Future of Computer Engineering

Zehra Çataltepe

October 14, 2014

BLG111E Seminar

Zehra Cataltepe Biography

Ph.D. in Computer Science from Caltech.

Worked at
    Bell Labs Research as a postdoc,
    StreamCenter (a startup in New Jersey) and
    Siemens Corporate Princeton, New Jersey.

Mother of two children.

Currently, Professor at Istanbul Technical University, Computer Engineering Department.

Teach: Machine Learning/Pattern Recognition, Analysis of Algorithms

R&D Projects on/using machine learning:
-feature selection, classifier fusion methods, active learning.
-time series classification (stock market, affective computing, plant condition monitoring)
-mood classification and prediction
-educational data mining
-social network analysis and classification
-recommendation (web, music, movie, document)

For more information, please see:
web.itu.edu.tr/~cataltepe/resume.html

Contents

- A Bright Outlook and Green Occupation Job:
  Software Developers, Systems Software
- Future Technologies
  - Gartner’s Hype Cycle
  - Areas in Turkey?
- Gaps Between Computer Science Education and Profession
- Advice?

Bright Outlook and Green Jobs of the Future

Software Developers, Systems Software Bright Outlook Green Occupation

Related Occupations:

15-1121.00	Computer Systems Analysts Bright Outlook
15-1122.00	Information Security Analysts Bright Outlook
15-1131.00	Computer Programmers Bright Outlook
15-1132.00	Software Developers, Applications Bright Outlook
15-1134.00	Web Developers
15-1141.00	Database Administrators
15-1143.00	Computer Network Architects
15-1199.01	Software Quality Assurance Engineers and Testers Bright Outlook
15-1199.02	Computer Systems Engineers/Architects Bright Outlook
15-1199.05	Geographic Information Systems Technicians Bright Outlook Green Occupation

Robotics Engineers Bright Outlook Green Occupation

From: http://www.onetonline.org

Software Developers, Systems Software

Tasks:

- **Modify** existing software to correct errors, to **adapt** it to new hardware, or to **upgrade** interfaces and **improve performance**.
- Develop or direct software system **testing or validation** procedures.
- Direct software **programming** and development of **documentation**.
- **Consult with customers** or other departments on project status, proposals, or technical issues, such as software system design or maintenance.
- **Analyze information** to determine, recommend, and plan installation of a new system or modification of an existing system.
- Consult with engineering staff to evaluate interface between hardware and software, develop specifications and performance requirements, or resolve customer problems.
- Design or develop software systems, using **scientific analysis and mathematical models** to **predict and measure** outcome and consequences of design.
- Prepare reports or correspondence concerning project specifications, activities, or status.
- **Confer with data processing or project managers** to obtain information on limitations or capabilities for data processing projects.
- **Store, retrieve, and manipulate data** for analysis of system capabilities and requirements.

From: http://www.onetonline.org

Software Developers, Systems Software

Skills

- **Reading Comprehension** — Understanding written sentences and paragraphs in work related documents.
- **Active Listening** — Giving full attention to what other people are saying, taking time to understand the points being made, asking questions as appropriate, and not interrupting at inappropriate times.
- **Critical Thinking** — Using logic and reasoning to identify the strengths and weaknesses of alternative solutions, conclusions or approaches to problems.
- **Complex Problem Solving** — Identifying complex problems and reviewing related information to develop and evaluate options and implement solutions.
- **Speaking** — Talking to others to convey information effectively.
- **Judgment and Decision Making** — Considering the relative costs and benefits of potential actions to choose the most appropriate one.
- **Mathematics** — Using mathematics to solve problems.
- **Social Perceptiveness** — Being aware of others' reactions and understanding why they react as they do.
- **Coordination** — Adjusting actions in relation to others' actions.
- **Monitoring** — Monitoring/Assessing performance of yourself, other individuals, or organizations to make improvements or take corrective action.

