IARIA
International Academy, Research, and Industry Association

NexTech 2017
Barcelona
November 12 – 16 2017

SOFTWARE THROUGH THE PERSPECTIVE OF HUMAN-CENTRIC COMPUTING PARADIGM

Evgeny Pyshkin

University of Aizu

Contributors

Natalia Bogach

Peter the Great St. Petersburg Polytechnic University

Vlatko Davidovski

Cognizant Business Consulting Strategy & Transformations

Vitaly Klyuev

University of Aizu

Maxim Mozgovoy

University of Aizu

Andrey Kuznetsov

T-Systems

Speaker

Senior Associate Professor, Ph.D., Doctoral Maru-Go

University of Aizu

Career: Software Engineer; Assistant Professor; Associate Professor; Senior Associate Professor ***Peter the Great St. Petersburg Polytechnic University***

ПОЛИТЕХ
Санкт-Петербургский политехнический университет Петра Великого

THE UNIVERSITY OF AIZU

会津大学

- Aizu-Wakamatsu (Japan's Tohoku, ~200000)
  - Rice growing and sake production
  - Samurai city
  - International university focused on computer technology

University of Aizu:
To Advanced Knowledge for Humanity

- Since 1993: First university in the region in modern times
- International Outlook **67.5** **1st** in Japan
- Citation **42.7** **8th** in Japan

THE WORLD UNIVERSITY RANKINGS

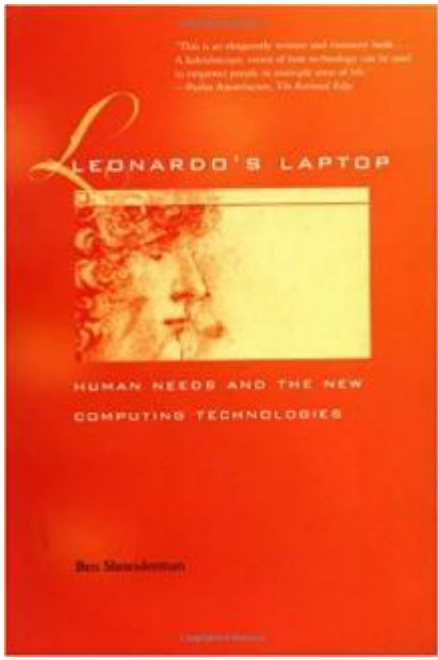
2017

23rd	601–800th
Japan University Rankings 2017	World University Rankings 2018

Charming Aizu

What is Human-Centric Computing

- Ben Shneiderman's *"Leonardo's Laptop: Human Needs and the New Computing Technologies"**
  - The focus is shifting from what computers can do to what users can do
  - A key transformation is to what he calls "universal usability," enabling participation of **young and old, novice and expert**, **able and disabled**
- Human Centered Computing (HCC) aims at bridging the gaps between the disciplines involved with the design and implementation of computing systems that support people's activities



What computers can do

What users can do

Universal usability

* Shneiderman B. Leonardo's laptop: human needs and the new computing technologies. Mit Press; 2003.

Insights into Human-Centric Computing

- Rapid transformation from HCI to its own distinctive research agenda
- Multiple links to digital transformation concepts
- Affecting society and individuals
- Bridging the gaps between different disciplines
- Towards more personalization and user collaboration

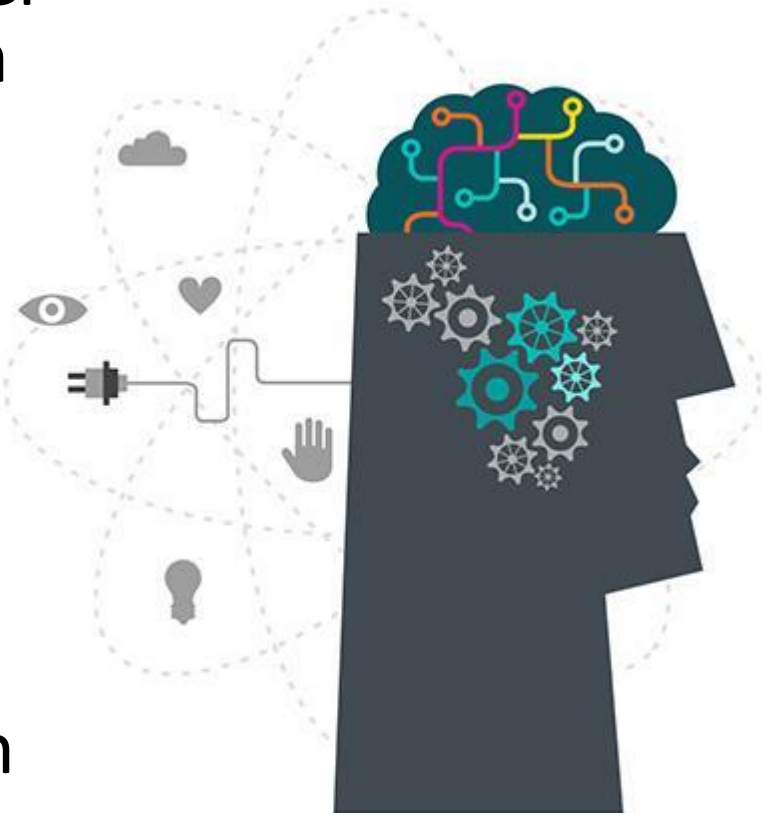


Image: http://icc.mtu.edu/hcc/

Inter- and even transdisciplinary nature of HCC

Applications and Transdisciplinarity

Numerous domains

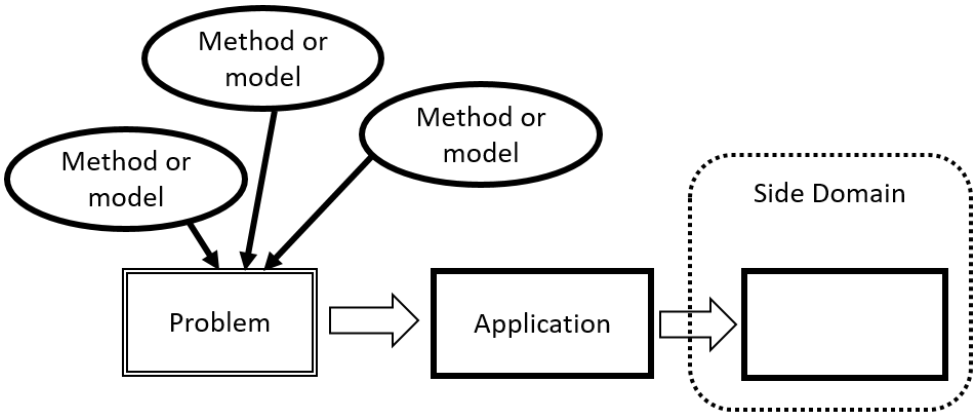
- Health informatics
- Learning and education
- Software engineering
- Internet-of-Things
- Geovisualizaton
- Multimedia networking
- GIS and traveling
- …

