

11<sup>th</sup> Central and Eastern European Software Engineering  
Conference in Russia - CEE-SECR 2015

October 22 - 24, Moscow



# Kotlin for robotics: it is functional

Roman Belkov, Iakov Kirilenko

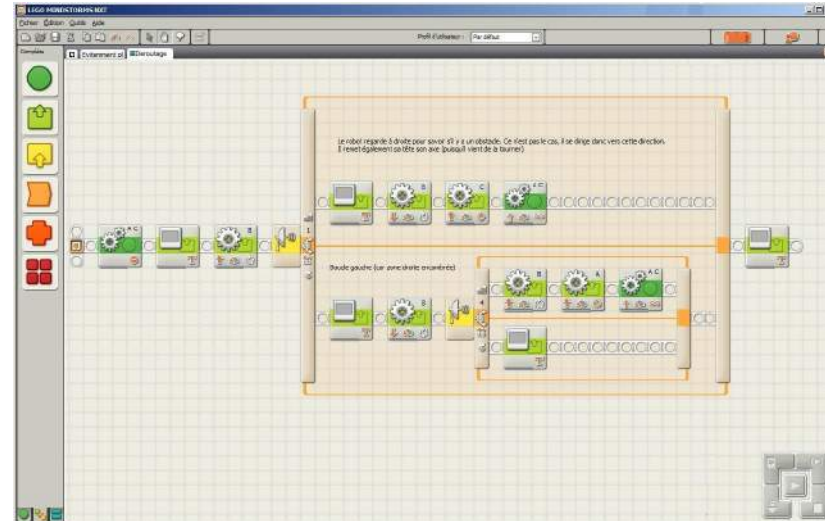
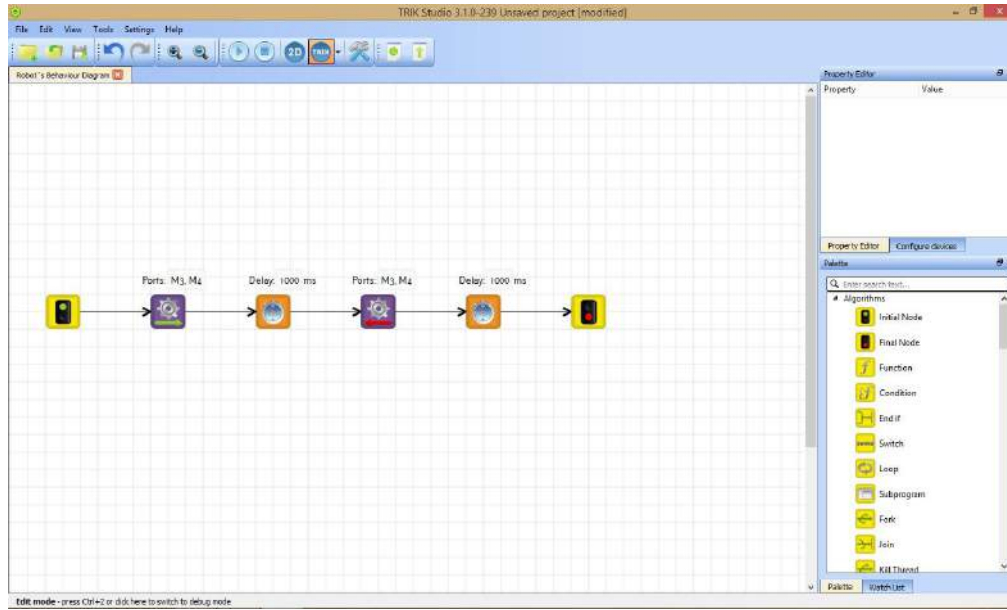
**TRIK**