From: http://www.onetonline.org

Software Developers, Systems Software

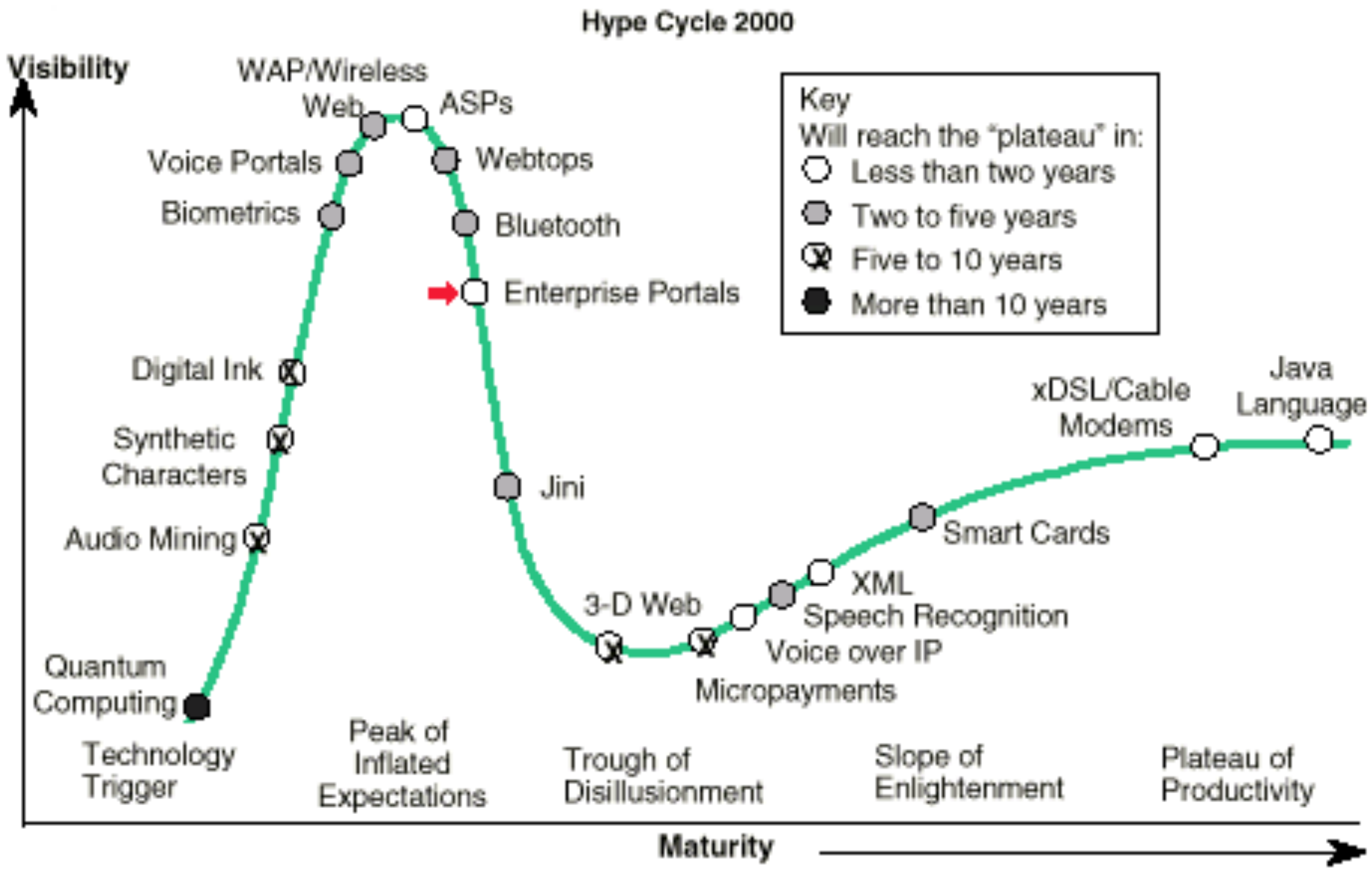
Abilities

- **Oral Comprehension** — The ability to listen to and understand information and ideas presented through spoken words and sentences.
- **Written Comprehension** — The ability to read and understand information and ideas presented in writing.
- **Oral Expression** — The ability to communicate information and ideas in speaking so others will understand.
- **Deductive Reasoning** — The ability to apply general rules to specific problems to produce answers that make sense.
- **Inductive Reasoning** — The ability to combine pieces of information to form general rules or conclusions (includes finding a relationship among seemingly unrelated events).
- **Problem Sensitivity** — The ability to tell when something is wrong or is likely to go wrong. It does not involve solving the problem, only recognizing there is a problem.
- **Written Expression** — The ability to communicate information and ideas in writing so others will understand.
- **Information Ordering** — The ability to arrange things or actions in a certain order or pattern according to a specific rule or set of rules (e.g., patterns of numbers, letters, words, pictures, mathematical operations).
- **Near Vision** — The ability to see details at close range (within a few feet of the observer).
- **Speech Recognition** — The ability to identify and understand the speech of another person.