Transdisciplinary research is

- one part, a kind of interdisciplinary research involving **scientific and non-scientific** sources or practices,
- another part, a form of **collaborative learning and problem solving**.
  - It does not simply the **integrated use of techniques referring to various disciplines**, but:
  - **an application of solutions in one area of knowledge to a distinct human-centric domain.**

Convergence of inter- and transdisciplinary models

- Two important aspects (at least):
  - A cooperation of a variety of algorithms and approaches from different knowledge areas struggling with a certain societal problem or applied to a certain technology
  - A ***transition*** of the successful solutions or applications to a distinct application domain



(a) *Multi-disciplinary connections* ***(b)*** *Trans-disciplinary connections*

Review of Selected Ongoing Projects

From non-native GUI and image recognition to mobile farms:

- Improving testing quality and advancing methodology of resource and time-consuming testing automation focusing on mobile applications

From signal processing to language learning:

- Intonation acquisition and representation for the benefits of practical phonology

From direct interaction to a companion model:

- Improving location data accuracy for the benefits of depending applications

From engineering to liberal arts:

- Revisiting a case of software engineering education

Review of Selected Ongoing Projects

From non-native GUI and image recognition to mobile farms:

- Improving testing quality and advancing methodology of resource and time-consuming testing automation focusing on mobile applications

From signal processing to language learning:

- Intonation acquisition and representation for the benefits of practical phonology

From direct interaction to a companion model:

- Improving location data accuracy for the benefits of depending applications

From engineering to liberal arts:

- Revisiting a case of software engineering education

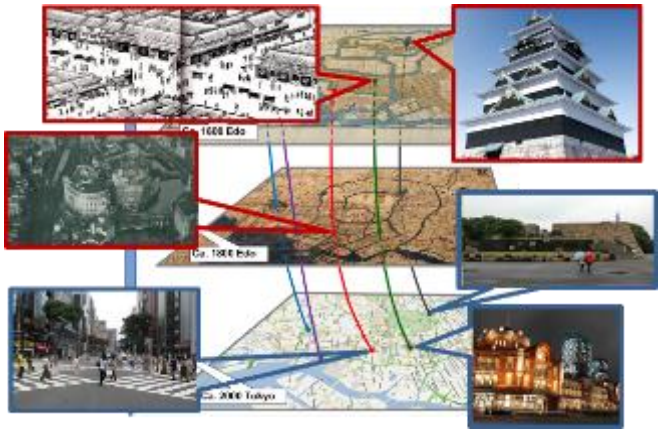
From Non-native GUI and Image Recognition to Mobile Farms

- Arranging automated smoke testing for big software projects
  - Very many tests running for each new build
  - Can be a part of a continuous integration pipeline

- Interesting case: mobile applications (including mobile games)
  - Resource-intensive applications, may crash or run slowly
  - May behave very differently on different devices
  - Often contains no native GUI controls. Contains elements that are easy to overlook with manual testing
  - Stress tests may be beneficial (such as running the app for 5 hours)
  - Real mobile devices required (not emulators / simulators)

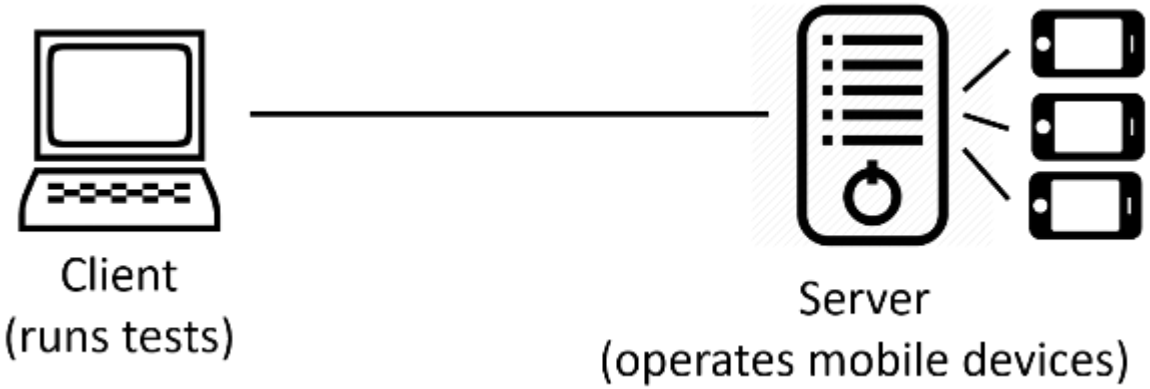
Addressing a Variety of HCC Domains

- **Resource-consuming dynamic applications such as mobile games** (such as ongoing *World of Tennis* project)
- **Mobile applications** with non-native user interface elements
- **Mobile components of travel-centric information systems** (by the example of a system supporting historical perspectives using interaction with graphical objects including maps and images)



Problem: Engineering the Tests

- How to design and run a GUI / smoke test on a mobile device?
- De facto standard approach:
  - Connect mobile devices to a “test server” running some special software
  - Execute testing scenarios on a remote machine via the server



Implementing Server-Side Software

- Example: using Appium:
  - Can be installed on Mac / Windows / Linux
  - Supports test scripts written in Java / JavaScript / Python / C# / etc.
  - Can test Android, iOS and Windows applications

```
wait 10 sec
tap location (100, 50)
assert that OK button
appears
press OK button
```

Problem: **in a case of applications that do not rely on the natively rendered GUI components of an underlying operating system and do not use standardized GUI libraries this approach does not work**

Related Research

- Take screenshots with *Appium* and use *OpenCV* to recognize areas via `matchTemplate()` function
  - **100% similarity cannot be achieved** due to screen rendering differences.
  - **Pattern recognition process is relatively slow**
  - An important research problem is **to find optimal parameters of image recognition algorithms (for example, provided by OpenCV library), in order to maximize GUI elements recognition similarity**

Review of Selected Ongoing Projects

From non-native GUI and image recognition to mobile farms:

- Improving testing quality and advancing methodology of resource and time-consuming testing automation focusing on mobile applications

From signal processing to language learning:

- Intonation acquisition and representation for the benefits of practical phonology

From direct interaction to a companion model:

- Improving location data accuracy for the benefits of depending applications

From engineering to liberal arts:

