The Power of Software

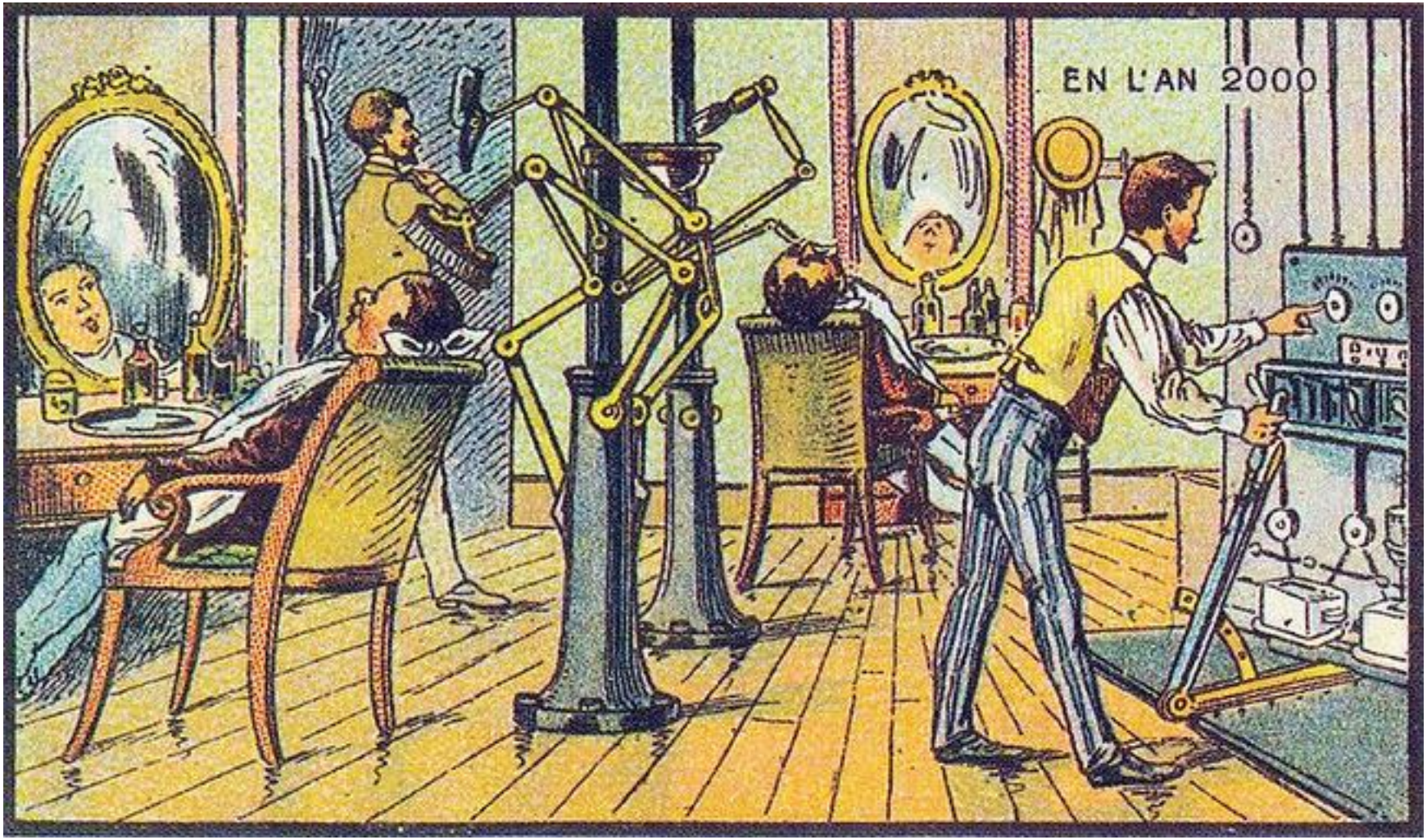
Hassan Charaf

hassan@aut.bme.hu

BME AUT

Department of
Automation and
Applied Informatics

France in 1900



Source: http://www.szineshir.hu/2015/10/17-meglepo-kep-ilyennek-kepzeltek-jovot.html

Outline

- Facts
- Trends
- Communication
- Software – the power in industry
- Applications, applications, and applications…

Outline

- Facts
- Trends
- Communication
- Software – the power in industry
- Applications, applications, and applications…

Mobile devices

Global Mobile Data Traffic Drivers

Source: Cisco VNI Global Mobile Data Traffic Forecast, 2014–2019

© 2014 Cisco and/or its affiliates. All rights reserved. Cisco Confidential

One billion plan

- 1995: PC
- 2005: Mobile phone
- 2015: Wearable devices

Generations of mobile communication

2G – 1992
Voice
SMS

3G – 2002
+ Data
+ Location

4G – 2012
+ Video
+ 3D

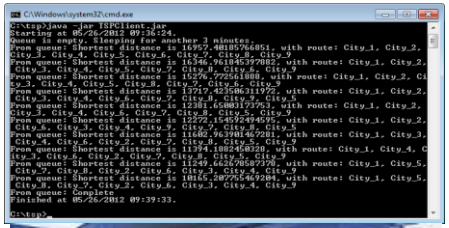
5G – 2022
+ ?
+ ?

Outline

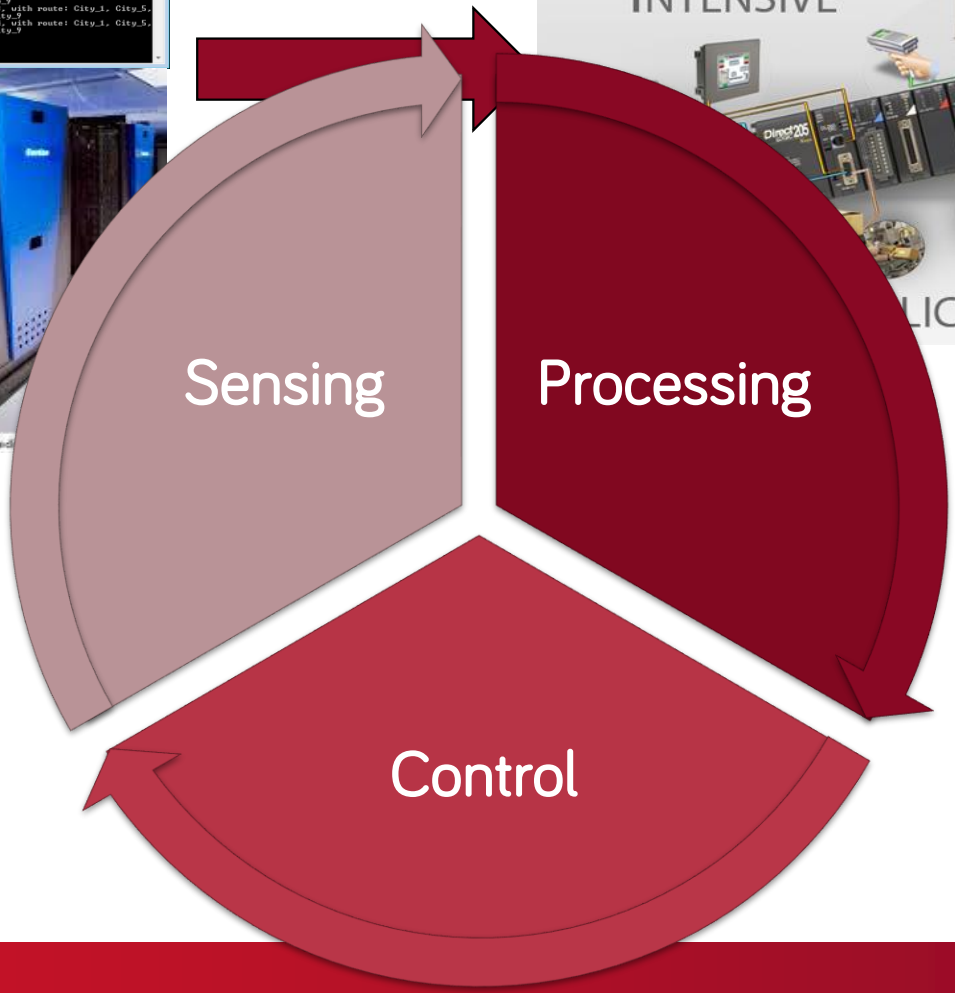
- Facts
- Trends
- Communication
- Software – the power in industry
- Applications, applications, and applications…

Trends

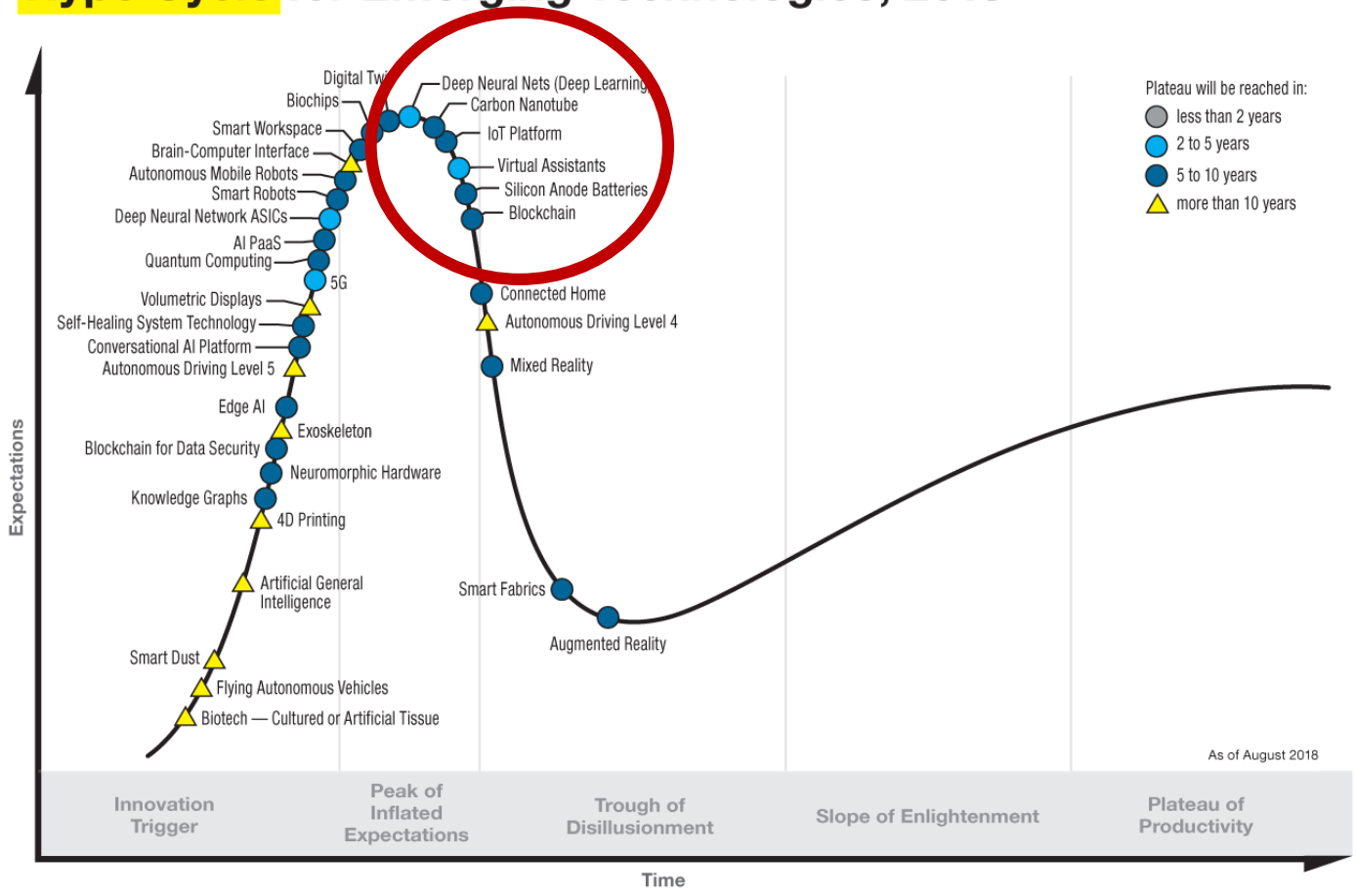
- Capabilities of mobile/wearable devices
- Power and Cloud services
- Internet of Things (IoT)
- Industry 4.0, Society 5.0
- Security
- Big Data
- Artificial Intelligence
- Smart… city, home, environment, car, etc.