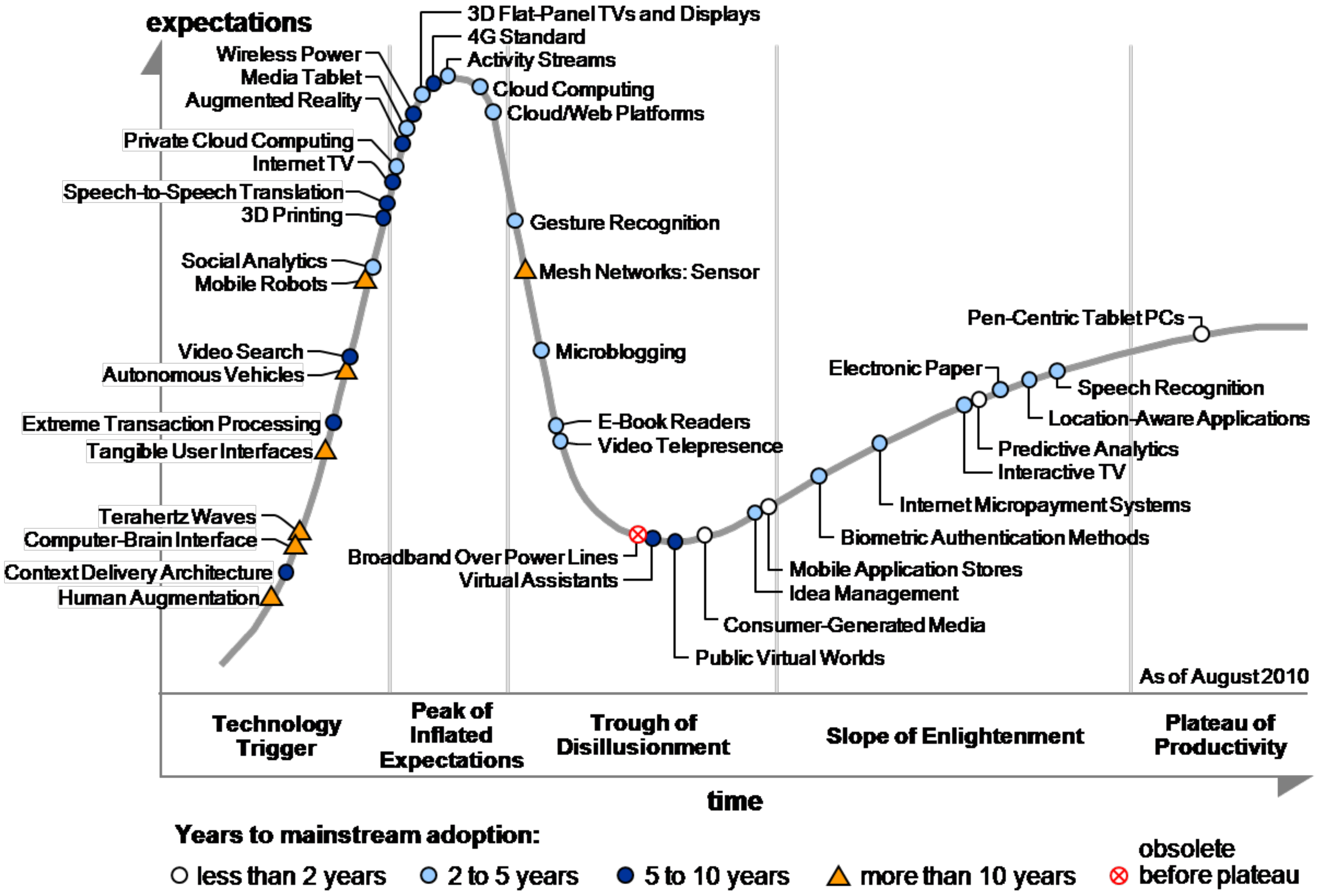
From: http://www.onetonline.org

Gartner's Hype Cycle

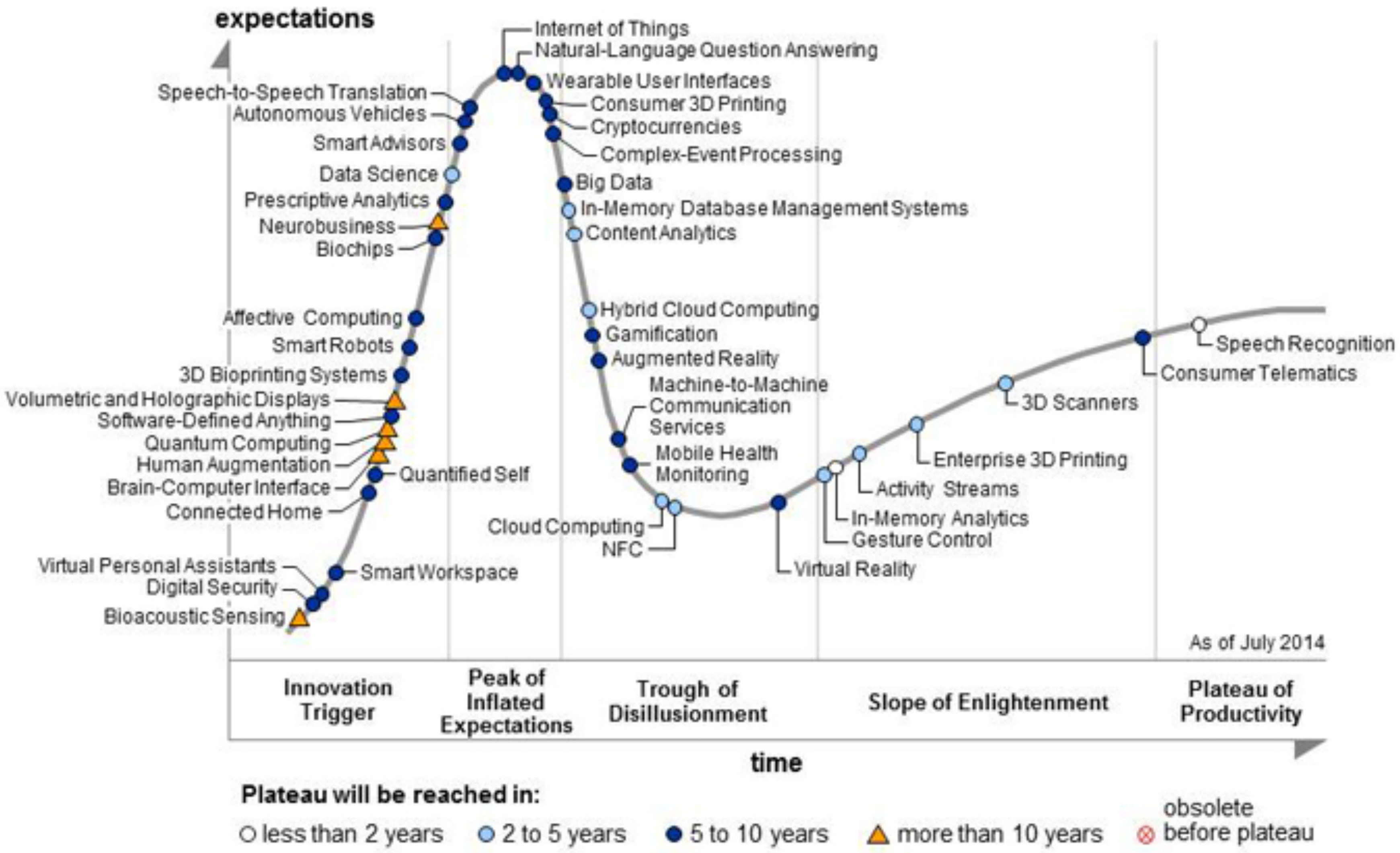
Interpreting Technology Hype
"When new technologies make bold promises, how do you discern the hype from what's commercially viable? And when will such claims pay off, if at all? Gartner Hype Cycles provide a graphic representation of the maturity and adoption of technologies and applications, and how they are potentially relevant to solving real business problems and exploiting new opportunities. Gartner Hype Cycle methodology gives you a view of how a technology or application will evolve over time, providing a sound source of insight to manage its deployment within the context of your specific business goals."