- Revisiting a case of software engineering education

From Signal Processing to Language Learning

- Computer-assisted language learning
  - *(considered generally)* CALL bridges the gap between learning environments and real-life projects
  - *(in language learning)* CALL bridges the gap between classroom language study and real-life communication
  - *(with respect to IT)* Language learning techniques form a symbiotic relationship with the current computer technology

www.upsidelearning.com

From Signal Processing to Language Learning

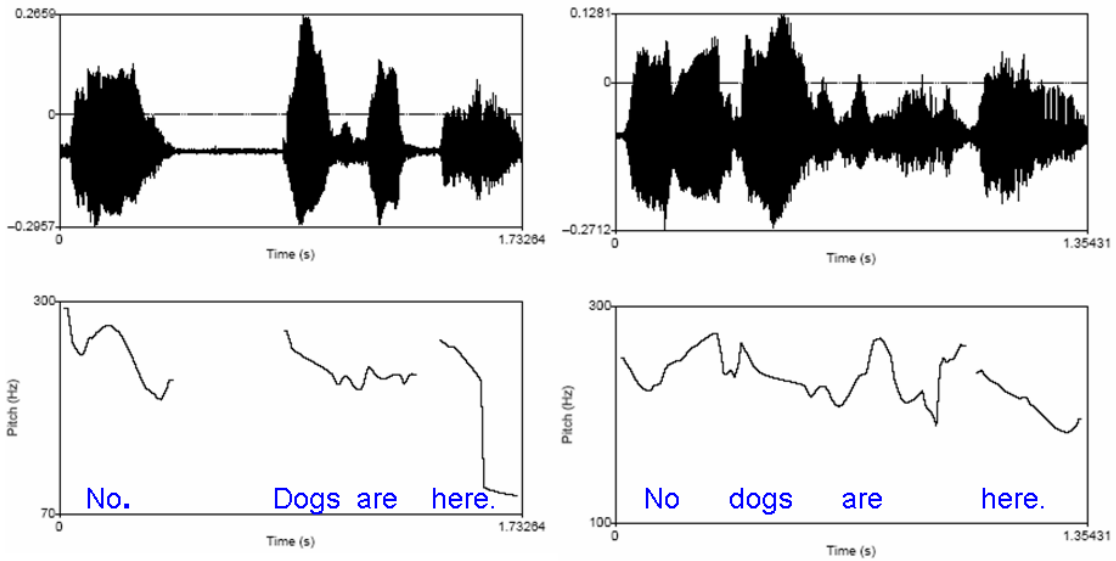
- Computer-assisted language learning?
  - CALL should not be limited to mere digitalization of learning process by transferring traditional techniques of managing language-related data with the use of computers. It should create **totally new use cases that are hard to implement without computer technologies***

:tps://www.go-fl.com/teflblog/computer-assisted-nguage-learning-call/

* E. Pyshkin, M. Mozgovoy, and V. Volkov. "Models and metamodels for computer-assisted natural language grammar learning," Int. J. of Social, Management, Economics and Business Engineering, vol. 9(1), 60–65, 2015.

Using Prosody Features while Learning

- Language learning
  - Vocabulary
  - Reading
  - Translation
  - Grammar
  - Writing
  - Conversation
  - **Prosody**
    - Particularities of non-verbal speech features, such as stress, rhythm, and intonation
    - Poor prosody could make speech unintelligible and incomprehensible*
    - Prosody can be used for meaning disambiguation**



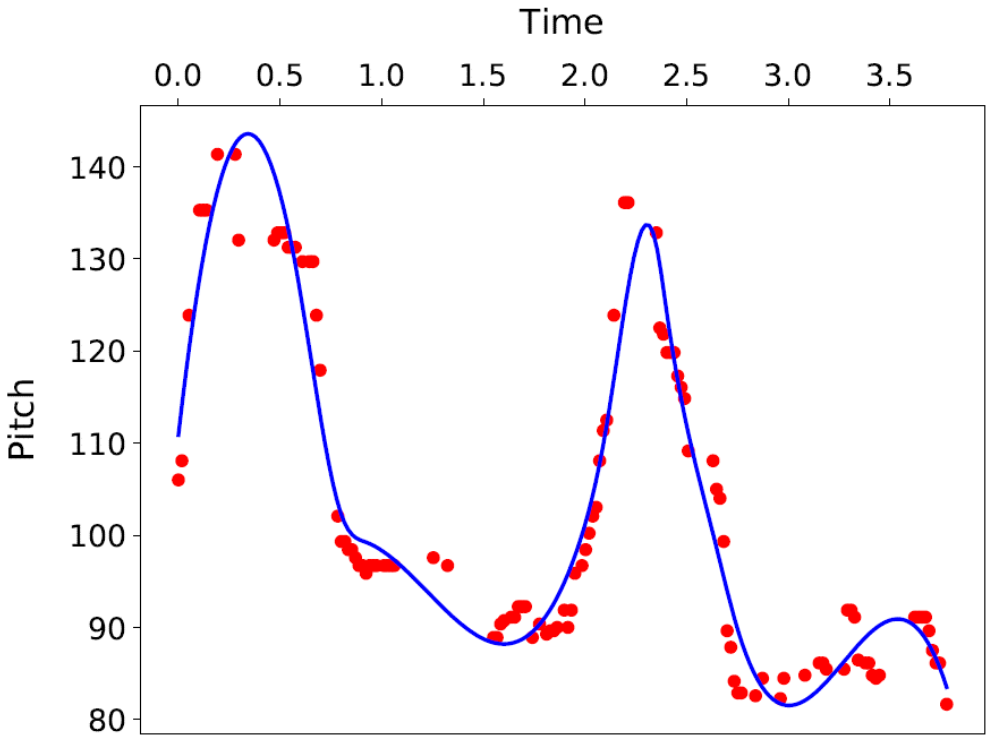
http://acoustics.org/pressroom/httpdocs/152nd/Liu.html.

* I. Lezhenin, A. Zhuikov, N. Bogach, E. Boitsova, and E. Pyshkin, "PitchKeywordExtractor: Prosody-based Automatic Keyword Extraction for Speech Content," FedCSIS 2017.

** Y. Liu, "More Than Words Can Say: Using Prosody to Find Sentence Boundaries in Speech," ASA/ASJ Joint Meeting, 2006, http://acoustics.org/pressroom/httpdocs/152nd/Liu.html.

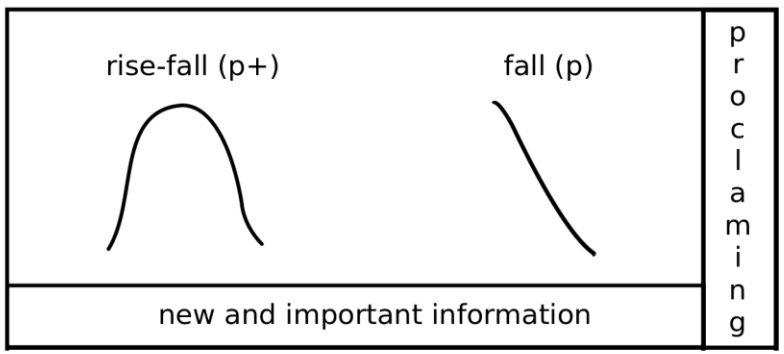
How to Introduce Prosody Learning to a Classroom?