Gordon Supercomputer at SDSC (cred



Hype Cycle for Emerging Technologies, 2018



gartner.com/SmarterWithGartner

Source: Gartner (August 2018)

© 2018 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

Three Trends

AI Everywhere

Deep Learning
Deep Reinforcement Learning
Artificial General Intelligence
Autonomous Vehicles
Cognitive Computing
Commercial UAVs (Drones)
Conversational User Interfaces
Enterprise Taxonomy
Ontology Management
Machine Learning
Smart Dust
Smart Robots
Smart Workspace

Transparently Immersive Experiences

4D Printing
Augmented Reality
Brain-Computer
Interface
Connected Home
Human Augmentation
Nanotube Electronics
Virtual Reality
Volumetric Displays

Digital Platforms

5G
Digital Twin
Edge Computing
Blockchain
IoT Platform
Neuromorphic Hardware
Quantum Computing
Serverless PaaS
Software-Defined Security

gartner.com/SmarterWithGartner

Source: Gartner
© 2017 Gartner, Inc. and/or its affiliates. All rights reserved.

Gartner

Outline

- Facts
- Trends
- Communication
- Software – the power in industry
- Applications, applications, and applications…

Via della Conciliazione

April 4, 2005

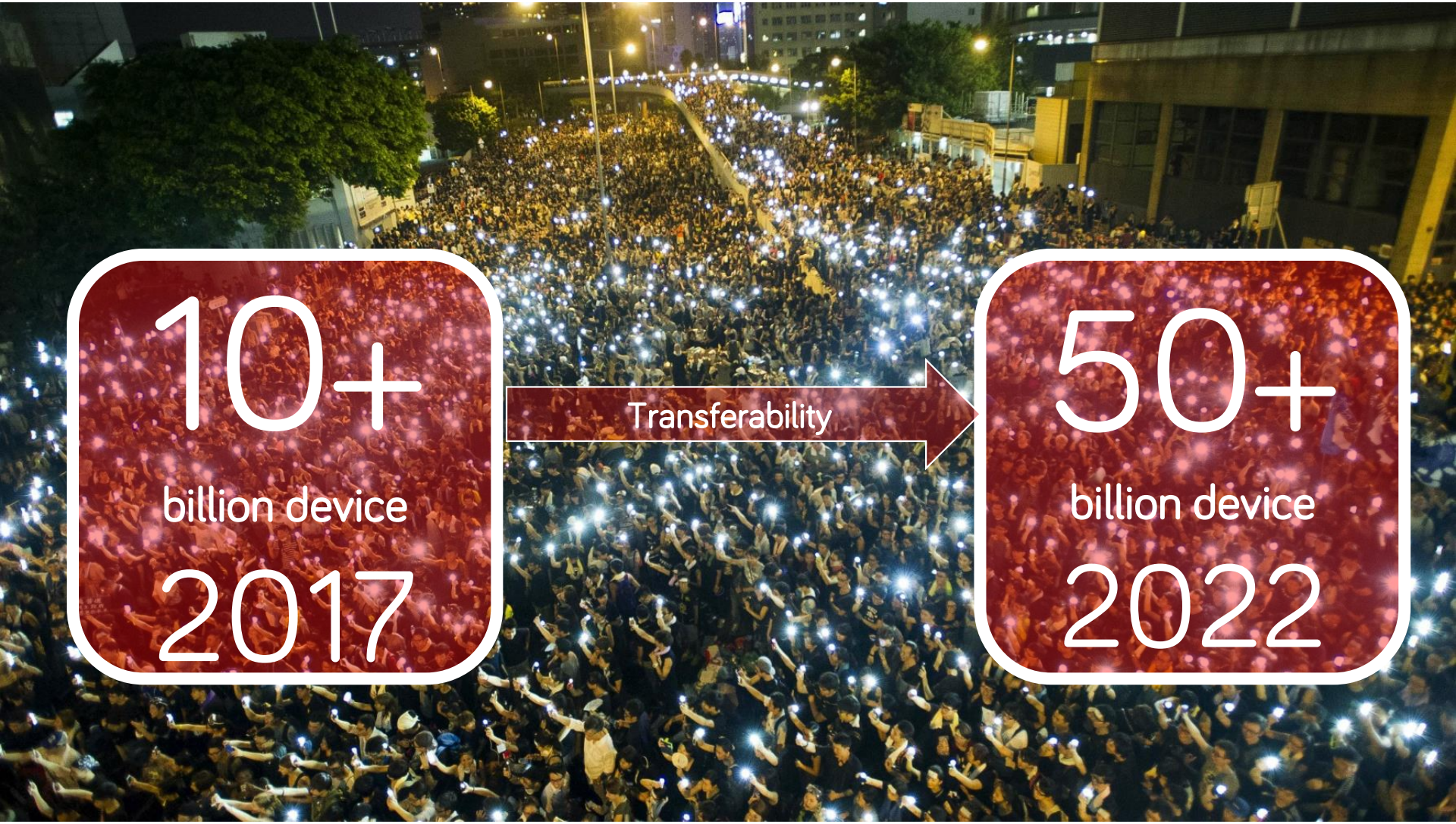
Source: http://www.spiegel.de/panorama/bild-889031-473266.html

March 12, 2013

Source: http://www.spiegel.de/panorama/bild-889031-473242.html

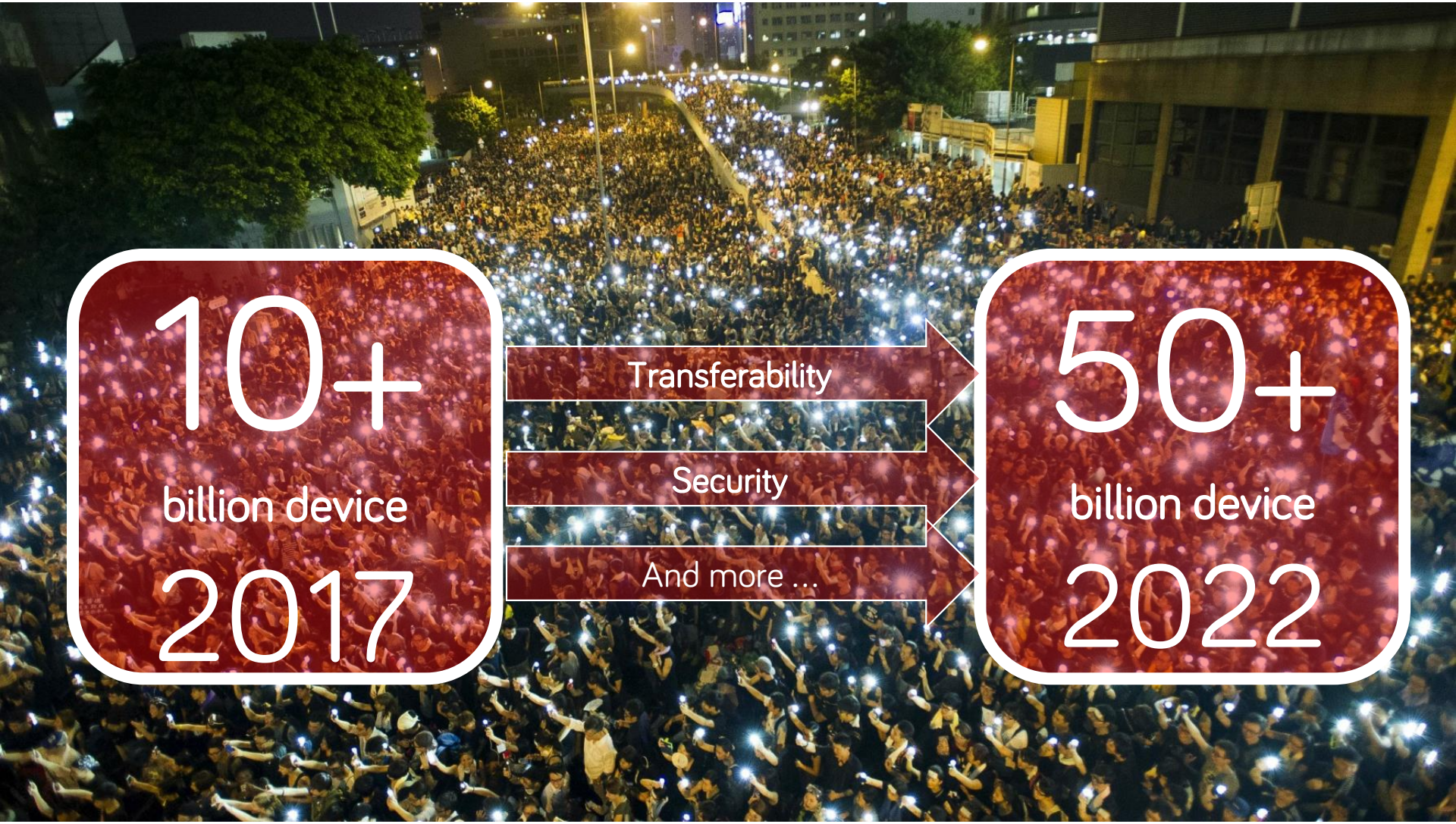
Forrás: japantimes.co.jp/news/2014/09/30/asia-pacific/hong-kong-democracy-protesters-set-deadline-for-demands/

Forrás: japantimes.co.jp/news/2014/09/30/asia-pacific/hong-kong-democracy-protesters-set-deadline-for-demands/



Forrás: japantimes.co.jp/news/2014/09/30/asia-pacific/hong-kong-democracy-protesters-set-deadline-for-demands/

Forrás: japantimes.co.jp/news/2014/09/30/asia-pacific/hong-kong-democracy-protesters-set-deadline-for-demands/

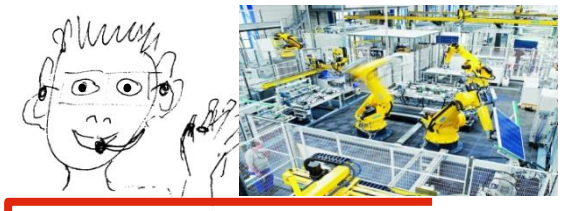


Forrás: japantimes.co.jp/news/2014/09/30/asia-pacific/hong-kong-democracy-protesters-set-deadline-for-demands/

The tactile Internet

Source: http://ostsee-spezial.de/?p=148

Mobile Communication



5G – 2022
+ Tactile internet
+ M2M
+ Tb/s
+ Reliable and secure

2G – 1992
Voice
SMS

3G – 2002
+ Data
+ Location

4G – 2012
+ Video
+ 3D

> 10 Gbit/s per user

< 1 ms RTT

> 10k sensors per cell

$< 10^{-8}$ outage

$< 10^{-12}$ security

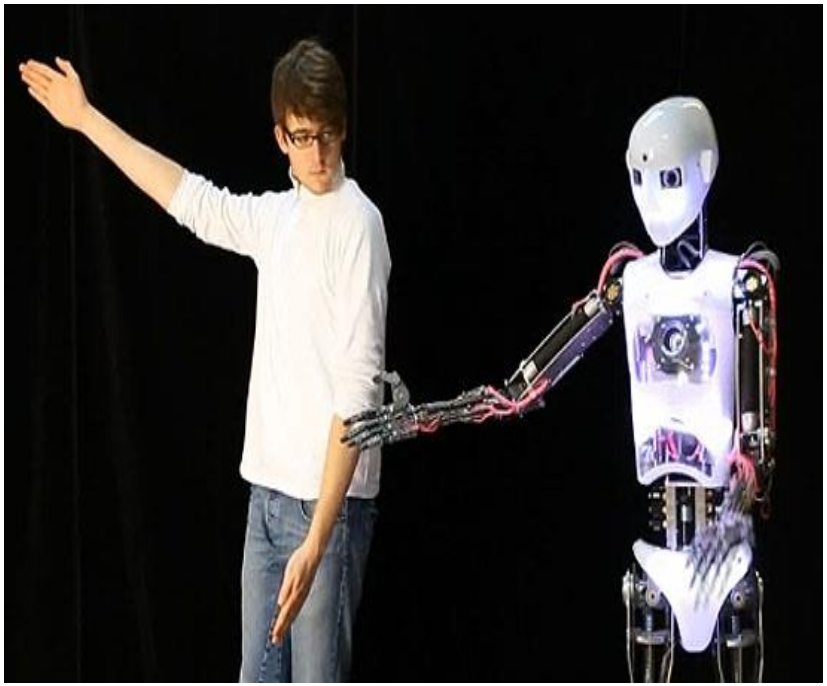
10x10 heterogeneity

Applications

- Games

Applications

- Games
- Health, remote monitoring



Source: http://www.dvice.com/archives/2011/05/kinect_controll_1.php; http://images.gizmag.com/hero/8456_51207105642.jpg

