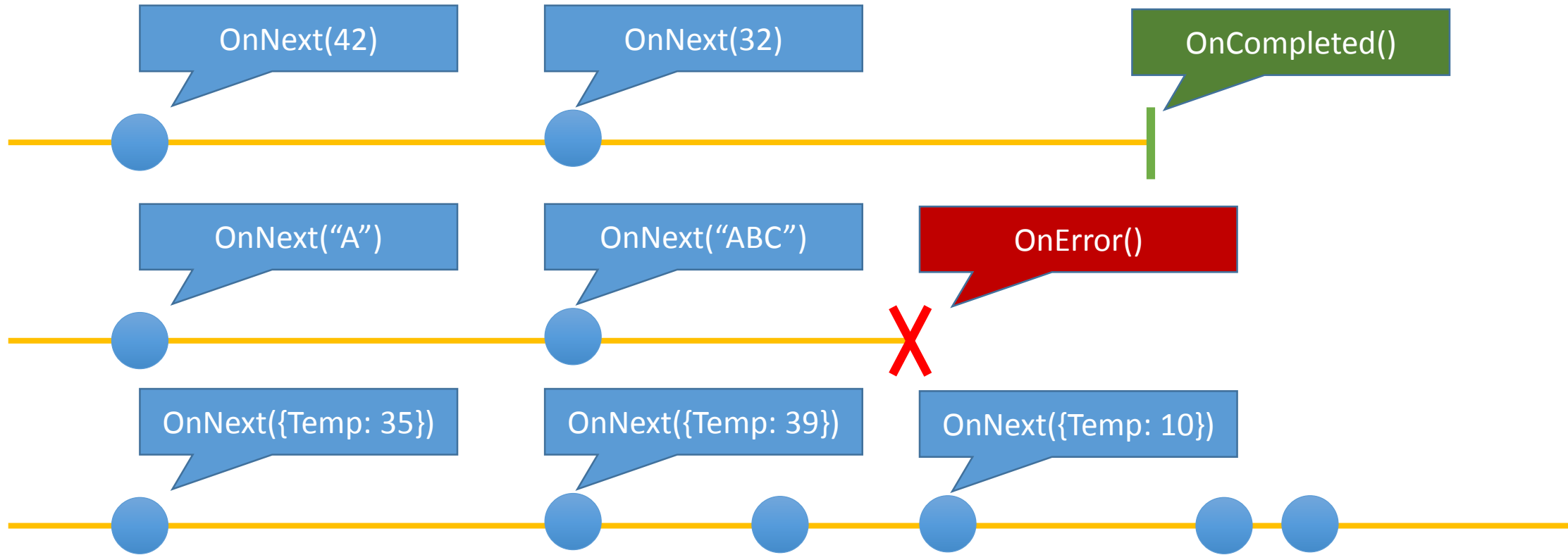
- Producenti dát – **observable sequences**
 - .NET event, sensor API, Task...
- Konzumenti – **observers**
 - EventHandler, ContinueWith...



```
namespace System
{
    public interface IObservable<out T>
    {
        IDisposable Subscribe(IObserver<T> observer);
    }

    public interface IObserver<in T>
    {
        void OnNext(T value);
        void OnError(Exception error);
        void OnCompleted();
    }
}
```

IObservable<T>



`OnNext*` (`OnError` | `OnCompleted`)?

Získanie IObservable<T>

Subject<T> : IObservable<T>, IObservable<T>

Observable.Range, .Interval, .Range, .Return...

Observable.FromEventPattern

Reactive Extensions

Rx = Observables + LINQ + Schedulers

<http://reactivex.io>

<https://github.com/Reactive-Extensions>

- C#/.NET
- C++
- JavaScript
- Java
- Ruby
- Python



Operátor

<http://reactivex.io/documentation/operators.html>

```
var window = /* zmeny v otvorení okna */;  
var temperatures = /* register */;  
var warnings = temperatures.Select(t => new  
    Timestamped<Temperature>(t.Temperature, DateTime.Now))  
    .CombineLatest(window, (t, w) => new RoomState(t,w))  
    .Where(t => t.Temperature > MAX_TEMP);  
  
warnings.Subscribe(w => SendNotification(w));
```