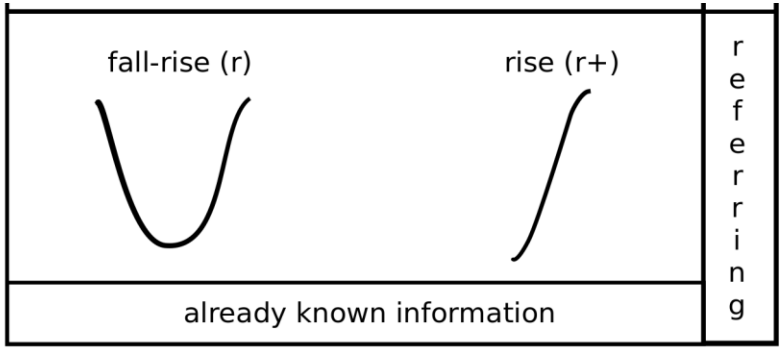
- Advancing pronunciation training organization
- We have to address many practical issues of pitch processing:
  - pitch detection
  - pitch approximation
  - pitch visualization
  - pitch estimation



How to Introduce Prosody Learning to a Classroom?

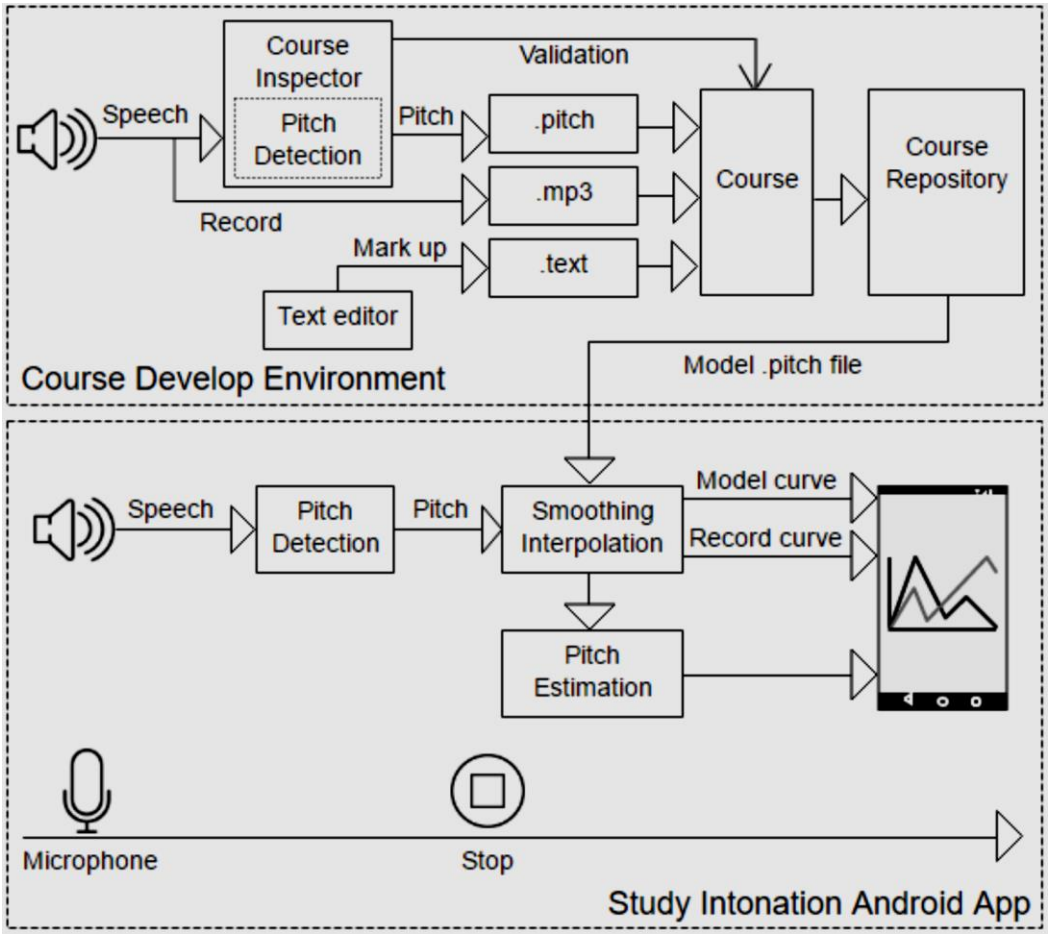
- We have to pay attention to modeling intonation styles and presenting them to a learner
  - Algorithms of signal processing meet problems of practical phonology
  - Particularly promising for many the languages where the stress is mainly phrasal (like in French) or tonic (like in Chinese)