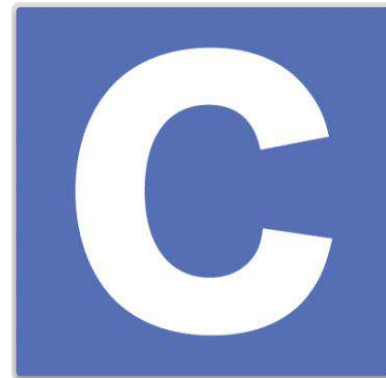
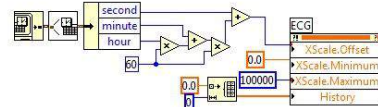
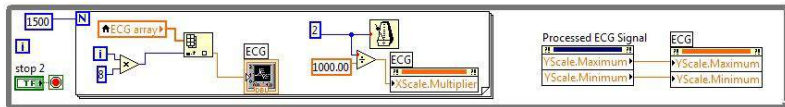
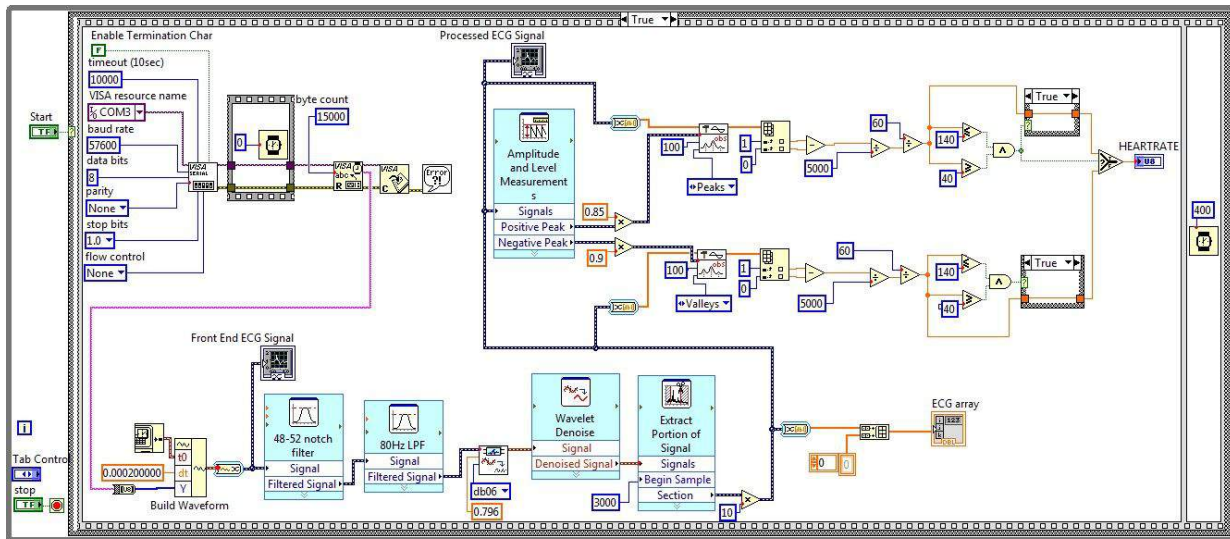
learn to invent the future