Applications

- Games
- Health, remote monitoring
- Energetics

Applications

- Games
- Health, remote monitoring
- Energetics
- Manufacturing

http://jerryrushing.net/wp-content/uploads/2012/04/robotic_assembly_line1.jpg

http://www.witchdoctor.co.nz/wp-content/uploads/2013/01/robot-fabrication-station.jpg

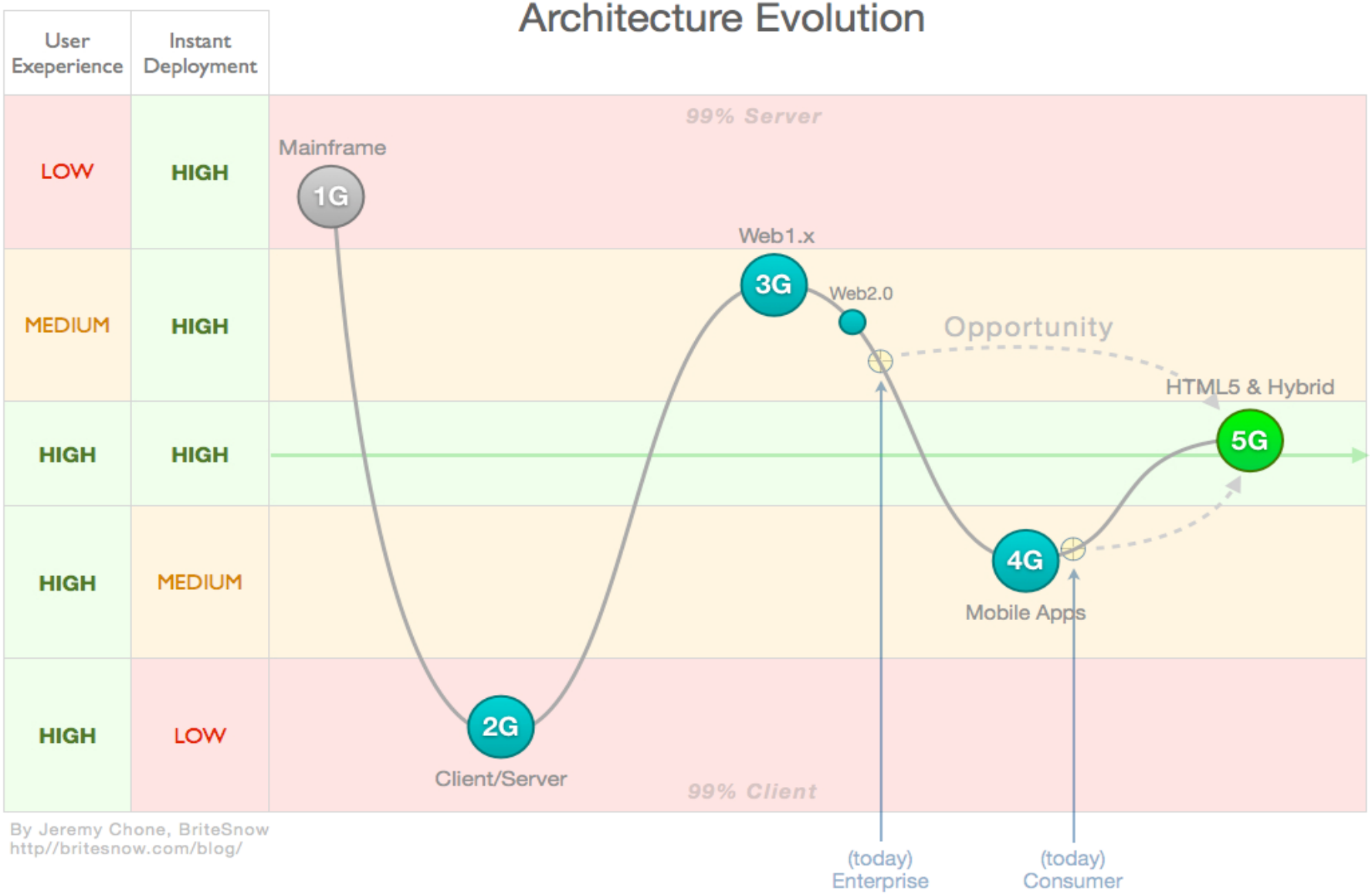
Applications

- Games
- Health, remote monitoring
- Energetics
- Production lines
- Transport

Outline

- Facts
- Trends
- Communication
- Software – the power in industry
- Applications, applications, and applications…

5 Generations of Software Architecture Evolution



The horizontal role of the software

- Software is a crucial part of our daily lives
  - The software is the determinant of the competitiveness of various fields
  - Due to the crosscutting power of ICT, taking into account the demands of the industry, the research results are utilized quickly in different domains
- Increased demand for applications and services
  - Applications and services have increased quantitative and qualitative, industrial and social expectations
  - Sustainable development processes mean competitiveness

Application domains

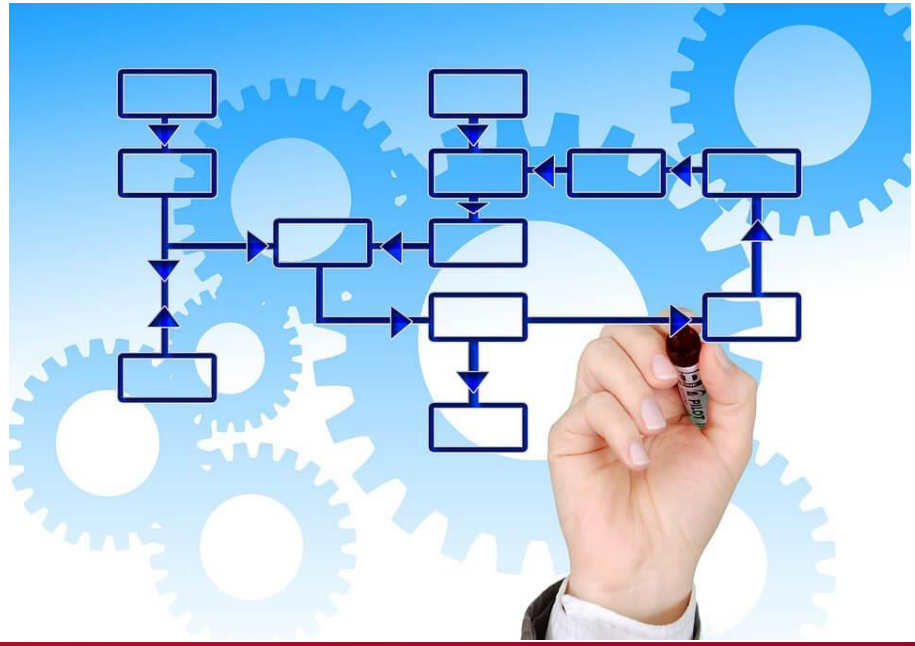
What do we need to develop good software?

Statement 1

- Experience says that ***TEAMS*** with different capabilities can achieve real results, this is true for innovation and software development too

Statement 2

- ***Consciously formed unified staff*** (researchers, engineers, lawyers, economists, marketers) and
- ***Processes that work along the same principles*** to make ideas work and be productive and then put them on the market

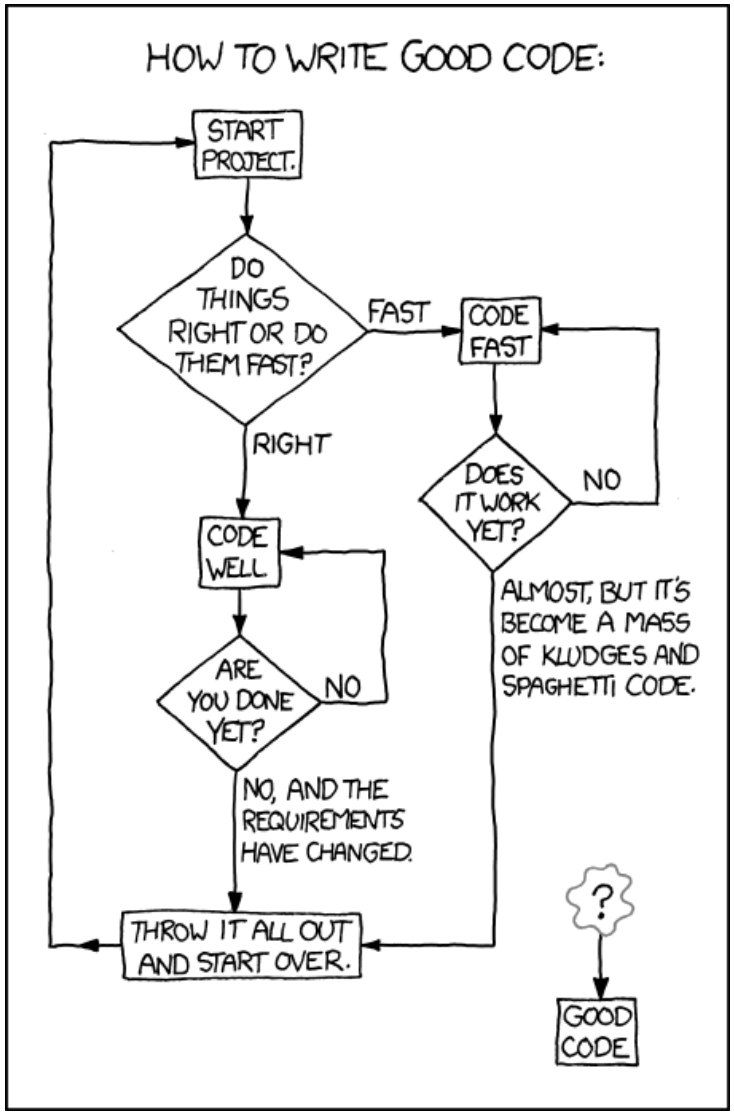


Receipt

- A powerful ***TEAM*** that covers the whole innovation process (development process)
- Consciously developed ***PROCESSES*** that work based on uniform principles
- **LEADERSHIP** commitment, continuous active support, iterative strengthening

Software Quality

Fast vs. Right coding



source: https://www.quora.com/What-are-the-five-most-important-programming-concepts

What do we say typically if the code is not working?

- It is strange…
- It was working yesterday
- It is a hardware issue
- Probably you are using a wrong version
- Somebody has changed my code
- Is it possible that your computer has a virus?
- This is not compatible with your system
- It works for me… ☺

Challenges during coding

- Did you ever had to work with bad code?
  - „Reading the code takes more time than rewriting it”
- Did you ever experienced that there is no time to implement a feature properly and refactor the code because of the short deadlines?

„As the code size increases, the productivity of the developers decrease.”

Code Rot

- The application starts with a clear and clean architecture
- What happens after a while?
  - > The code starts to rot: small hacks, more and more if-else branches and later these will dominate in the code
- Impossible to maintain, hard to add new features - > developers beg for a refactor

Symptoms of Code Rot

- Rigidity
  - The cost is high even for small changes
- Fragility
  - Small changes cause large errors
- Immobility
  - Main parts can not be organized into modules
- Viscosity
  - > It is easier to implement new features with „hacks”

Code rot – What is the cause?

- Changing requirements
  - If the architecture does not support changes
- What changes cause code rot?
  - *Changes that require new, unexpected dependency on class level*
- Most symptoms lead back to improper class dependencies
- What can we do?

OO Design Principles help to avoid Code rot!