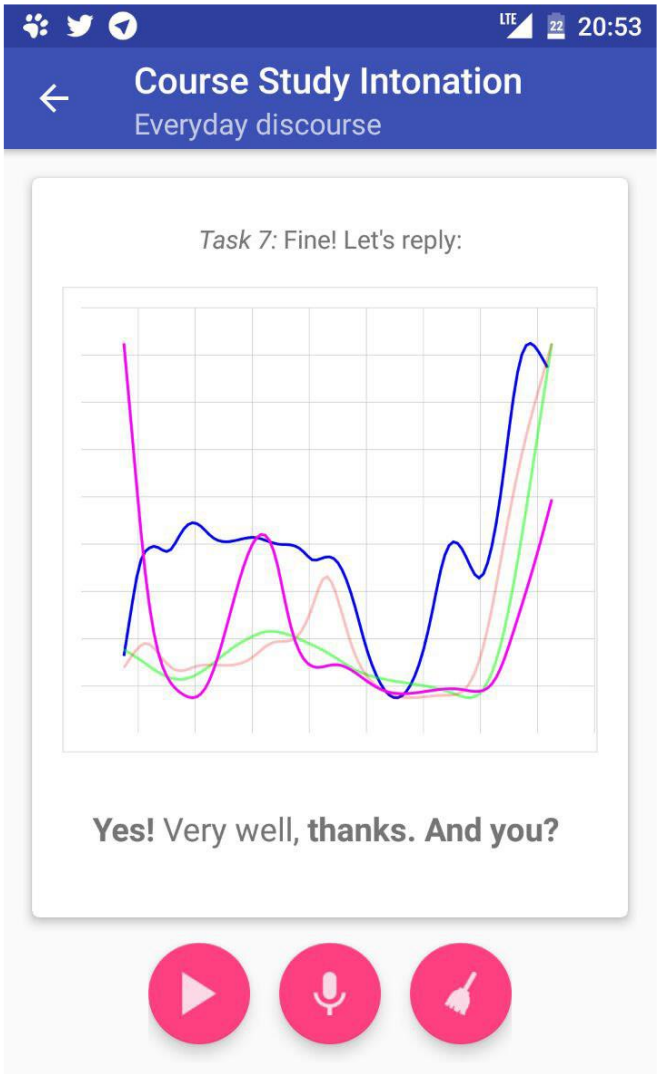




Implementations and Applications

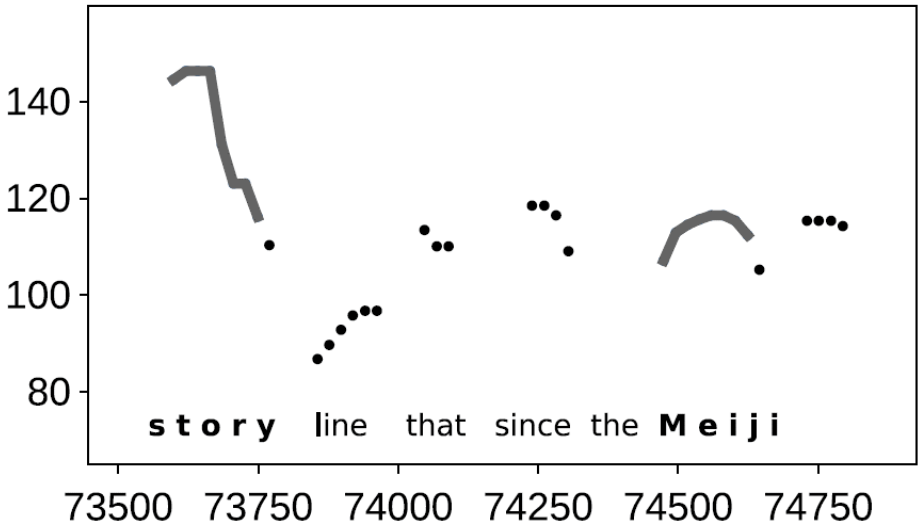
- Android app for learners

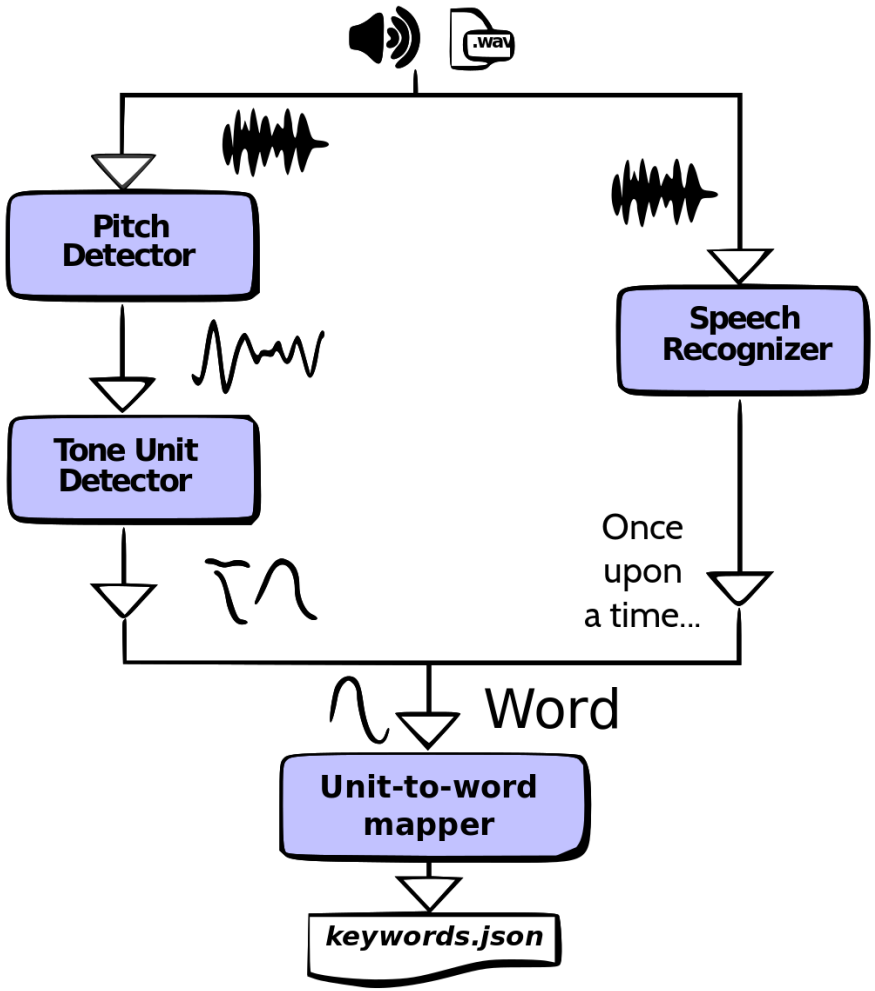




Implementations and Applications

- Keyword extraction
  - Audio and audiovisual content
  - Information indexing





Review of Selected Ongoing Projects

From non-native GUI and image recognition to mobile farms:

- Improving testing quality and advancing methodology of resource and time-consuming testing automation focusing on mobile applications

From signal processing to language learning:

- Intonation acquisition and representation for the benefits of practical phonology

From direct interaction to a companion model:

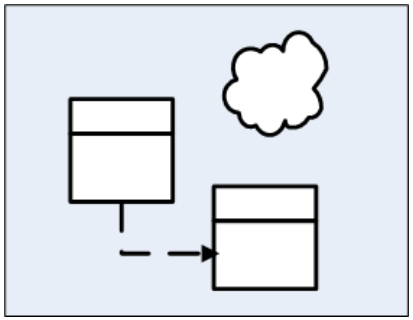
- Improving location data accuracy for the benefits of depending applications

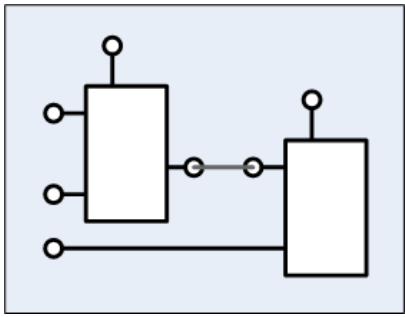
From engineering to liberal arts:

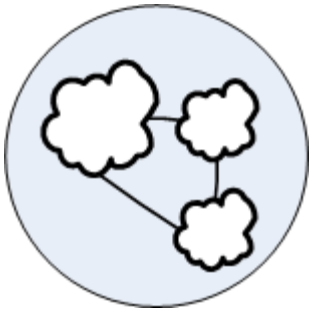
- Revisiting a case of software engineering education

From Direct Application Interaction to a Companion Model

- Application interaction models
  - “Embedded” interfaces
  - Component architecture
  - Service architecture



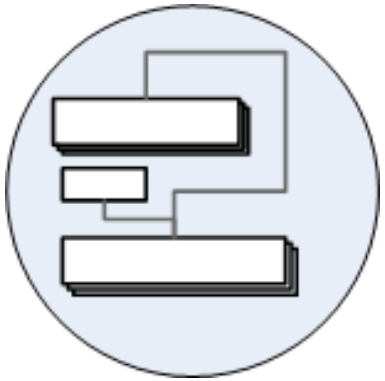




- Open source workflow
- Application cooperation
  - Two applications do not interact directly between each other, but an existing one benefits from the fact that a companion is running