Demo

FromEventPattern

Throttling, DistinctUntilChanged

FromAsync

Demo

Sledovanie pohľadu – eye tracking

Fixácie pohľadu

Work in progress

Sledovanie pohľadu

Eye tracker

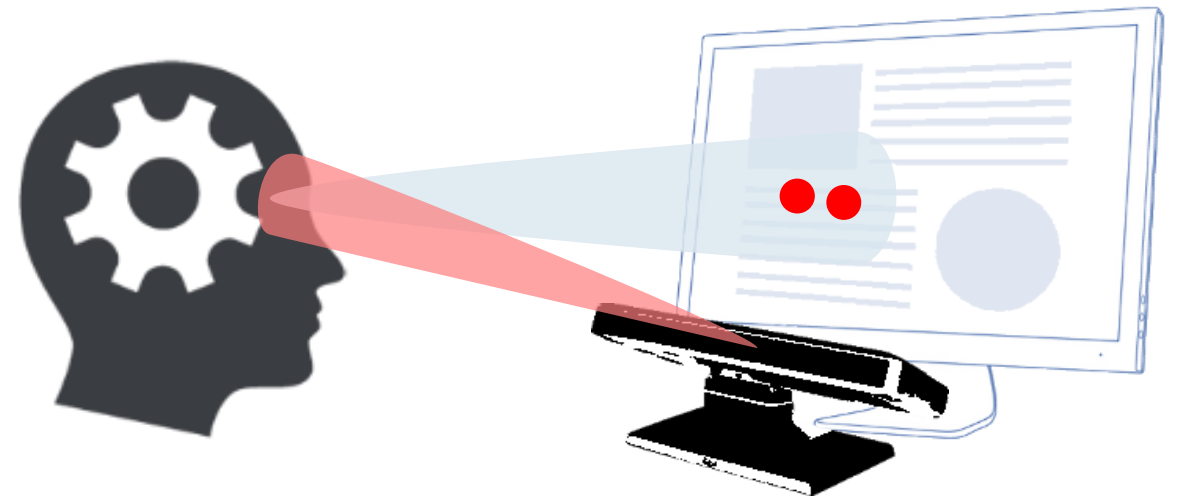
- Infračervené kamery
- Sledovanie zreničky
- Poloha oka