Cloud

Service-oriented business model

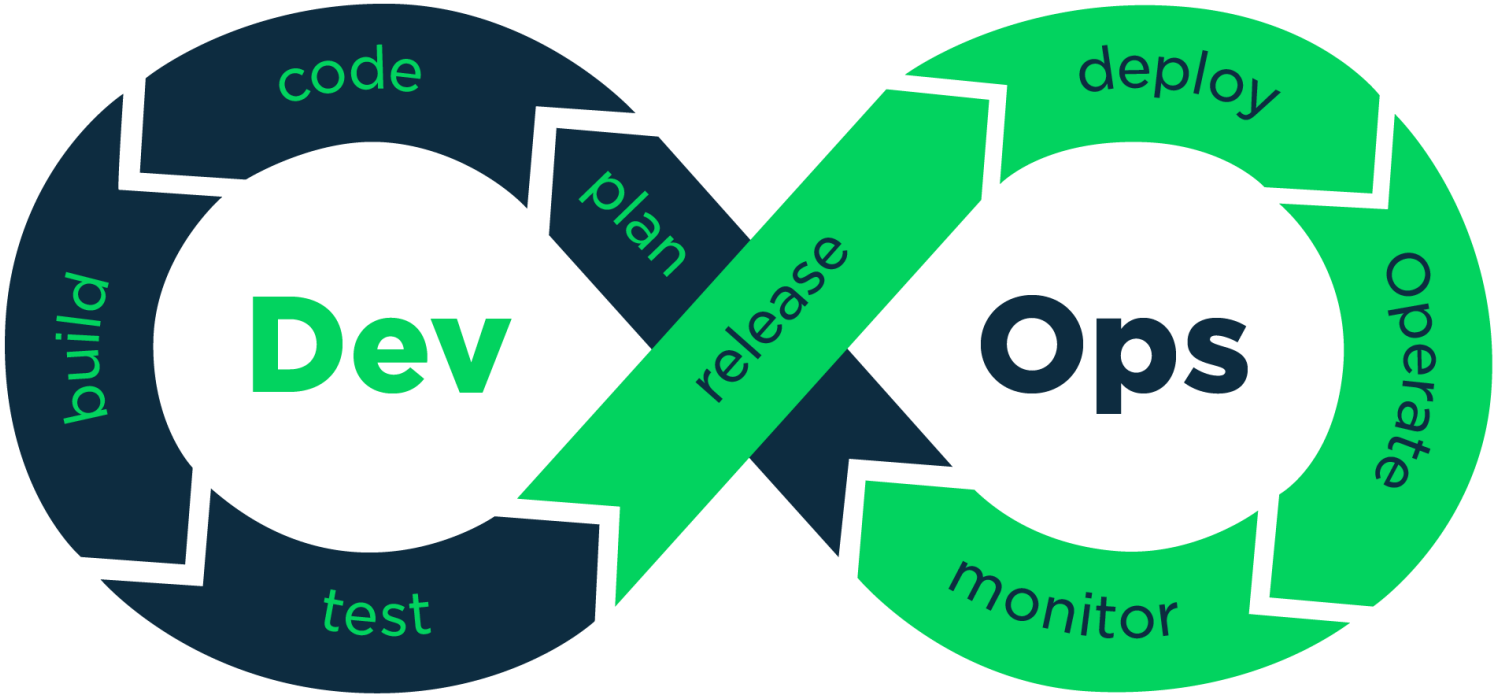
Cloud computing

„Cloud computing is a model for enabling ubiquitous, convenient, ***on-demand network access*** *to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that* ***can be rapidly provisioned and released*** *with minimal management effort or service provider interaction. This cloud model is composed of five essential characteristics, three service models, and four deployment models."*

Cloud solutions

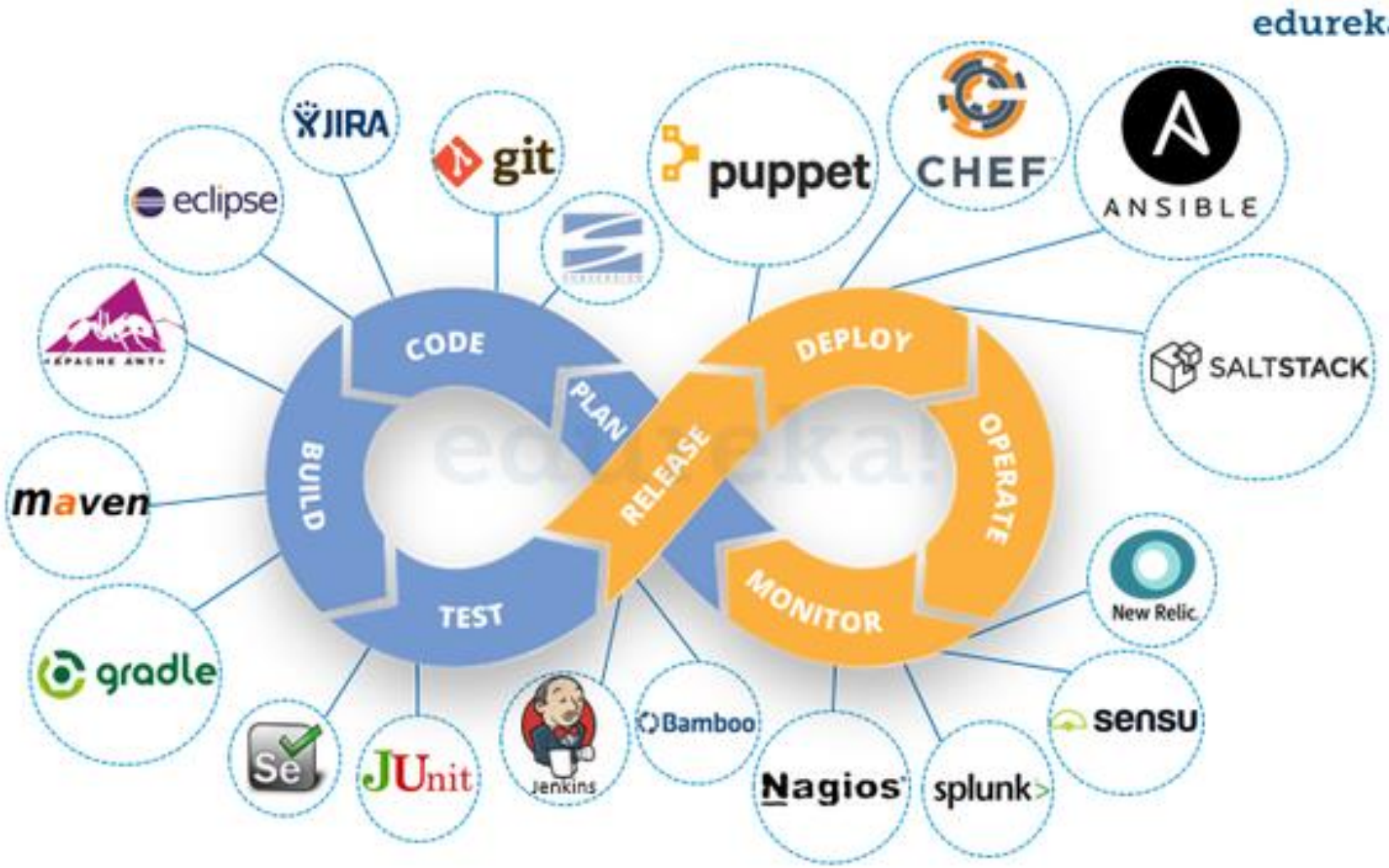
- Amazon Web Services (AWS)
- Microsoft Azure
- Google Cloud Platform
- Jelastic
- Others…

DevOps



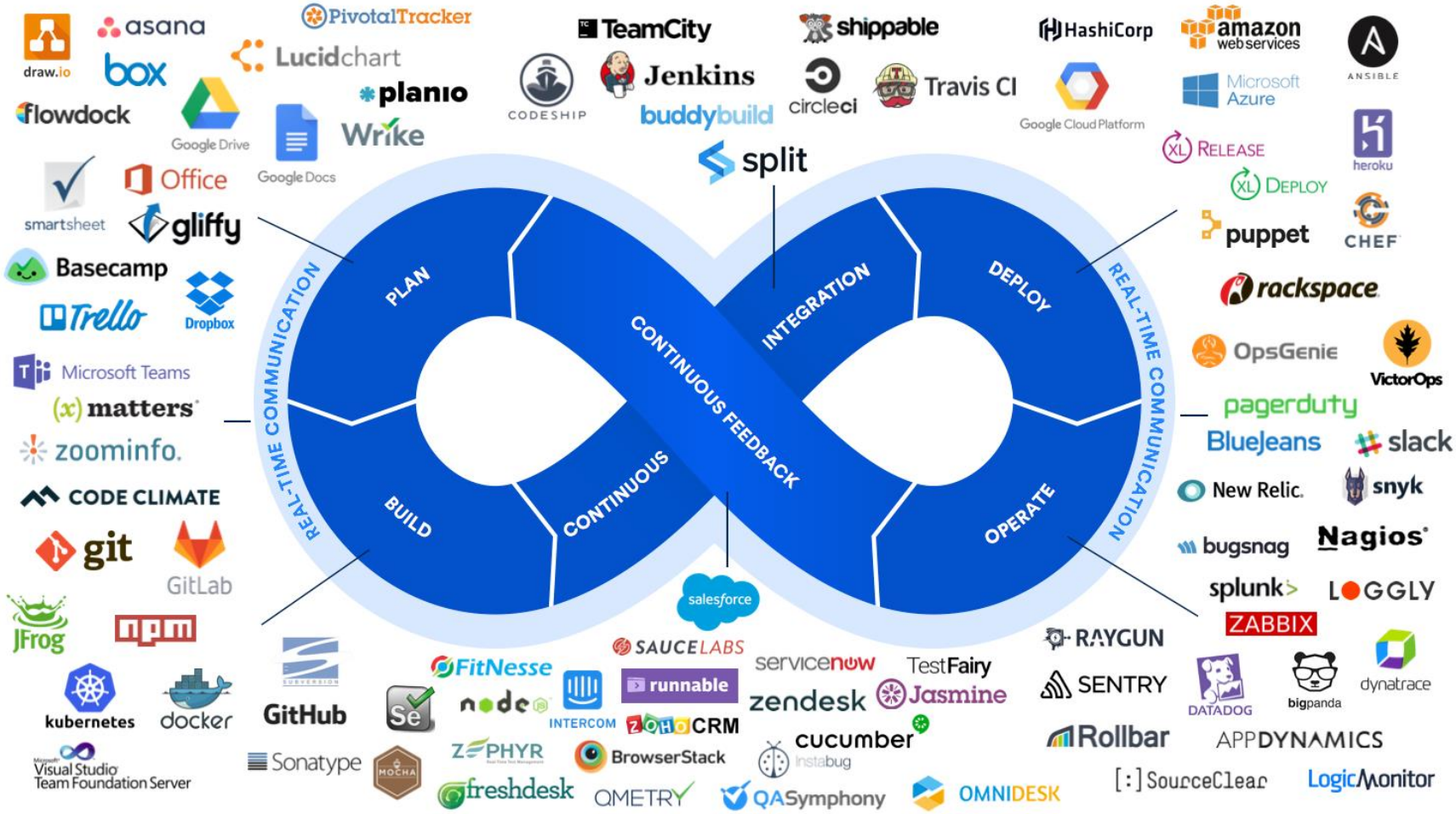
Source: medium.com/@neonrocket

DevOps



Source: technofaq.org

DevOps



Source: atlassian.com

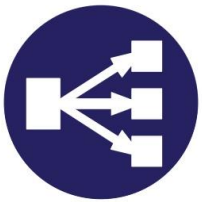
Elastic Digital Architecture

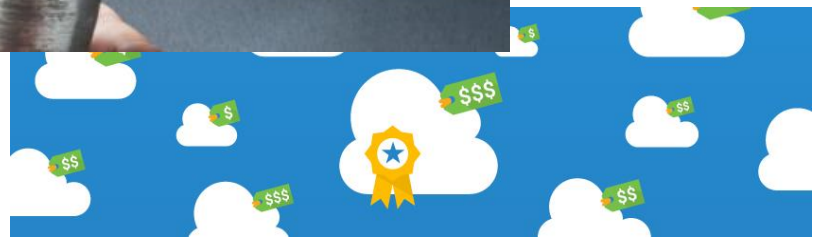
Key Questions

- What is the **best way to structure** our enterprise for success in the 21st century?
- Is our current operating model **adaptable** or **fixed**? What could we do with more enterprise elasticity?
- Does our current IT architecture **future-proof** our enterprise?
- Does our technology deliver enough **elasticity** to compete?
- What must IT do to **ensure** the **future** of our company?

Scalable vs. Elastic Architecture

- **Scalability**: Increasing the capacity to meet the increasing workload.
- **Elasticity**: Incr eet the increasing **or**
- Why is it impo ure?