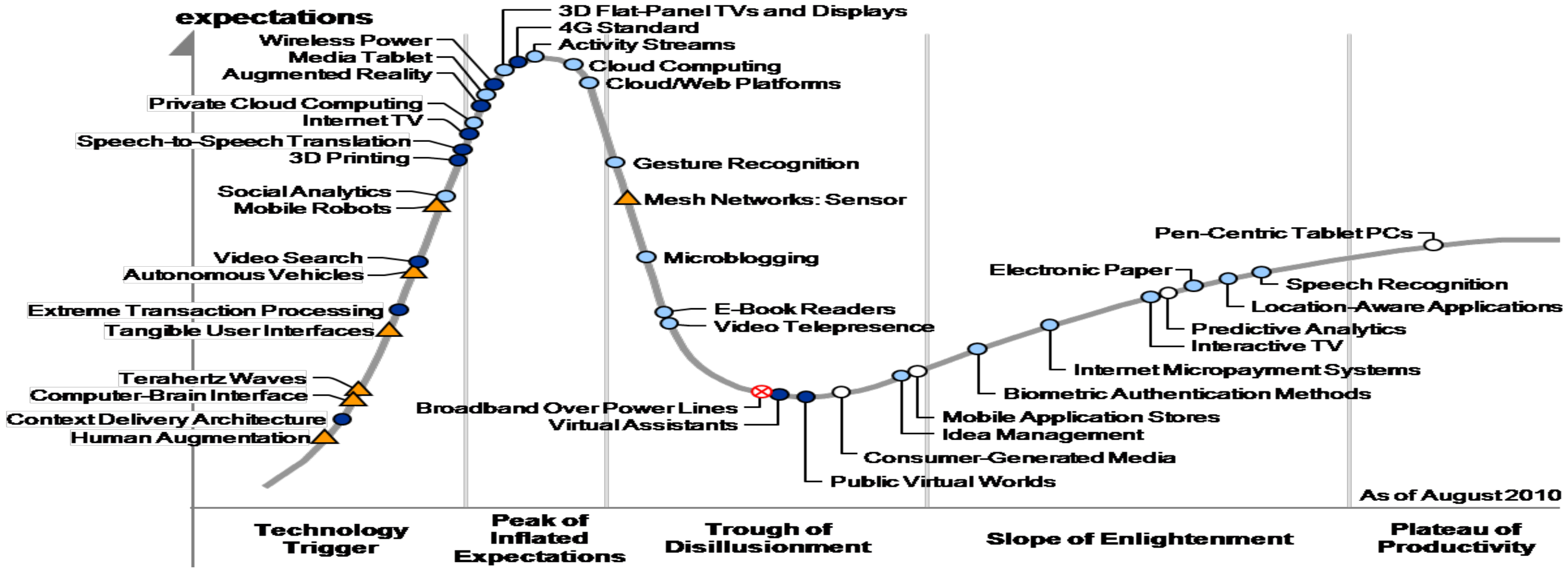


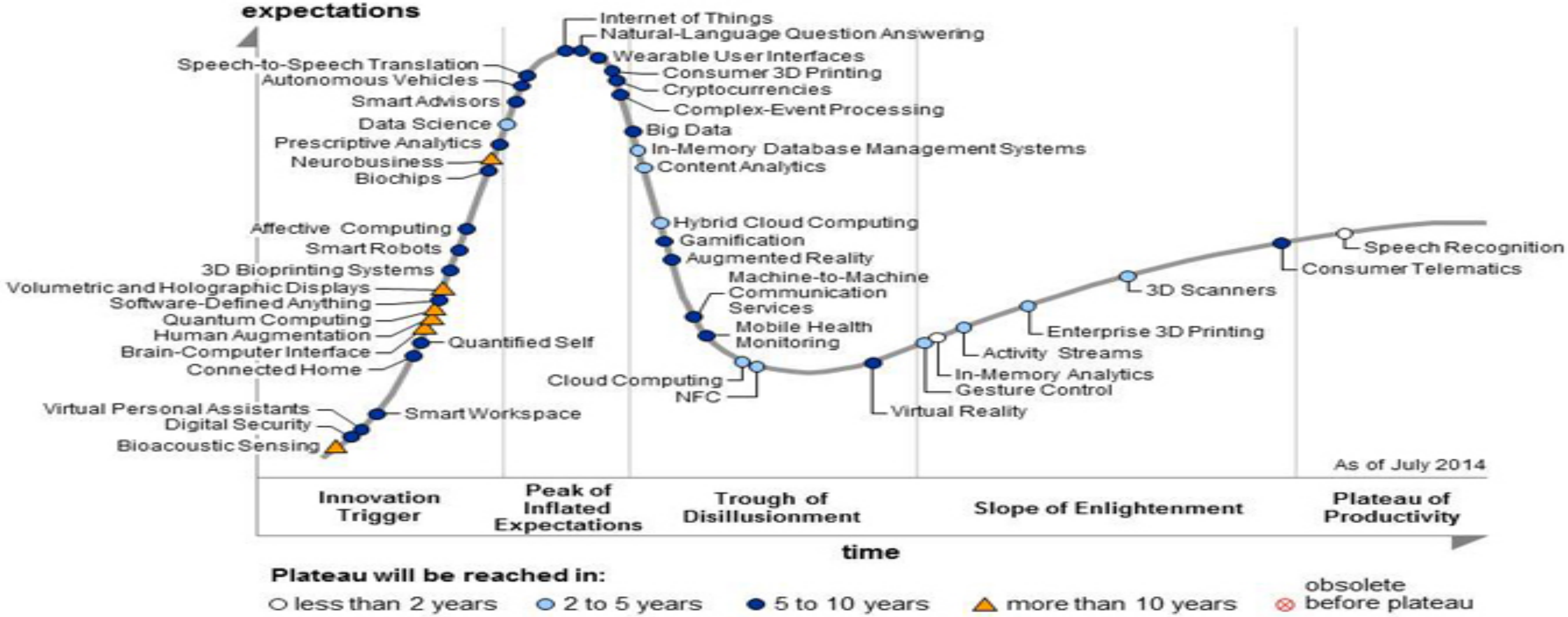
Gartner's Hype Cycle



Gartner's Hype Cycle




expectations
3D Flat-Panel TVs and Displays
4G Standard
Activity Streams
Cloud Computing
Cloud/Web Platforms
Wireless Power
Media Tablet
Augmented Reality
Private Cloud Computing
Internet TV
Speech-to-Speech Translation
3D Printing
Gesture Recognition
Mesh Networks: Sensor
Social Analytics
Mobile Robots
Microblogging
Video Search
Autonomous Vehicles
E-Book Readers
Video Telepresence
Extreme Transaction Processing
Tangible User Interfaces
Pen-Centric Tablet PCs
Electronic Paper
Speech Recognition
Location-Aware Applications
Predictive Analytics
Interactive TV
Internet Micropayment Systems
Biometric Authentication Methods
Terahertz Waves
Computer-Brain Interface
Context Delivery Architecture
Human Augmentation
Broadband Over Power Lines
Virtual Assistants
Mobile Application Stores
Idea Management
Consumer-Generated Media
Public Virtual Worlds
As of August 2010
Technology Trigger
Peak of Inflated Expectations
Trough of Disillusionment
Slope of Enlightenment
Plateau of Productivity