Prepočítanie na súradnice (x,y) na obrazovke

Rôzne frekvencie: 60 Hz, 300 Hz

Tobii X2-60, TX-300

Tobii EyeX



UXI@FIIT

UX Lab



emotion
detection



Tobii studio



300 Hz
gaze tracking



3D depth camera



TV



EEG



ECG, GSR, FSR, °C



Smartphones,
tablets

UX Class

20x



Enable an entire classroom with eye tracking



3D depth camera



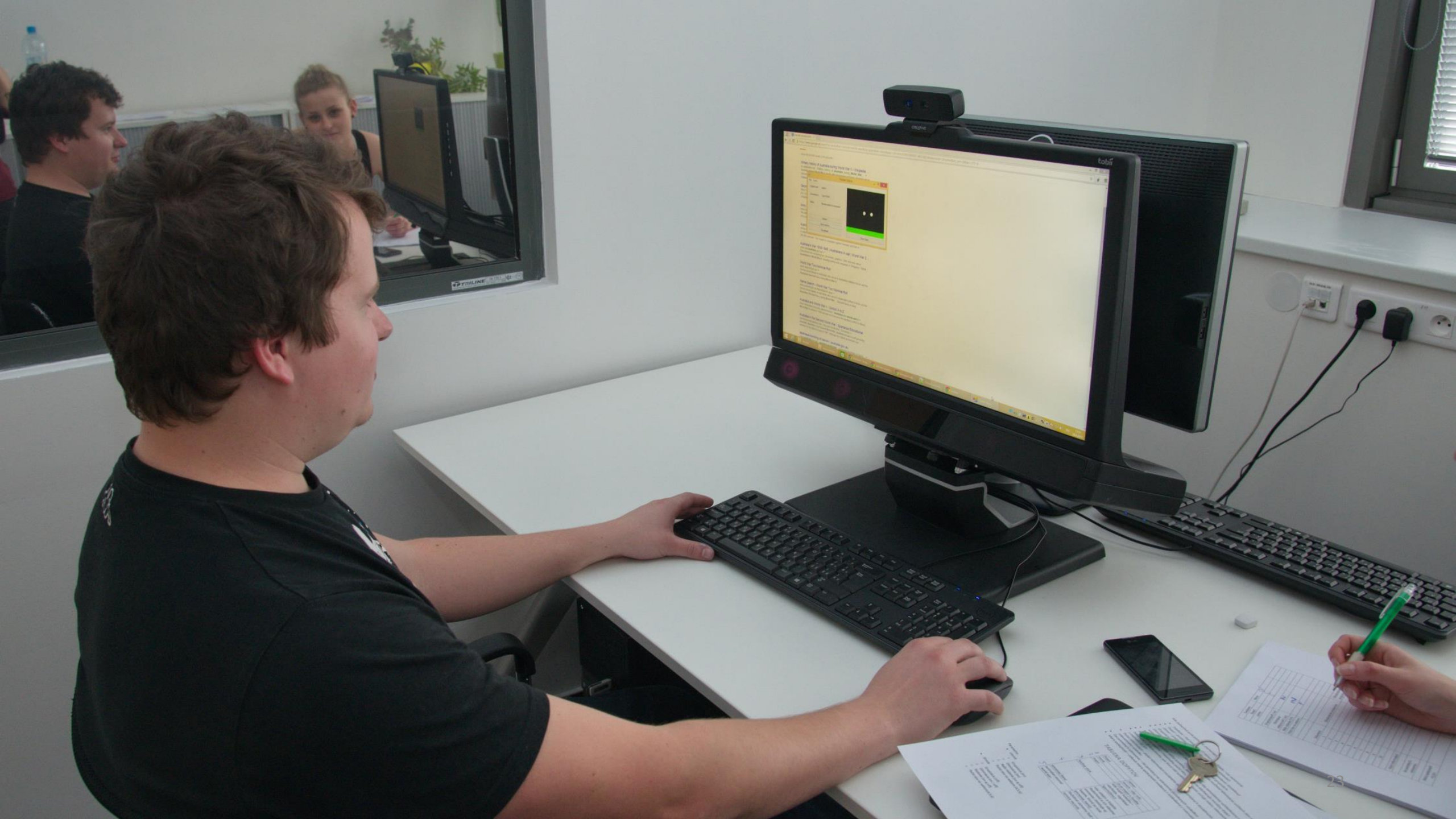
60 Hz gaze tracking



UX Research
infrastructure software



Emotion detection





Eye tracker

Jeden zo zdrojov dát

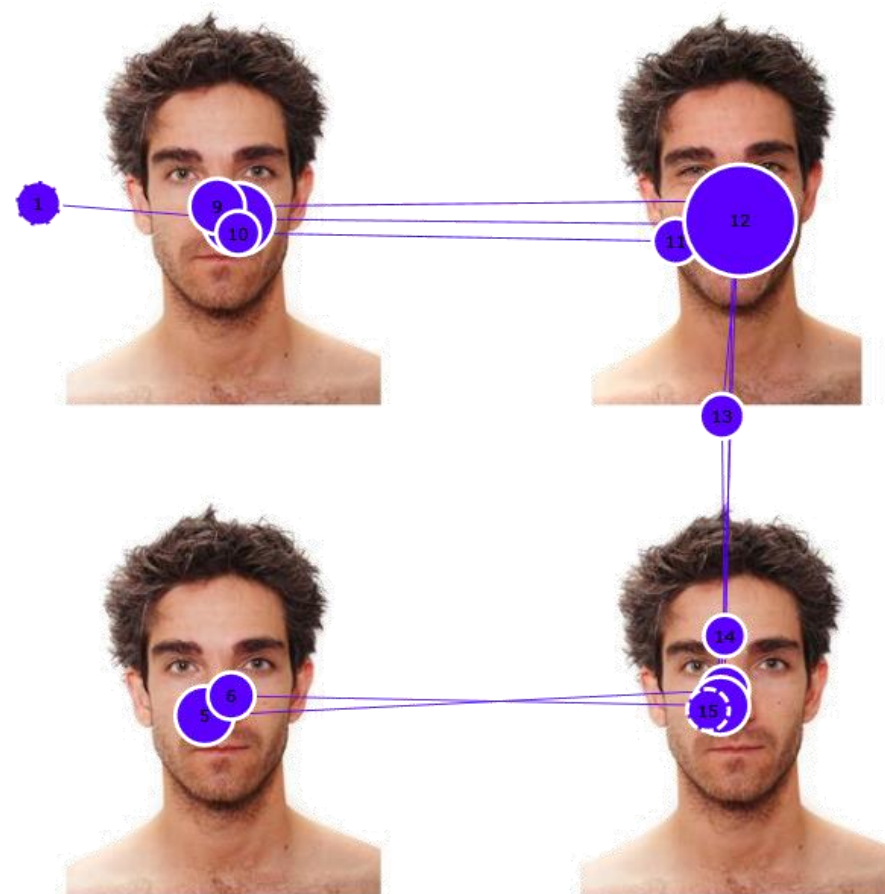
Vystavuje event DataReceived

Surové dáta – 60/sek.