Best Way to Structure our Enterprise in 21st

- Importance of elastic enterprise structure
- Industrial era business models are no longer effective
- Knowledge economy
- Turbulent environments
- The most flexible and nimble enterprise wins

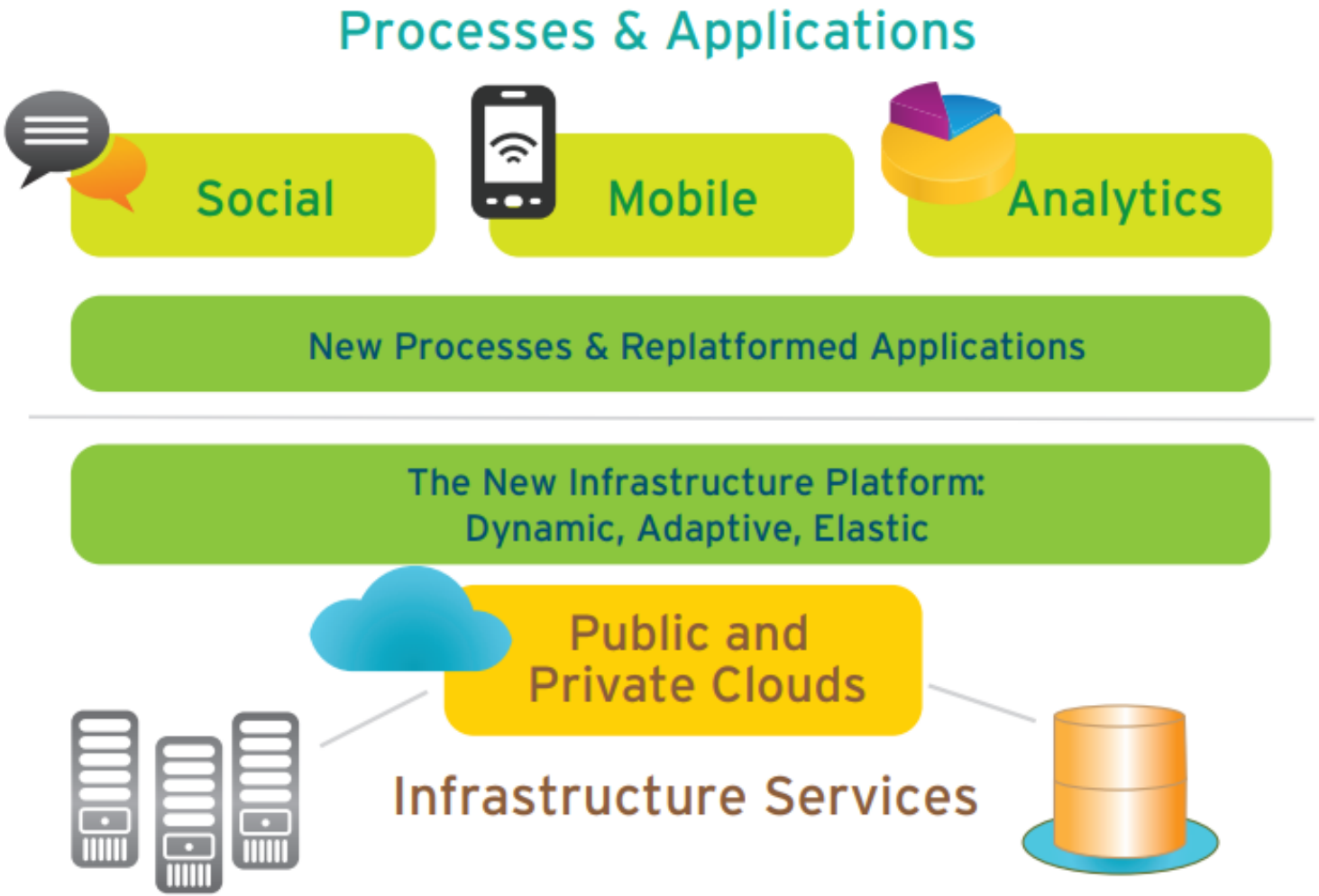
Problems to Solve for Large Enterprises

- Inherent rigidity
- Business processes
- Demanding customers
- Changing requirements
- Changing workload
- Cost efficiency
- New technologies

SMAC Stack

- Four critical technologies that have emerged and achieved prominence in the last decade:
  - **S**ocial: real-time and real-life collaboration
  - **M**obile: anytime, anywhere, location aware, sensor-based
  - **A**nalytics: advantage of big data generated by mobile, social, web and other sources smart algorithms, knowledge
  - **C**loud: shifting investments from services to platforms to infrastructure (Cap-Ex -> Op-Ex), more adaptive

Elastic Digital Architecture (EDA)



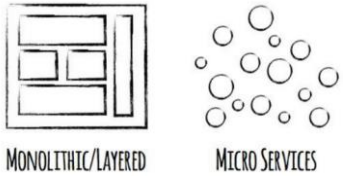
Key Elements of EDA

- Infrastructure
  - On-demand
  - Cloud
- Technology
  - Database
  - Backend
  - Frontend
- Services
  - Modular
  - Support different client technologies
  - Prepare for changes
- Architecture

Microservices

Microservice Architecture

- Context:
  - Server-side enterprise application
  - Must support a variety of different clients (desktop, browser, mobile, 3rd parties)
- Structures an application as a collection of loosely coupled services
- Enables the continuous delivery/deployment
- Large and complex applications
- Enable to evolve the organization's technology stack



What are microservices and why they take over

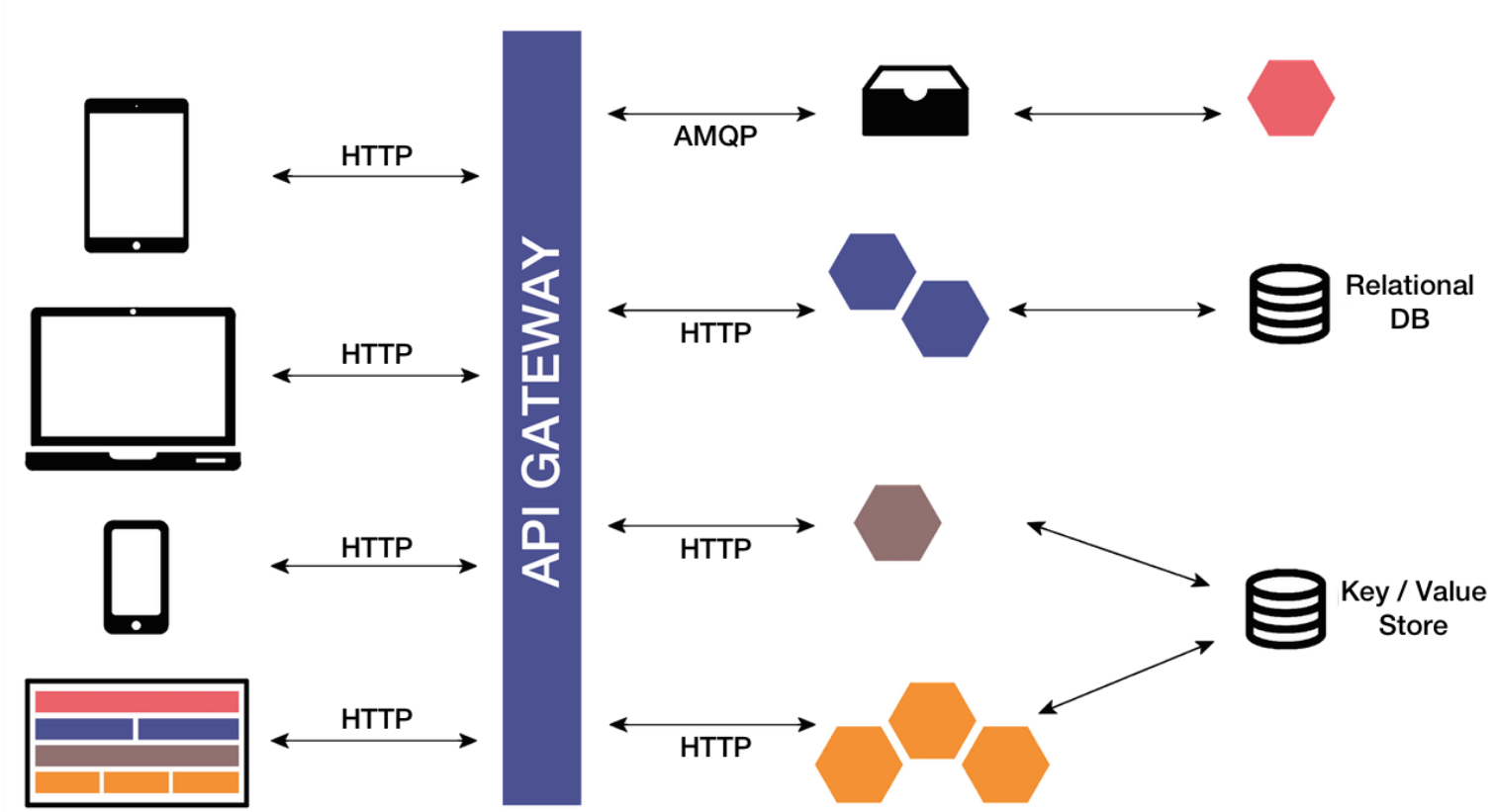
- From the Monolith through SOA to Microservices
  - > Monolithic (1990s and earlier): Tight coupling
  - > SOA (2000s): Looser coupling
  - > Microservices (2010s): Decoupled

Forces

- There is a team of developers working on the application
- New team members must quickly become productive
- The application must be easy to understand and modify
- You want to practice continuous deployment of the application
- You must run multiple copies of the application on multiple machines in order to satisfy scalability and availability requirements
- You want to take advantage of emerging technologies (frameworks, programming languages, etc.)

- **Solution**: Architecture that structures the application as a set of loosely coupled, collaborating service

Microservice Architecture Reference



Benefits

- Each microservice is relatively small
  - > Easier for a developer to understand
  - > IDE is faster
  - > The application starts faster and speeds up deployments
- Each service can be deployed independently (versions)
- Easier to scale development and teams
- Improved fault isolation
- Each service can be developed and deployed independently
- Eliminates any long-term commitment to a technology stack

Drawbacks

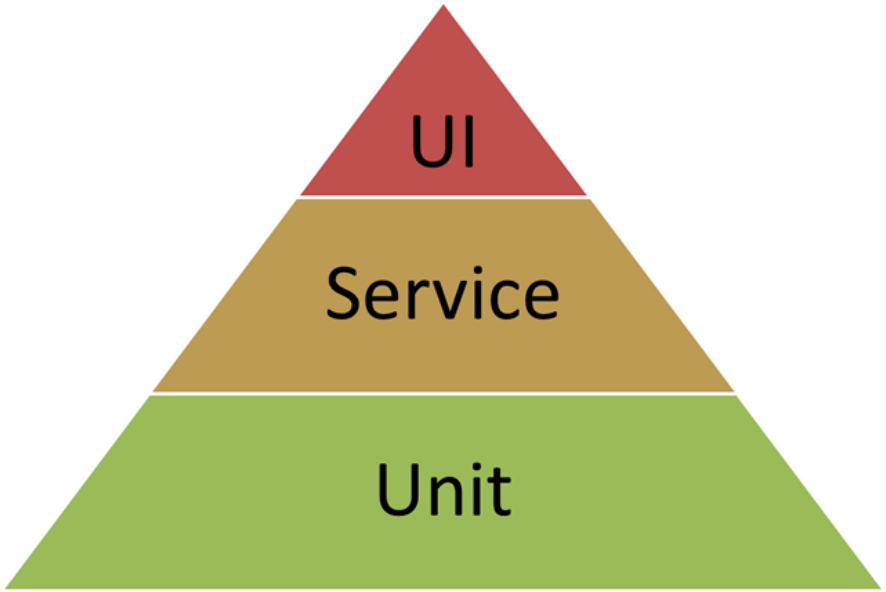
- Additional complexity of creating a distributed sys
  - Developer tools/IDEs are not for developing distribu applications
  - Testing is more difficult
  - Implement inter-service communication mechanism
  - Implementing use cases that span multiple services without using distributed transactions is difficult
  - Implementing use cases that span multiple services requires careful coordination between the teams
- Deployment complexity
- Increased memory consumption:
  - Replaces N monolithic application instances with NxM services instances (M times as many JVM runtimes)