**JetBRAINS**

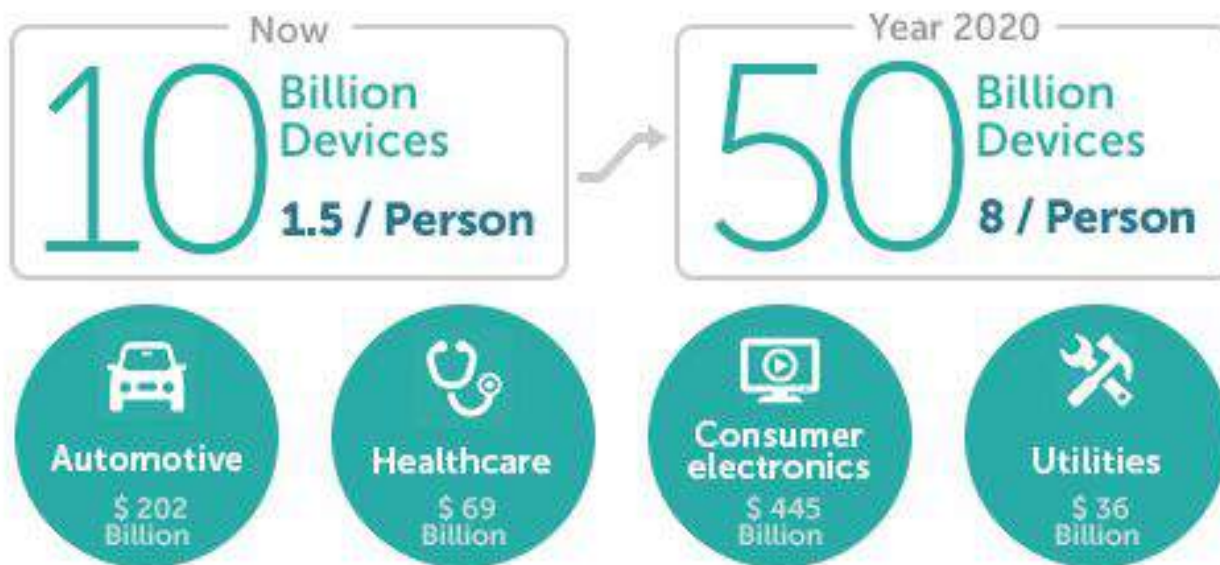
# Gap to fill





# Growth of IoT

## IoT Predictions 2020



# Kotlin



- Multi-paradigm
- Compact
- Type-safe
- Event-driven
- 100% Java interoperable

# What we wanted

- Support for all common TRIK sensors
- Kotlin/Java programming
- Event-driven mode
- No GPIO programming
- Tons of fun

# Example



# Code was

```
function setBatteryTimer() {
  function batteryLoop() {
    g_bc = c_fb / brick.battery().readVoltage();
  };

  var batteryTimer = script.timer(500);
  batteryTimer.timeout.connect(batteryLoop);
  batteryTimer.start();
}

function setKeys() {
  brick.keys().buttonPressed.connect(
  function(code, value) {
    if (code == KeysEnum.Up && value == 0) {
      print("exit");
      brick.stop();
      script.quit();
    }
    if (code == KeysEnum.Down && value == 2) {
      print("calibrate");
      g_offset = g_od;
    }
  }
);
}
```



# Code now

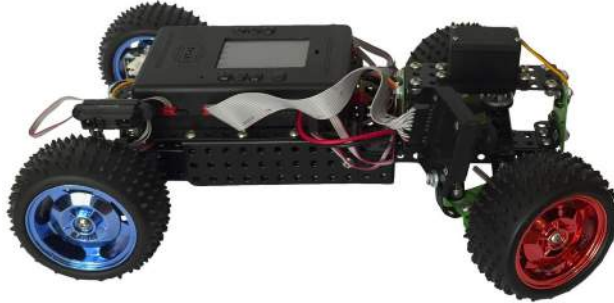
```
val observableButtons = buttons.toObservable()
observableButtons.filter { it.button == ButtonEventCode.Power }.subscribe { Exit() }
observableButtons.filter { it.button == ButtonEventCode.Down }.subscribe { g_offset = g_od }

fun setBatteryTimer() {
    fun batteryLoop() {
        g_bc = c_fb / battery.readVoltage()
    }
    Observable.interval(500, TimeUnit.MILLISECONDS).subscribe { batteryLoop() }
}
```