- Zašumené – eye tracker nevidí oko (oči)
- Nepresné

Zaujímajú nás fixácie

- Spracovanie informácie mozgom
- Pohyb medzi fixáciami – sakády



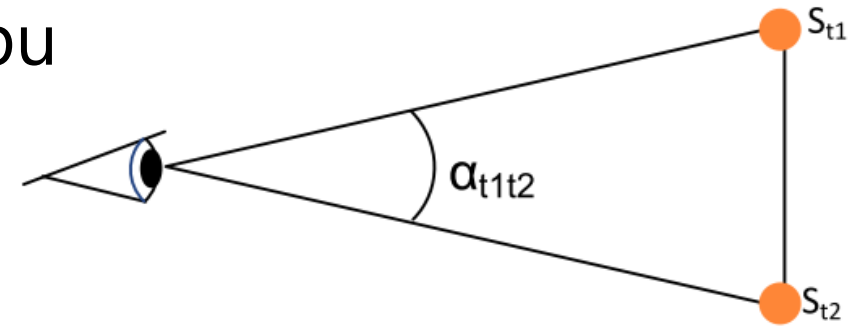
Identifikácia fixácií

Pozícia oka (pohľadu) pred a na konci pohybu

V reálnom čase

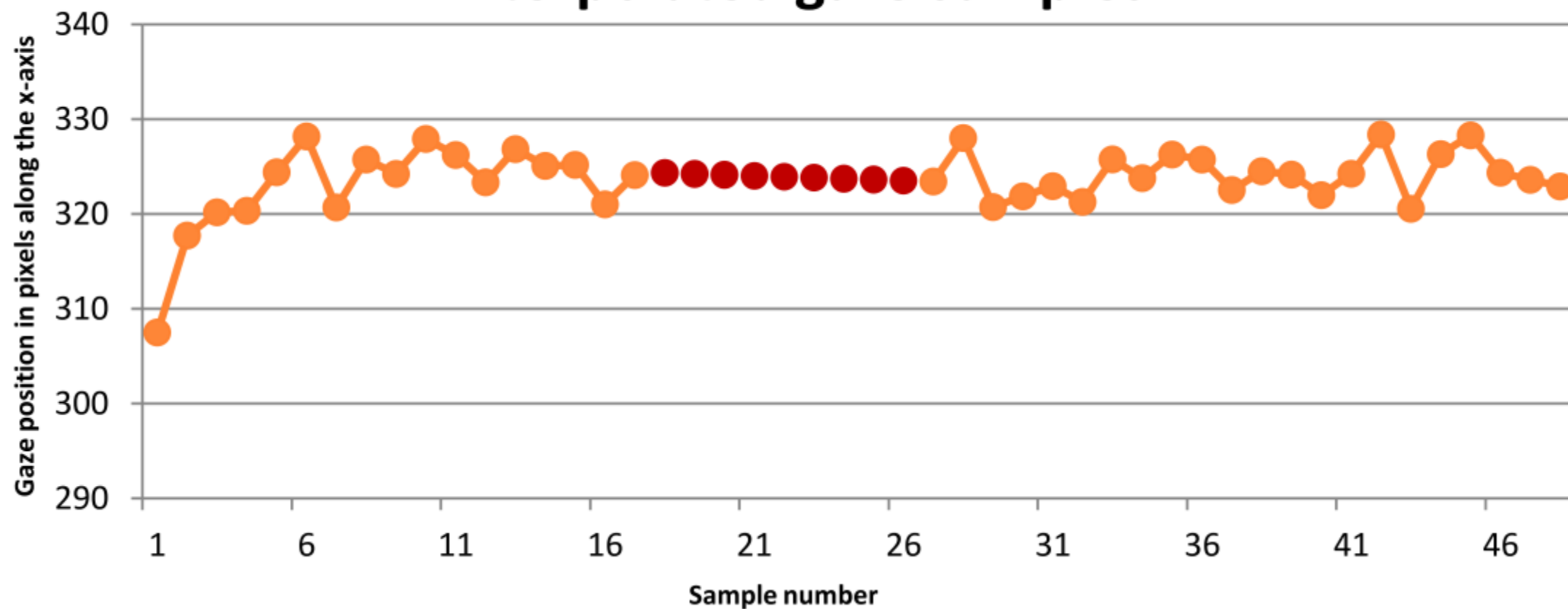
Výpočet rýchlosti pohybu oka

- Uhlová rýchlosť
- Pri simulácií však počítame rýchlosť vzhľadom na cieľ pohľadu



Predtým však doplnenie chýbajúcich dát

Interpolated gaze samples



Identifikácia fixácií

Interpolácia pohybu pohľadu

Výber oka pre výpočet

Výpočet rýchlosti

Klasifikácia medzi fixácie a sakády

Krátke fixácie = sakády



Rx.OnCompleted()

Rx = Observables + LINQ + Schedulers

Nie len .NET, ale JavaScript, Python, Ruby, Java, C++
<http://reactivex.io>

Cortana



<http://www.uxi.sk/>

Martin Konôpka

Microsoft Student Partner

PhD student @ FIIT STU

martinkonopka@outlook.sk