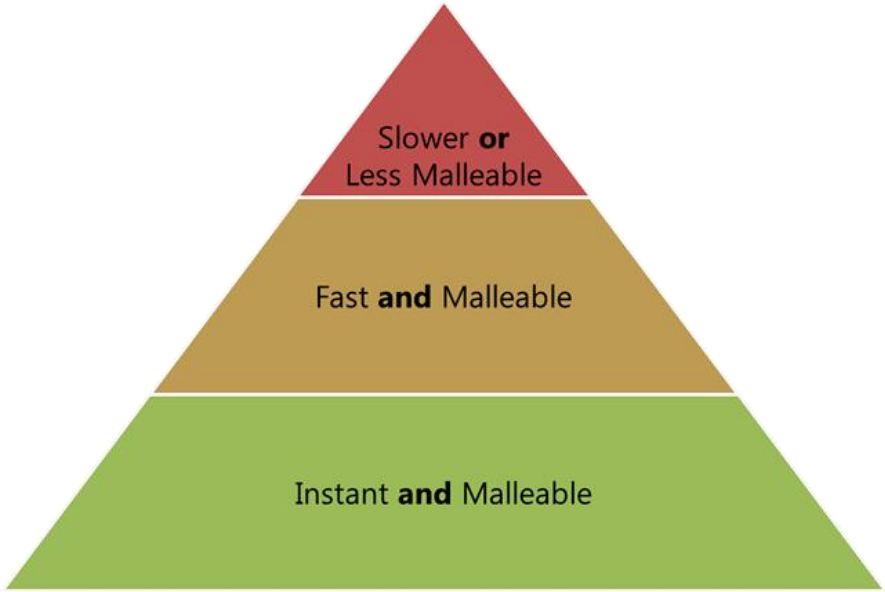
Testing

- Unit tests:
  - Write test code to verify the rightness of very tiny parts of the code
- UI testing:
  - Execute UI operations (like click) from test and see how they work

Mike Cohn test pyramid

- UI test: expensive, slow and harder to maintain
- Unit test: efficient, quick and easy to maintain





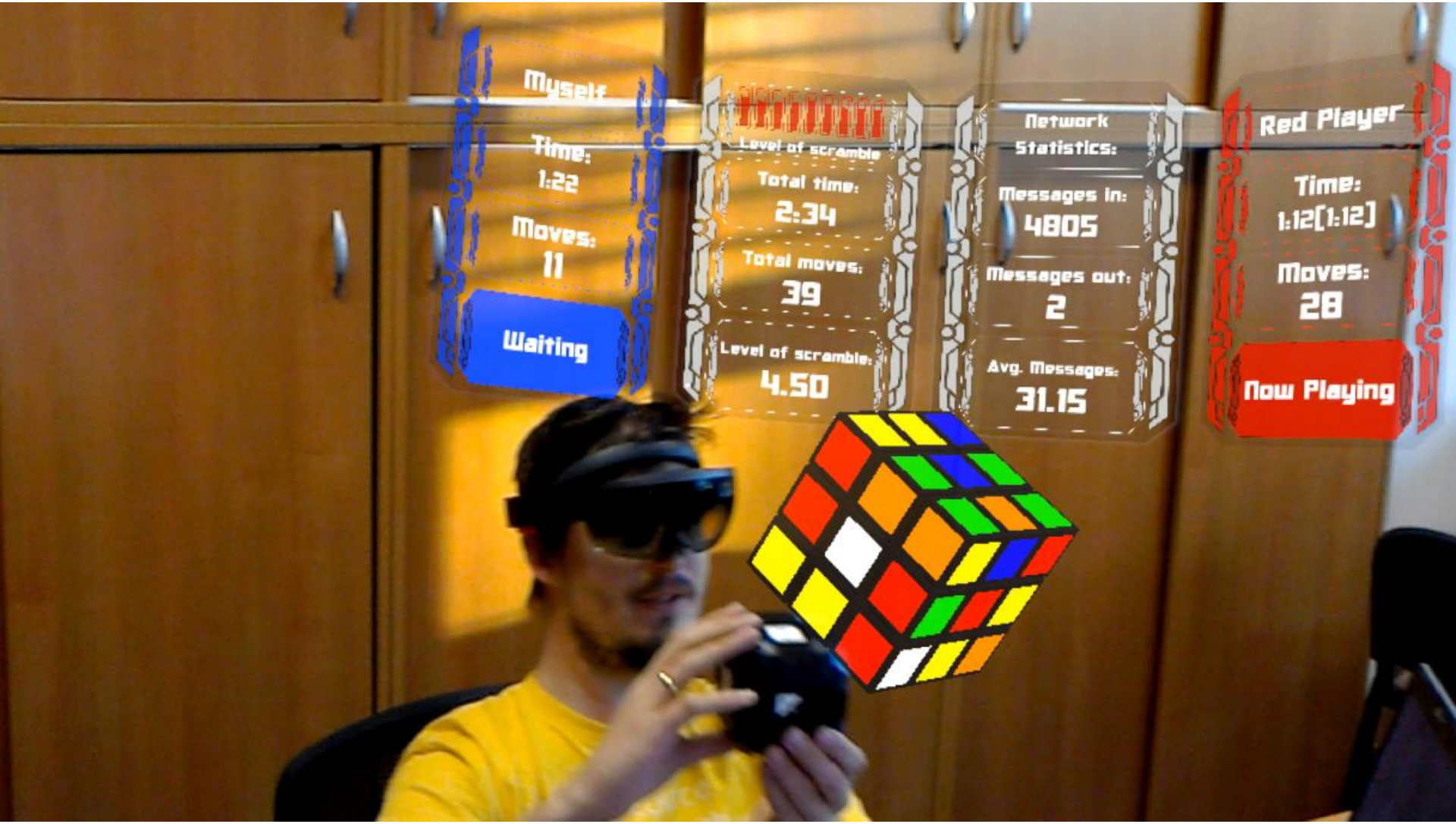
What is the real power of the software?

- The software can dramatically change the capabilities of a system
- Today, 99.9% of the systems contain some software component
- The amount of data is exploding
- The role of effective algorithms remains outstanding
- Huge responsibility on software developers
- A tiny bug can have a finite consequence
- The role of testing is becoming more and more important

Outline

- Facts
- Trends
- Tools
- Communication
- Software – the power in industry
- Applications, applications, and applications…

5G Digital Cube

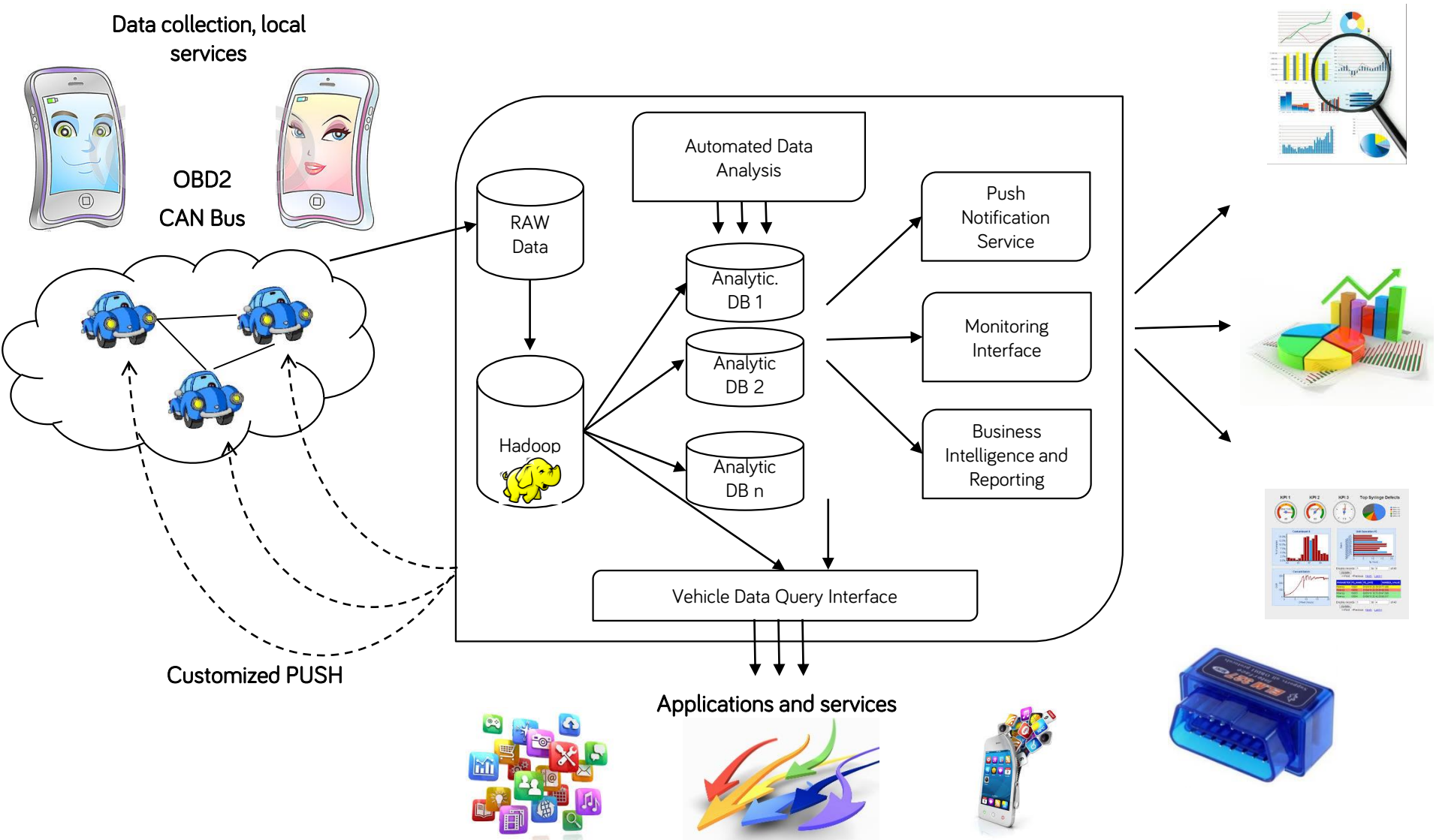


Audi Q7

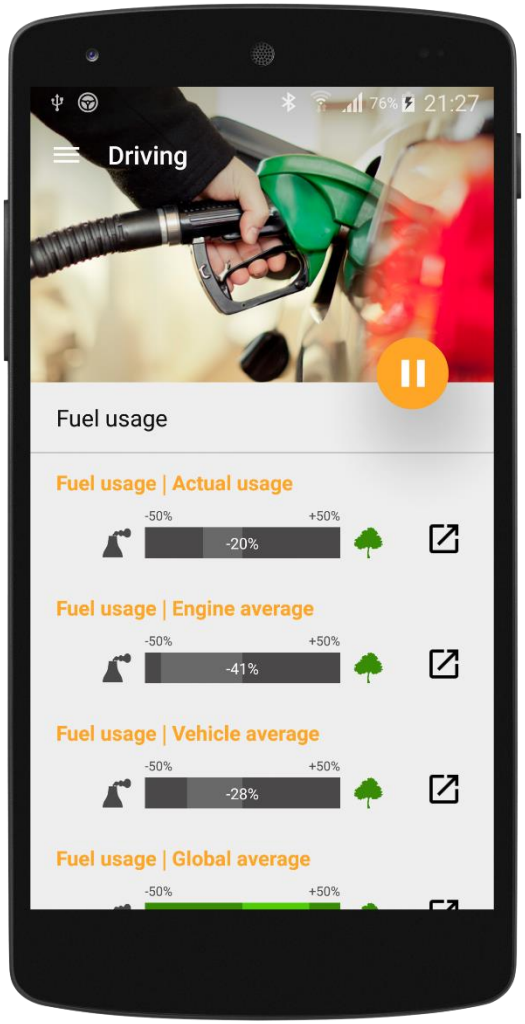
VehicleICT

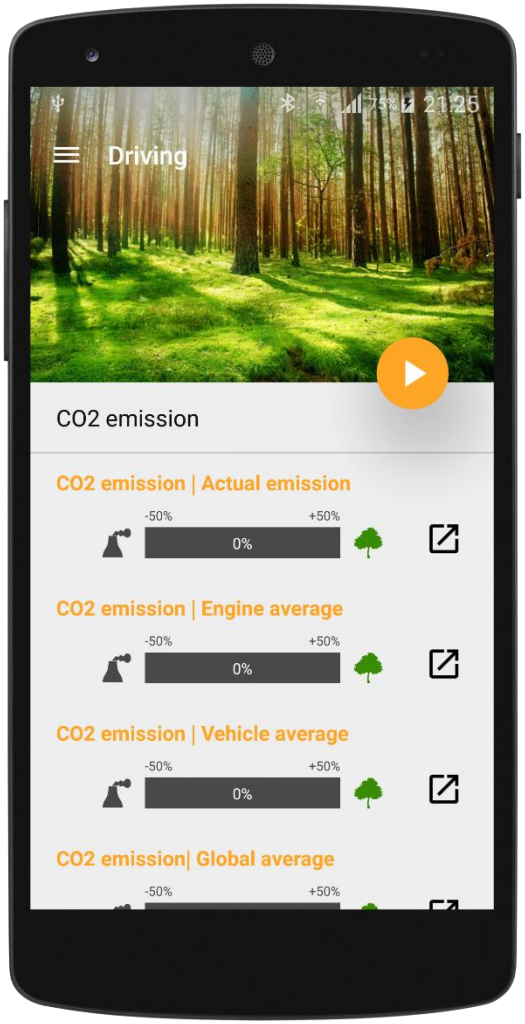
BME AUT
Automatizálási és
Alkalmazott
Informatikai Tanszék