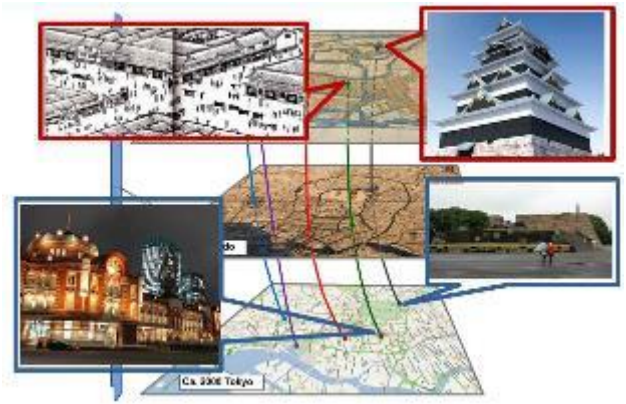
Companion Model: Use Case

- Application cooperation
  - Use case: Google Timeline
  - Combining a general-purpose application and an intelligent assistant aimed at managing location data quality



- Collecting information about user location within the context of traveling
  - This context explains specific requirements for location precision as well as scenarios how the collected data are used

Managing Location Data Issues



- Most of the traveler-oriented scenarios (in regards to post-travel experience) are about timing and location
- Surprisingly, exact and accurate location data might sometimes become even more important in after-journey activities, than during the trip

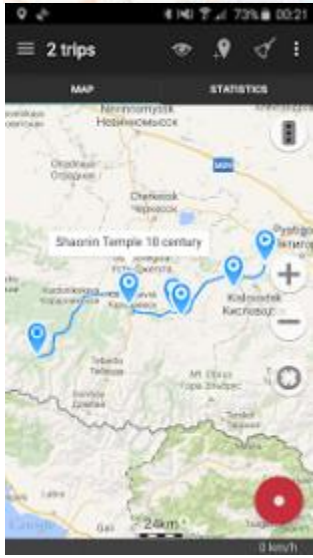
- On the base of precise traveler's location information, time, many other parameters (like a place name, city, country, etc.) can be calculated
- These calculated parameters can be used to annotate pictures, notes and other records, even those which are produced by location unaware legacy devices (e.g. cameras with no built-in GPS sensor)

What Does A Companion App Do

- The companion takes care about location data accuracy, while the other applications focus on their main use cases (like putting pictures on the map or building photo albums for each visited place)
  - In doing so we:
    - separate concerns of the different components,
    - improve user experience by providing a more accurate location when it’s needed, and
    - save battery energy

Why not “Greenfield”?

- Capturing user location information is to use traveler’s device (such as a smartphone or a tablet) as a tracker
  - Examples:
    - **Maps.Me***
    - **geotracker****
    - **GPS Tracker*****
  - Advantages:
    - A developer has full control on data collection and data
    - analysis processes
    - Location information is registered at the desired rate
    - Location data are sent to a storage using the format that fits particular use case and that can be used further in order to implement new features of the application



* https://maps.me/
** https://play.google.com/store/apps/details?id=com.ilyabogdanovich.geotracker
*** https://play.google.com/store/apps/details?id=com.app.LiveGPSTracker

Why not “Greenfield”?

- Capturing user location information is to use traveler’s device (such as a smartphone or a tablet) as a tracker
  - Significant drawbacks:
    - Battery life issue

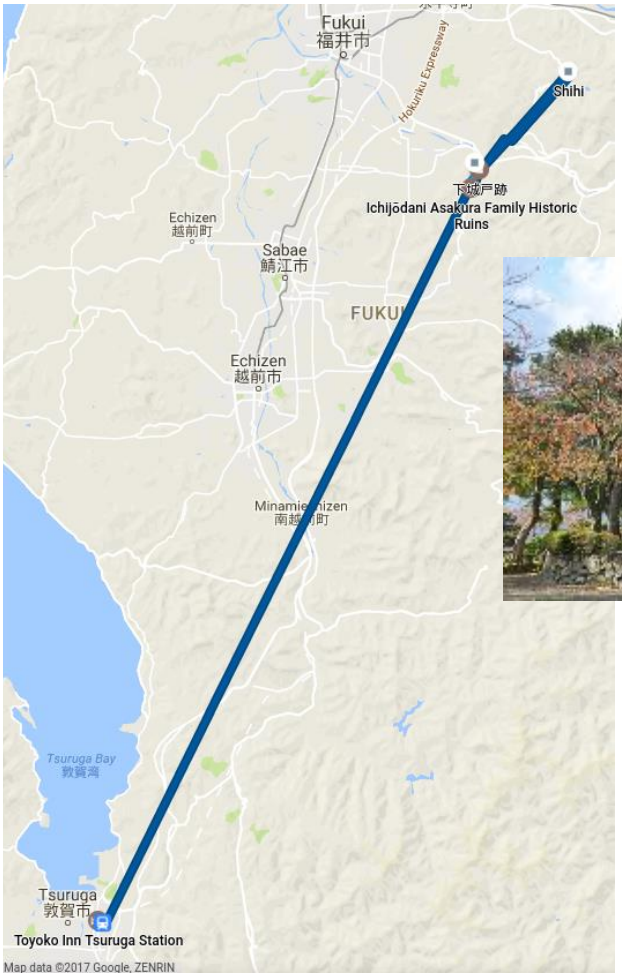
      GPS sensors consume much energy
    - Development efforts

      Application features have to be supported for the entire application life
    - Duplicate functions and ignored user habits

      There could be applications already installed on the user’s device, and the user might not be satisfied to be forced to install yet another application for collecting the GPS data and drawing the tracks on the map
    - Interoperability with other applications and services

      We could expect that additional applications (supporting advanced use cases such as detecting visited places and asking the user for a review) would like to benefit from the tracks captured by the GPS tracker.

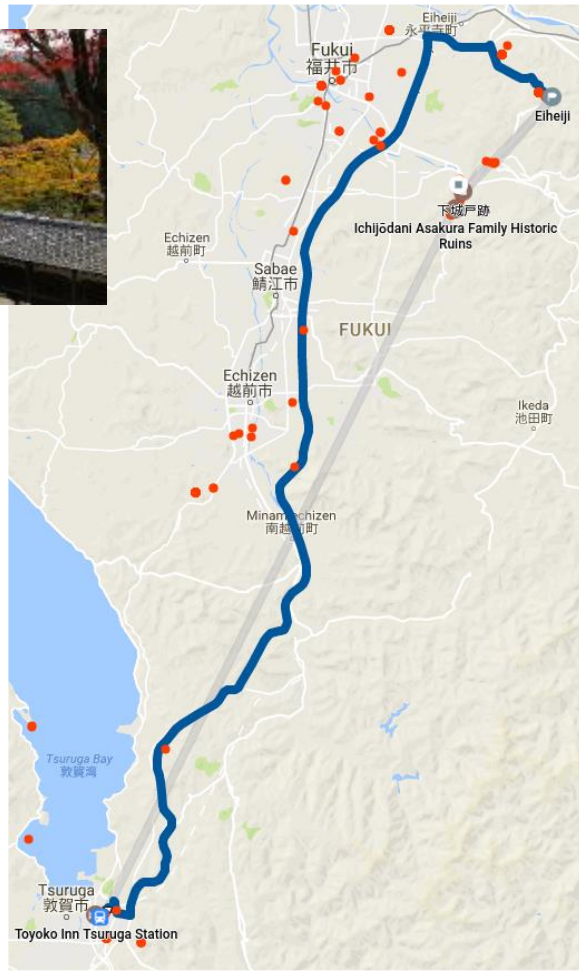
Post-Travel Analysis, or How the Problem Appears, or One Travel Day in Fukui