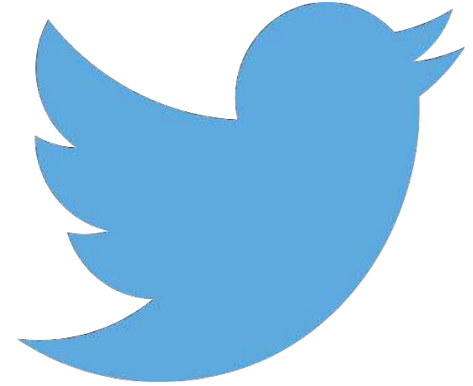
# Example



+



+



=

**AWESOMENESS!**

# All you need is

```
fun limitServo(x: Int) = limitAbs(MAX_SERVO_ANGLE, x)
```

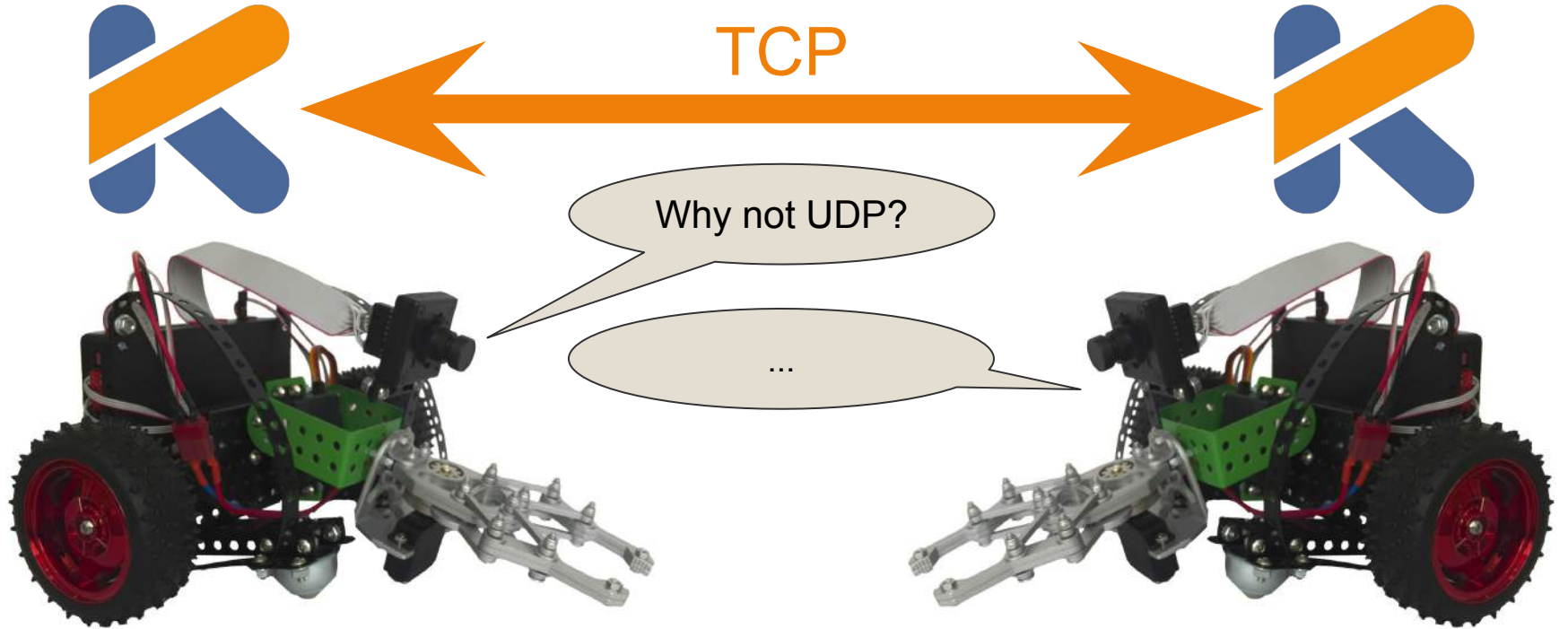
```
val aimX =  
  locationStream  
    .scan(0, { acc, loc -> limitServo(acc - updatePositionX(loc.x)) })  
    .subscribe(servoX)
```

```
val aimY =  
  locationStream  
    .scan(0, { acc, loc -> limitServo(acc - updatePositionY(loc.y)) })  
    .subscribe(servoY)
```

# Live in Twitter

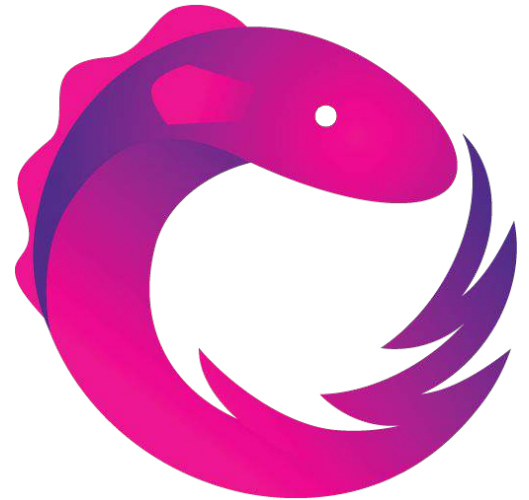
Look at [twitter.com/TRIK\\_Live](https://twitter.com/TRIK_Live)

# Example



# Kotlin experience

- Totally cool for most of the tasks
- Interop with Java feels natural
- Concise syntax
- RxJava & RxKotlin



# Kotlin experience

- Great tools (IDEA)
- Compatibility over functional paradigm
- Runtime errors are compilation-time errors
- Not released yet



**IntelliJIDEA**

# Results

- Java SE on TRIK
- Kotlin library
- Extremely friendly API to Java programs
- Demos



# //TODO

- Implement basic DSL
- Observables optimization
- Try out new JDKs
- More Demos & Docs

# Contacts

- [github.com/RomanBelkov/TrikKotlin](https://github.com/RomanBelkov/TrikKotlin)
- [Roman.Belkov@gmail.com](mailto:Roman.Belkov@gmail.com)
- [kotlinlang.org](http://kotlinlang.org)