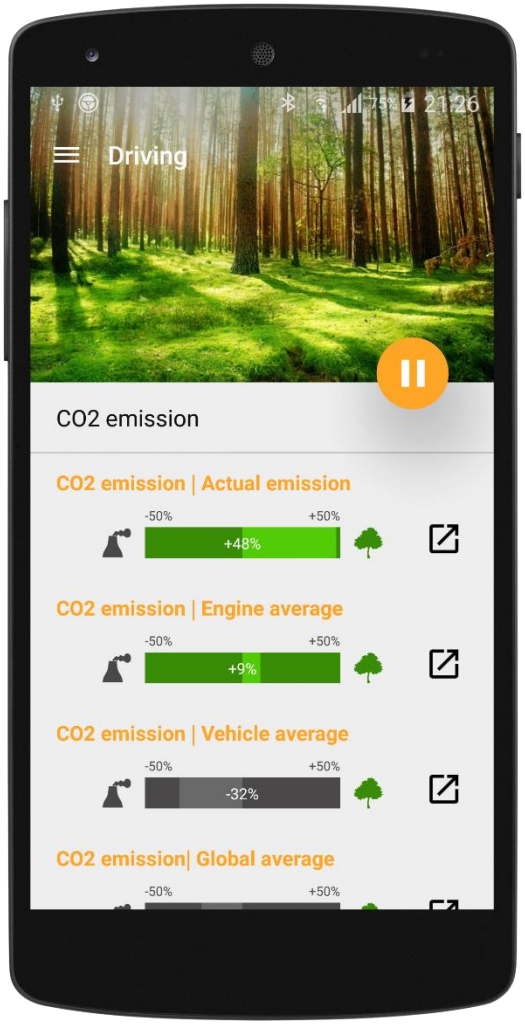
VehicleICT


Data collection, local services
OBD2
CAN Bus
RAW Data
Automated Data Analysis
Analytic. DB 1
Analytic DB 2
Analytic DB n
Hadoop
Push Notification Service
Monitoring Interface
Business Intelligence and Reporting
Vehicle Data Query Interface
Customized PUSH
Applications and services

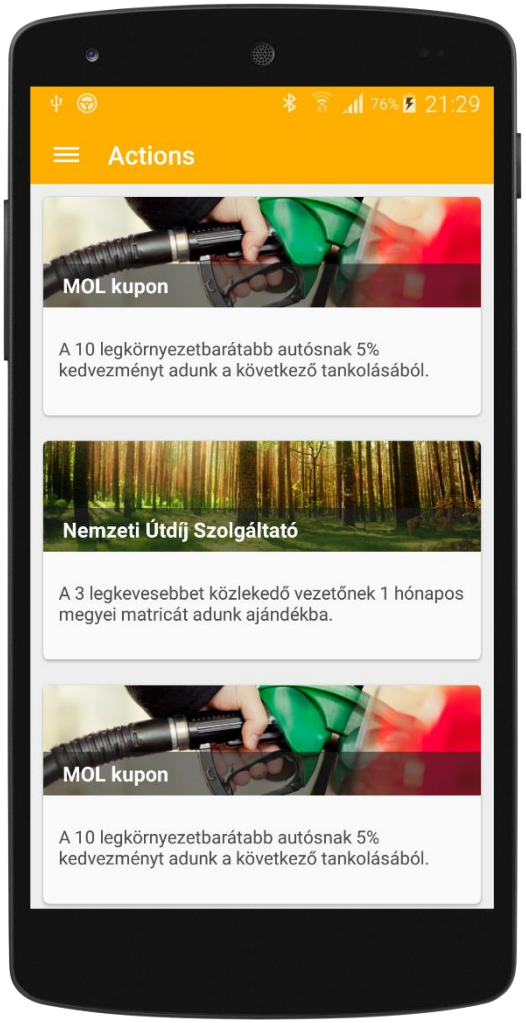
Fuel, CO_2

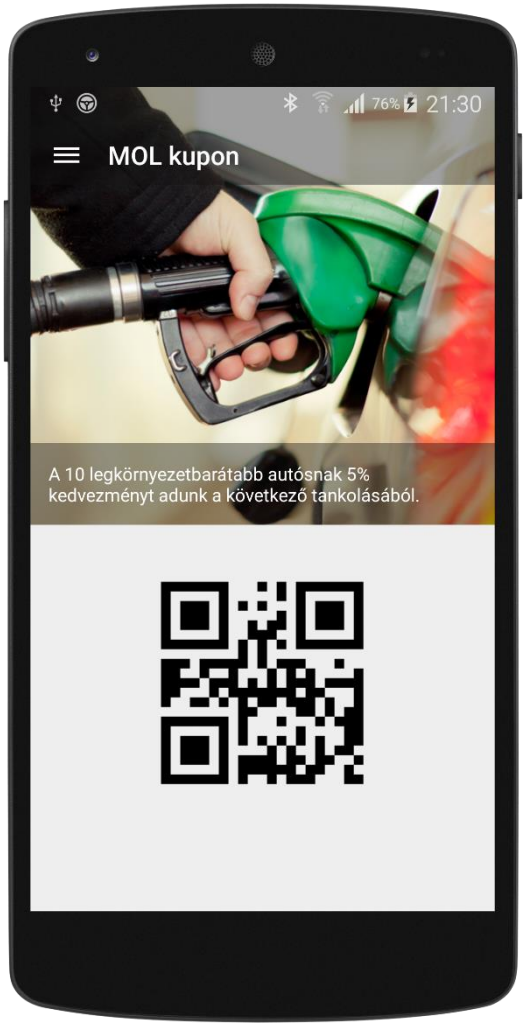

Driving
Fuel usage
Fuel usage | Actual usage
-50%
+50%
-20%
Fuel usage | Engine average
-41%
Fuel usage | Vehicle average
-28%
Fuel usage | Global average


Driving
CO2 emission
CO2 emission | Actual emission
0%
CO2 emission | Engine average
CO2 emission | Vehicle average
CO2 emission| Global average

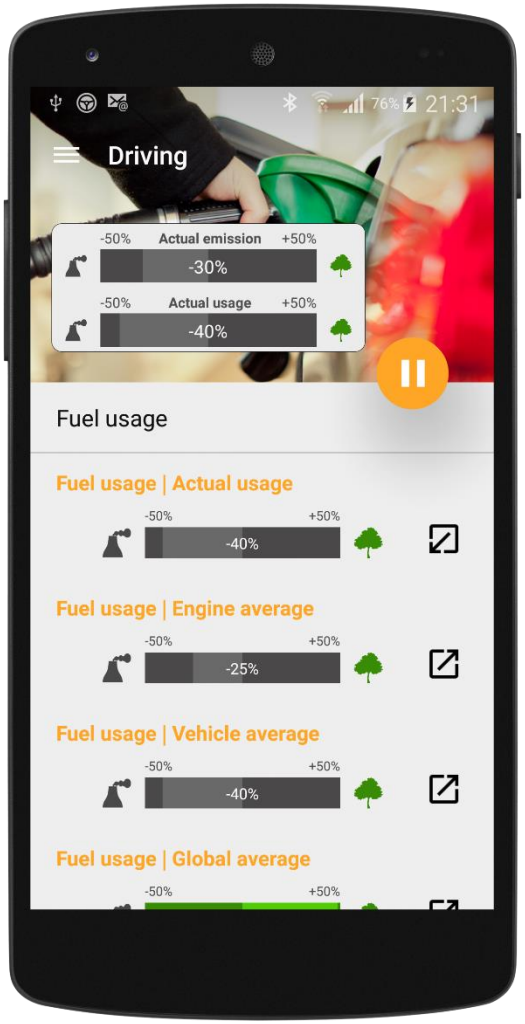

Driving
CO2 emission
CO2 emission | Actual emission
+48%
CO2 emission | Engine average
+9%
CO2 emission | Vehicle average
-32%
CO2 emission| Global average