Google Timeline for one day (automatic, no raw rata)

Images: https://www.japan-guide.com; http://www.turuga.org

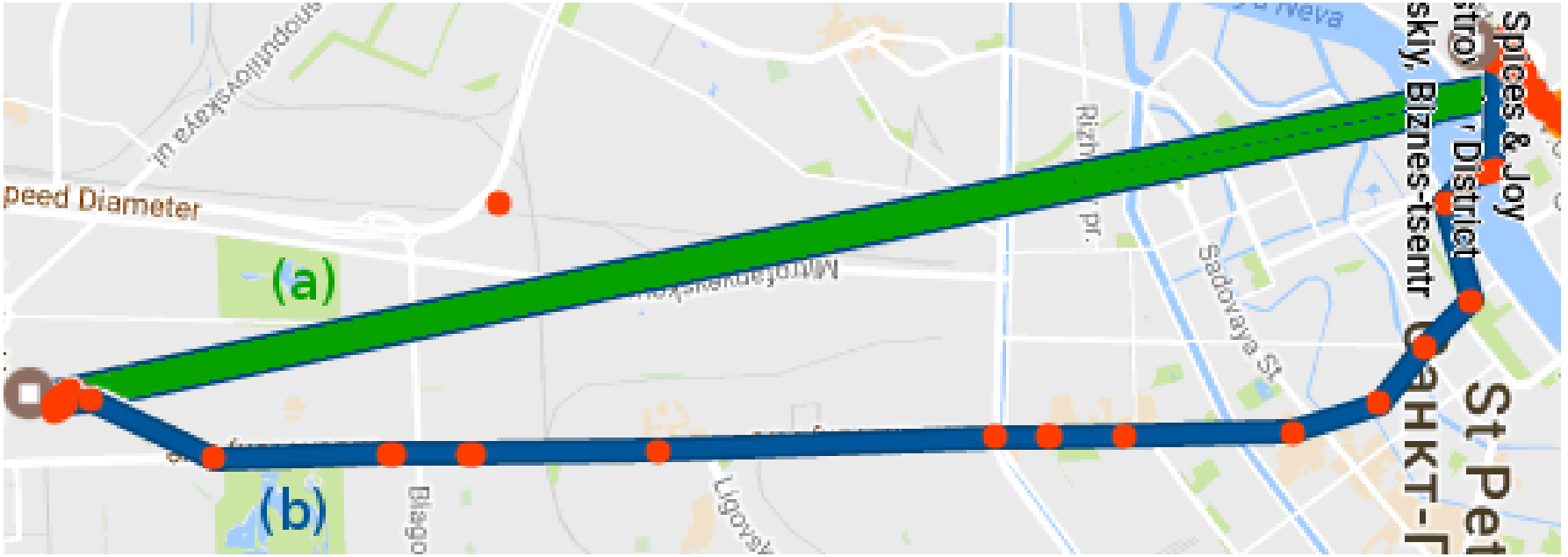
Route source: courtesy of Andrey Kuznetsov



Google Timeline for one day (automatic, raw rata)

Assessment: Primary Experiments

- The companion app does not simply aim at receiving location information, but at making this information available to the Timeline application
- In fact, the companion can even ignore the received location updates without additional processing or storage, because these location events are also handled by the Timeline application



Route source: courtesy of Andrey Kuznetsov

40-minutes city trip (driving):
(a) Standard accuracy of the Google Timeline
(b) Companion application is running

Review of Selected Ongoing Projects

From non-native GUI and image recognition to mobile farms:

- Improving testing quality and advancing methodology of resource and time-consuming testing automation focusing on mobile applications

From signal processing to language learning:

- Intonation acquisition and representation for the benefits of practical phonology

From direct interaction to a companion model:

- Improving location data accuracy for the benefits of depending applications

From engineering to liberal arts:

- Revisiting a case of software engineering education

From Engineering to Liberal Arts: Revisiting a Case of Software Engineering Education

- Considering CS and software disciplines within the context of liberal arts is connected to significant changes in the learning models
  - We anticipate more than only professional developers' skills from our students
    - They have to be able to work in a collaborative environment
    - Significance of organizational learning models favoring **public display, teamwork and professional discussion** significantly increases
- It is extremely important to find ways to create a collaboration environment where students can actively participate in the **co-learning** process together with their more experiences colleagues

Bridge a Methodology Gap in Software Education

Attention to important particularities of software development process with respect to a software development course

Software changeability

- Much different from products of engineering

Software as a community product

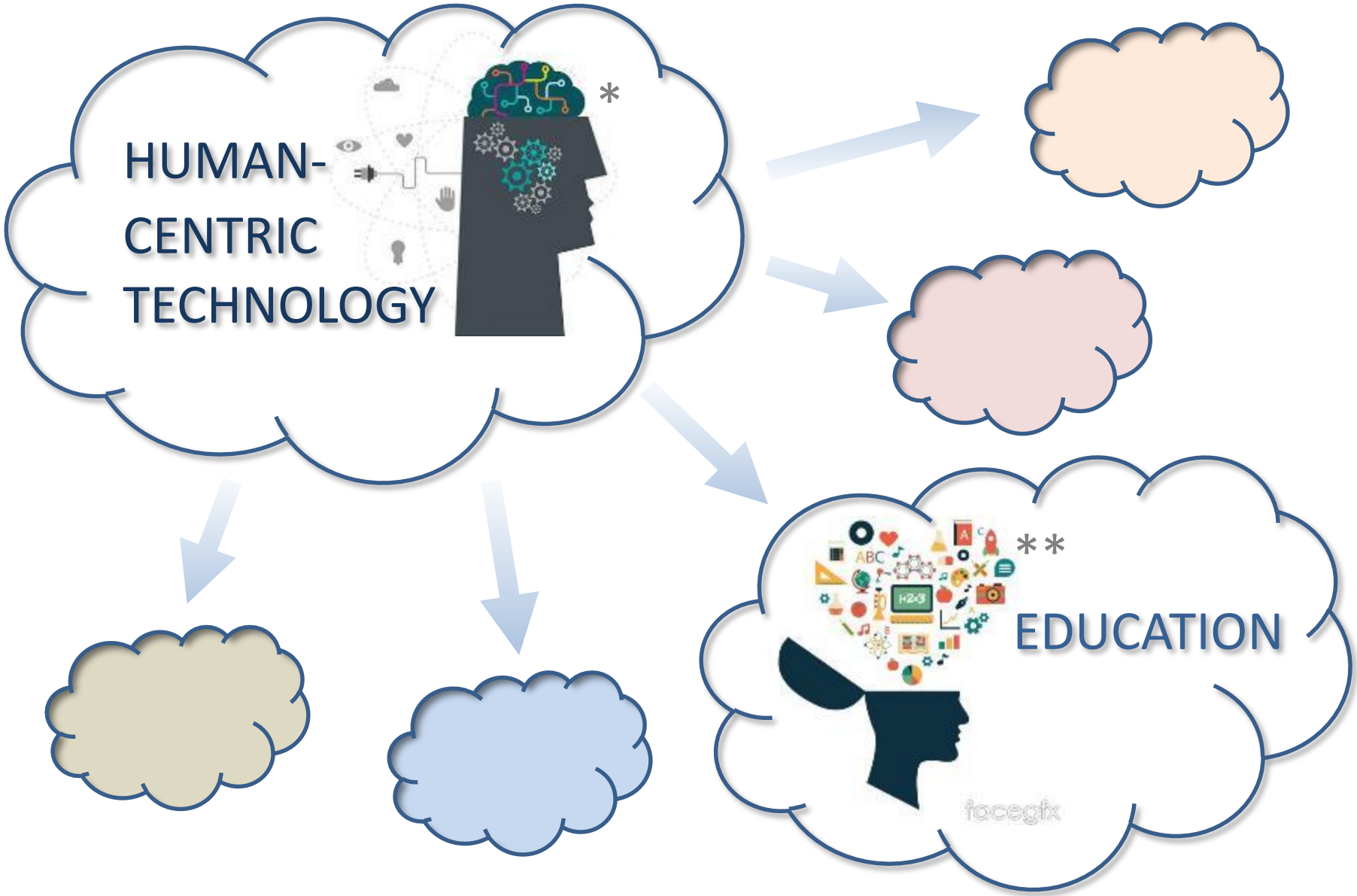
- Contributing to open-source solutions requires specific skills and abilities