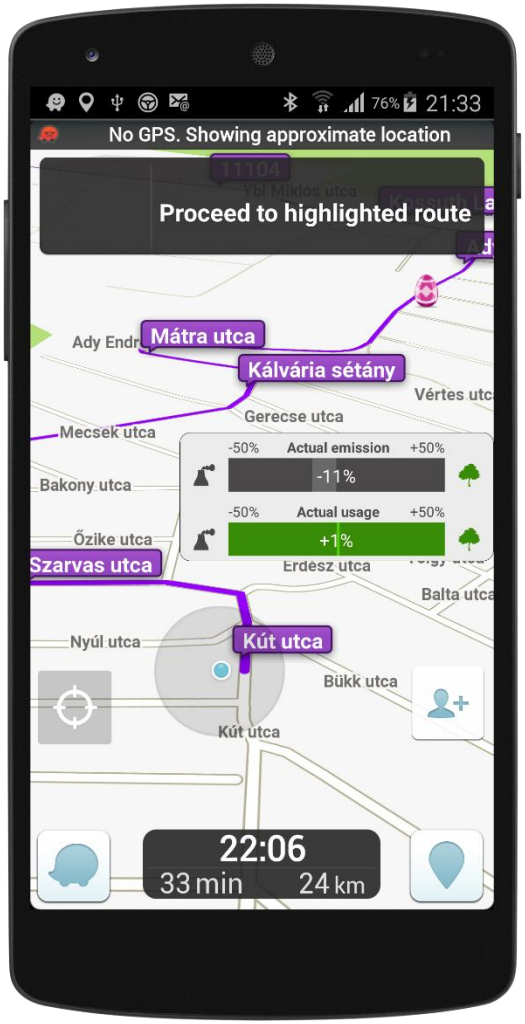
Live map, location-based notification

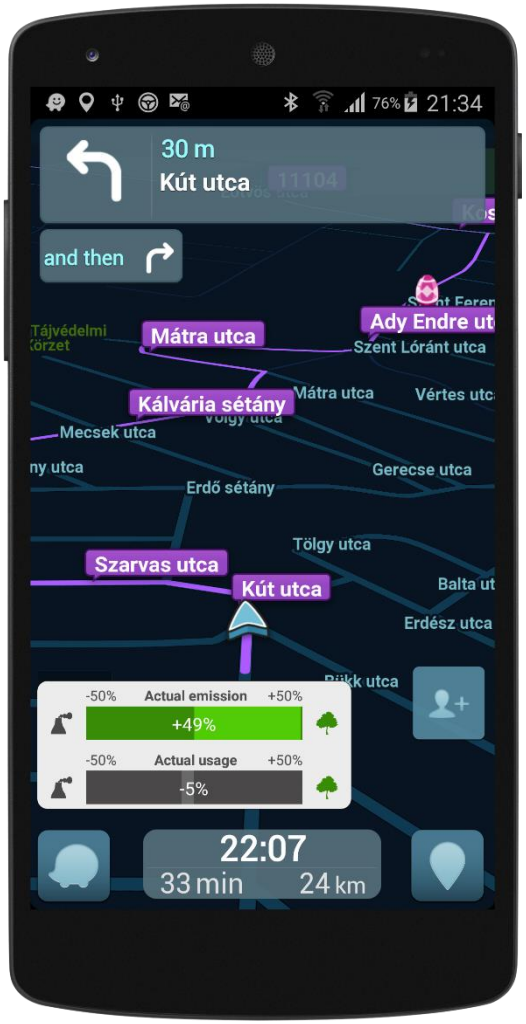




Integration into navigation applications







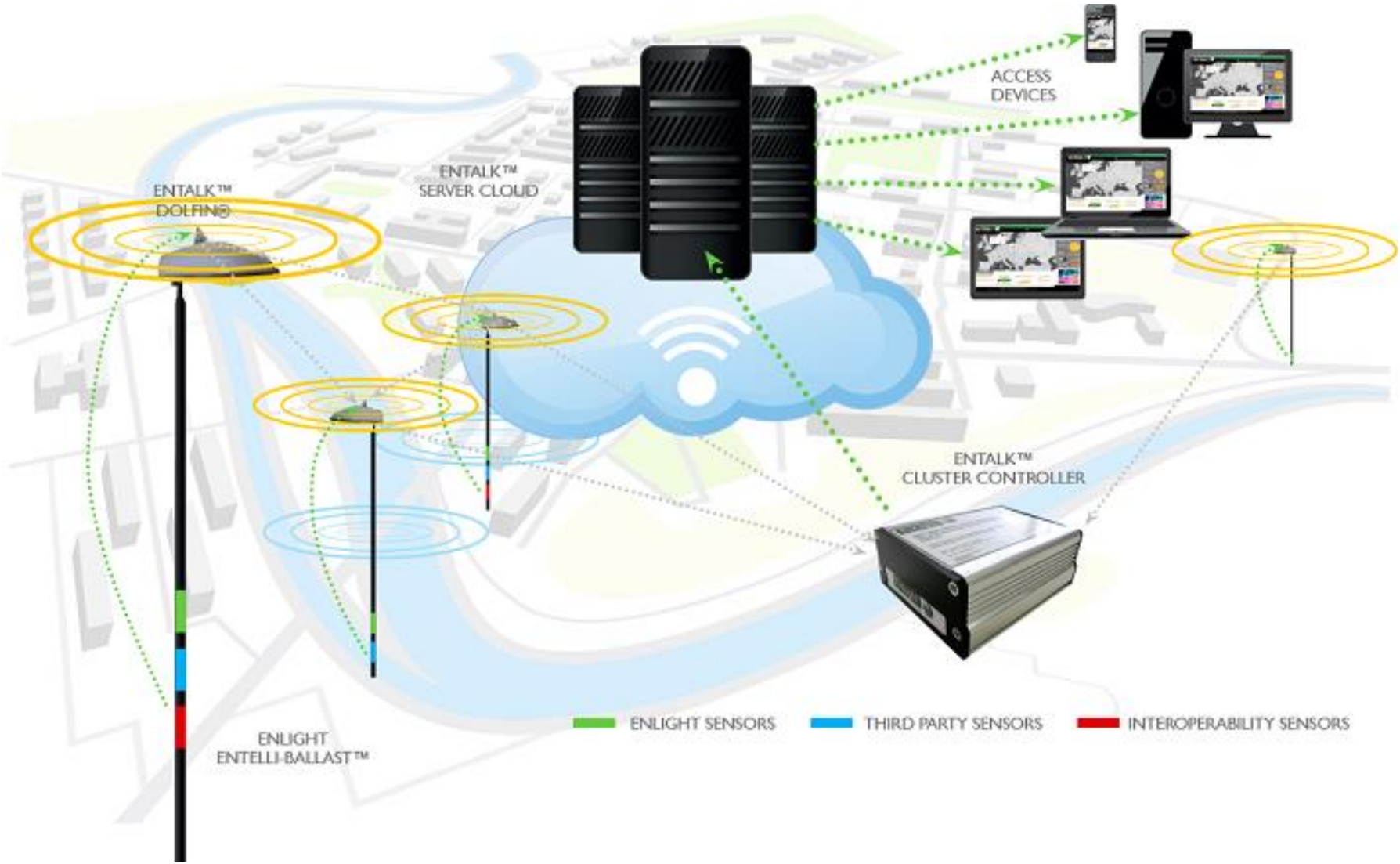
SocialDriving – smart watch

SOLSUN

Smart city

BME AUT
Automatizálási és
Alkalmazott
Informatikai Tanszék

SOLSUN


ACCESS
DEVICES
ENTALK™
SERVER CLOUD
ENTALK™
DOLFIN®
ENTALK™
CLUSTER CONTROLLER
ENLIGHT
ENTELLI-BALLAST™
ENLIGHT SENSORS
THIRD PARTY SENSORS
INTEROPERABILITY SENSORS

SOLSUN – SmartCity domain

Sustainable Outdoor Lighting and Sensory Urban Network

sustainable smart city infrastructure

5 international partners

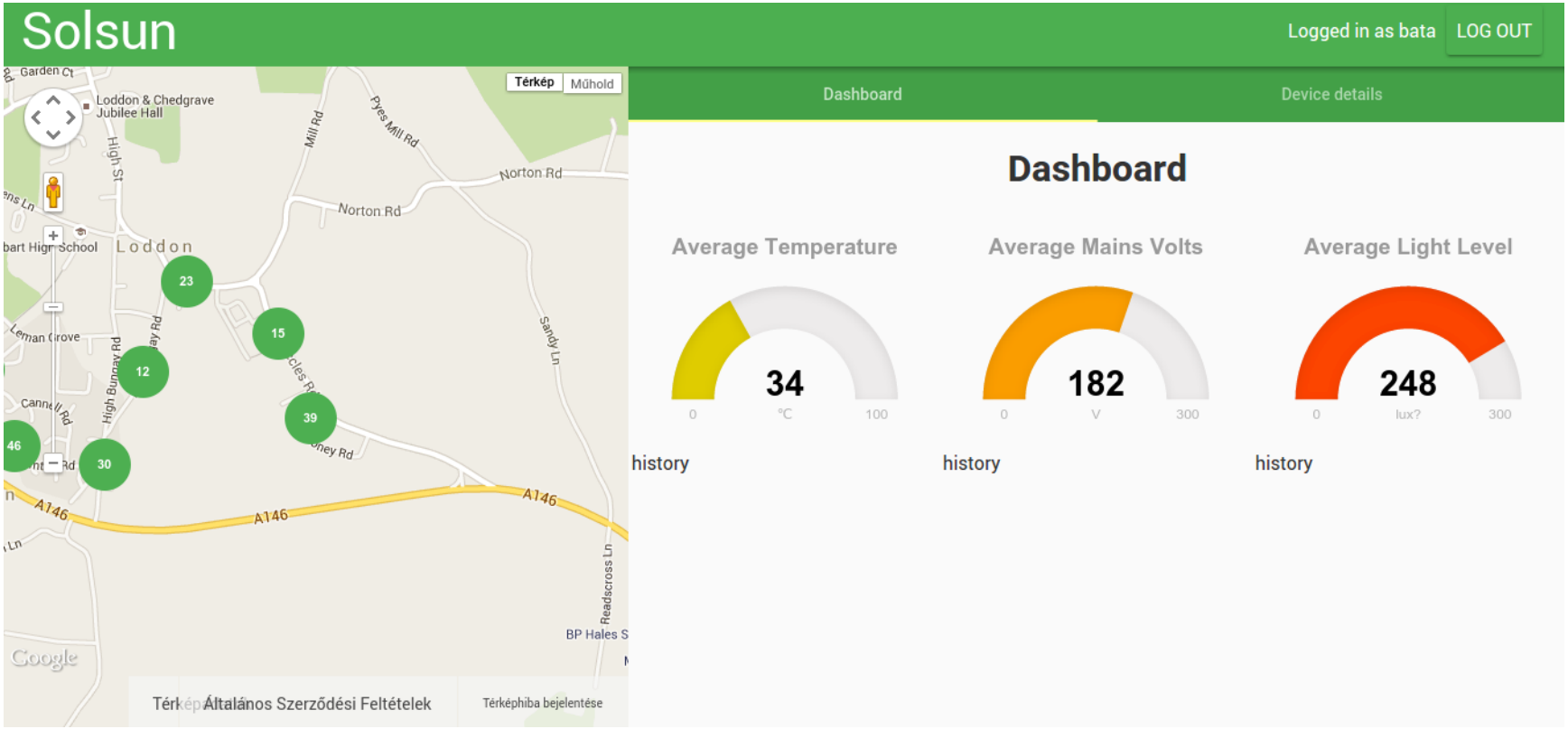
period 2015-2017

reduce energy consumption and air polution

EU project

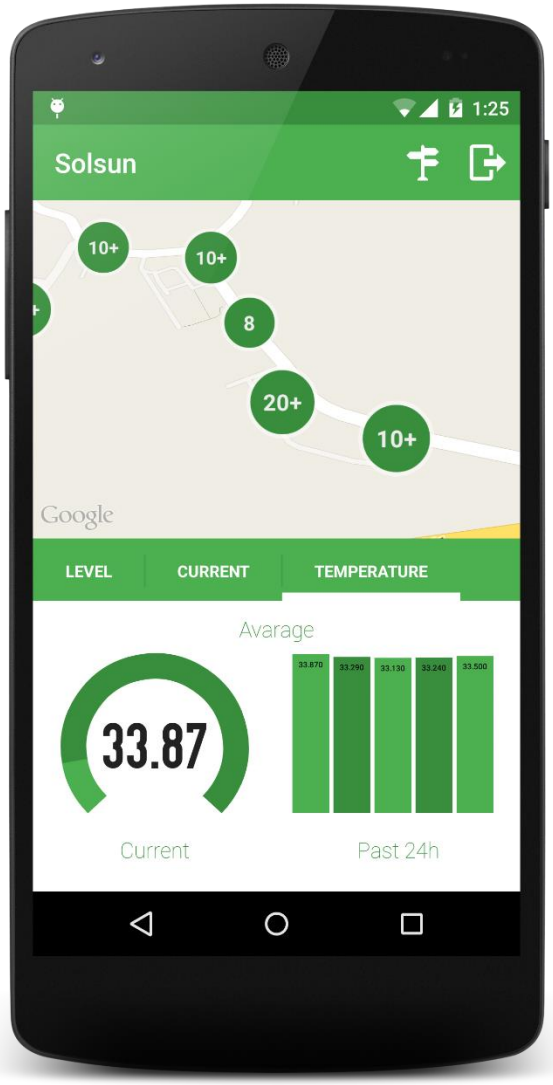
integrated technology platform

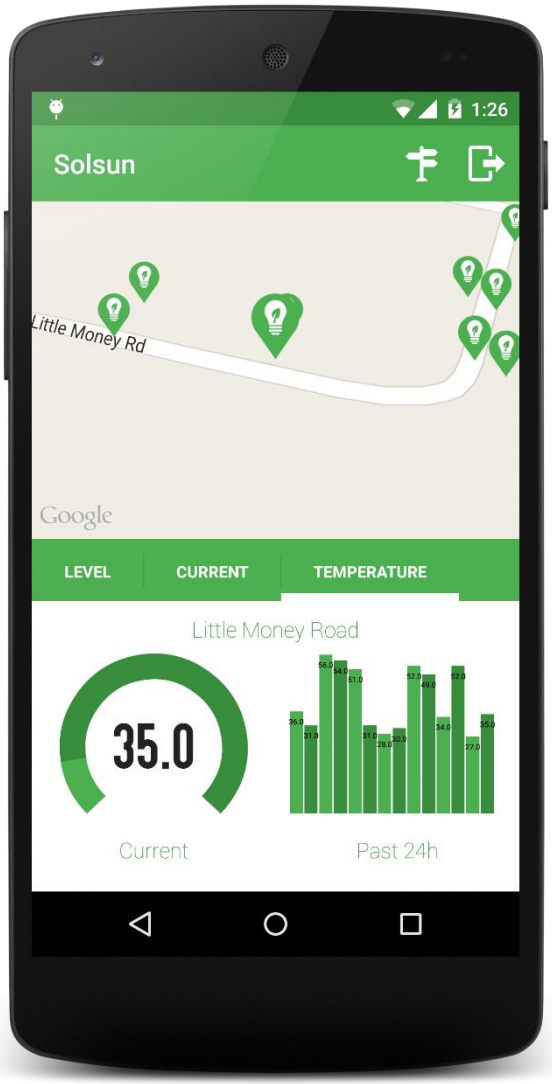
Sensors– web, Big Data



Mobile client

- Overview of sensors
- Real-time data
- History, statistics
- Route planning based on sensor values
- For example: recommend a less polluted route for cyclists





Summary

- Software is increasingly dominating the systems
- Software development has entered the 5th generation
- The 5th generation of communication provides new opportunities for software developers
- Care should be taken to keep the software code clean
- Successful software activities require well-trained teams
- Software research focuses on software reliability and development productivity

hassan@aut.bme.hu

www.aut.bme.hu

Thanks for : Prof. Frank Fitzek, Dr. László Lengyel, Dr. Péter Ekler, Dr. Bence Kővári , Domonkos Kiss, Gábor Simon, Attila Hideg for the slides,