Many interdisciplinary activities

- Students have to get programming skills, but also to learn how to communicate with stakeholders, and how to cooperate in multidisciplinary teams

Programming is close to language study

- A software problem may have a variety of acceptable solutions



* http://icc.mtu.edu/hcc/ ** http://gfx9.com

To the Context of Liberal Arts


MARTIANI
MINEI FELICIS
CAPELLAE,
AFRI CARTHAGINIENSIS,
DE
NUPTIIS PHILOLOGIAE ET MERCURII
ET DE SEPTEM
ARTIBUS LIBERALIBUS
LIBRI NOVEM
AD CODICUM MANUSCRIPTORUM FIDEM
ET COMMENTARIO PERPETUO
EDIDIT
ULRICUS FRIDERICUS KOPP,
FRANCOFURTI AD MOENUM MDCCCXXXVI

- Arithmetic
- Geometry
- Astronomy
- Logic
- Grammar
- Rhetoric
- Music

Computer Science

Ultimate of Liberal Arts?

- Computer science draws upon perspectives from many disciplines and has a symbiotic relationship with the liberal arts disciplines, so it might be considered the ultimate of them*

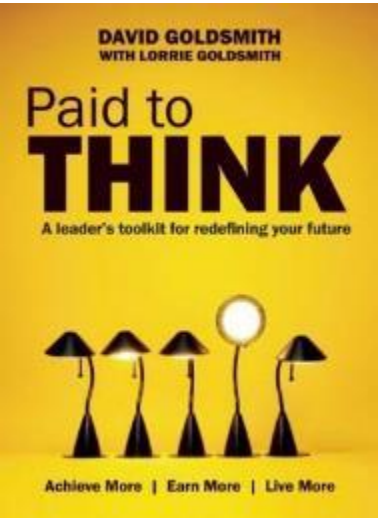
Image: Extracted from Google image search output for "computer science"

* H.M. Walker and C. Kelemen, "Computer science and the liberal arts: a philosophical examination," ACM Transactions on Computing Education (TOCE), Mar 1, 2010, vol. 10, no. 1, pp.2:1–2:10.

Ultimate of Liberal Arts?

as well as perhaps of any kind of good education ☺

- A primary goal of *computer science education* is to develop students' abilities to ***think effectively***,



and it is no less important than developing *specific skills* required for the successful career and professional growth:

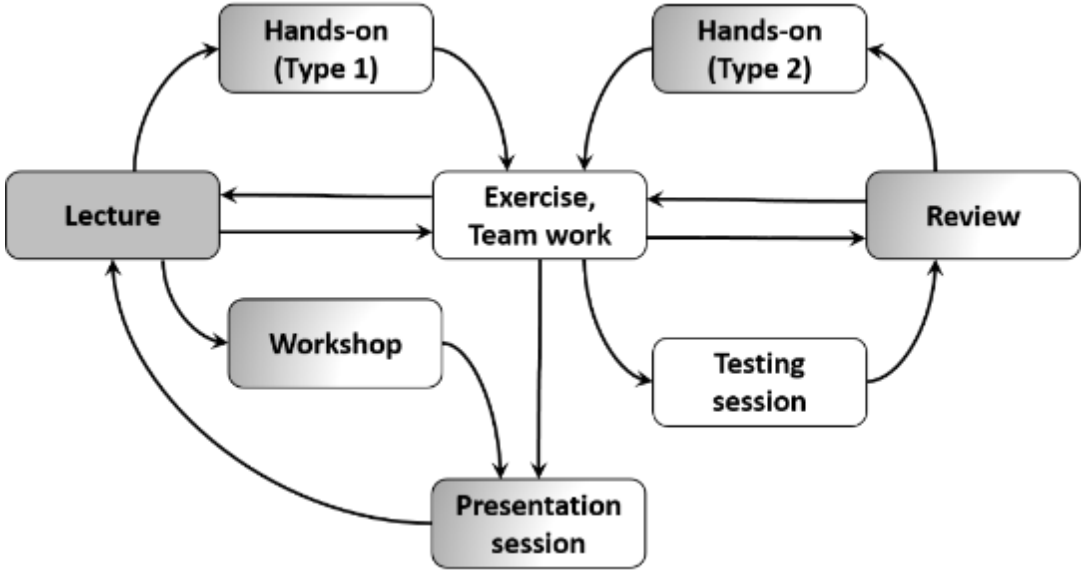
soft skills

*Excellent experts are **"paid to think"****

* David Goldsmith. 2012. Paid to Think: A Leader's Toolkit for Redefining Your Future. BenBella Books, Inc.

Use Case:
Learning Activities in a Programming Course

- Practices and lessons learned after teaching the connected undergraduate courses “Introduction to Programming” and “C Programming” in the University of Aizu (128 class hours in total)



- Diversity of activity forms
  - How computer science can learn from teaching forms and practices which exists in fine arts

AFTERWORD

- How the human-centric computer technology affect the individuals, the society, the technology development in itself
  - Tools focusing on the recognition of human behavior particularities and their habits
  - Co-learning and co-design process
  - Ability to adapt to dramatic changes in expectation of users acting in the digitally transformed world

IARIA
International Academy, Research, and Industry Association
NexTech 2017
Barcelona
November 12 – 16 2017

SOFTWARE THROUGH THE PERSPECTIVE OF HUMAN-CENTRIC COMPUTING PARADIGM

Evgeny Pyshkin

University of Aizu