expectations
Internet of Things
Natural-Language Question Answering
Wearable User Interfaces
Consumer 3D Printing
Cryptocurrencies
Complex-Event Processing
Speech-to-Speech Translation
Autonomous Vehicles
Smart Advisors
Data Science
Prescriptive Analytics
Neurobusiness
Biochips
Big Data
In-Memory Database Management Systems
Content Analytics
Affective Computing
Smart Robots
3D Bioprinting Systems
Volumetric and Holographic Displays
Software-Defined Anything
Quantum Computing
Human Augmentation
Brain-Computer Interface
Connected Home
Quantified Self
Hybrid Cloud Computing
Gamification
Augmented Reality
Machine-to-Machine Communication Services
Mobile Health Monitoring
Cloud Computing
NFC
Virtual Personal Assistants
Digital Security
Bioacoustic Sensing
Smart Workspace
Speech Recognition
Consumer Telematics
3D Scanners
Enterprise 3D Printing
Activity Streams
In-Memory Analytics
Gesture Control
Virtual Reality
As of July 2014
Innovation Trigger
Peak of Inflated Expectations
Trough of Disillusionment
Slope of Enlightenment
Plateau of Productivity
time
Plateau will be reached in:
○ less than 2 years
○ 2 to 5 years
● 5 to 10 years
△ more than 10 years
⊗ obsolete before plateau

Hype Cycle for **Advanced Analytics and Data Science**, 2014

On the Rise

Advanced Anomaly Detection

Uplift Modeling

Spark

Crowdsourcing of Microwork

Data Lakes

Decisions and Recommendations as a Service

Hadoop-Based Data Discovery

Smart Pattern Discovery

Graph Analysis

Climate-Driven Forecasting

Optimization

Deep Learning

Machine Learning

Emotion Detection/Recognition

PMML

Prescriptive Analytics

At the Peak

Data Science

Decision Management

Geospatial and Location Intelligence

Predictive Analytics

Autonomous Vehicles

Natural-Language Question Answering

R

Complex-Event Processing

Climbing the Slope

In-Memory Analytics

Simulation

Sliding Into the Trough

Big Data

Model Managers

Audio Mining/Speech Analytics

Facial Recognition

Video Analytics

Ensemble Learning

Linked Data

Case-Based Reasoning

Prediction Markets

Text Analytics

Hype Cycle for **Big Data**, 2014

On the Rise

Context Brokers

Virtual Personal Assistants

Personal Analytics

Information Valuation and Infonomics

Data as a Service

Operational Intelligence Platforms

Supply Chain Big Data Analytics

Quantified Self

Data Lakes

Hadoop-Based Data Discovery

Graph Analysis

Open Data

Search-Based Data Discovery Tools

Graph Database Management Services

MDM and Social Data

Prescriptive Analytics

At the Peak

Big Data Analytics for Customer Intelligence

Data Science

High-Performance Message Infrastructure

Predictive Analytics

Smart Advisors

Information Capabilities Framework

Big Data Analytics for Fraud and Security

Information Semantic Services

Table-Style Database Management Services

Enterprise Information Architecture

Hadoop SQL Interfaces

Internet of Things

Logical Data Warehouse

Natural-Language Question Answering

Wearable User Interfaces

Document Store Database Management Systems

Complex-Event Processing

Sliding Into the Trough

Key-Value Database Management Systems

In-Memory Database Management Systems

Content Analytics

Dynamic Data Masking

Telematics

Hadoop Distributions

Entity Resolution and Analysis

In-Memory Data Grids

Data Warehouse Platform as a Service (dwPaaS)

Social Analytics

Cloud Computing

Linked Data

Climbing the Slope

In-Memory Analytics

Data Federation/Virtualization

Hype Cycle for **Internet of Things**, 2014

On the Rise

Digital Security
Smart Dust
Licensing and Entitlement Management
Smart Workspace
Ambient Energy Harvesters for the Internet of Things
Connected Home
IoT Platform
Operational Intelligence Platforms
Quantified Self
802.11ah
Embedded Software and Systems Security
Internet of Things in Insurance
Decisions and Recommendations as a Service
Weightless
Message Queue Telemetry Transport
Machine Learning

At the Peak

High-Performance Message Infrastructure
Smart City Framework
Autonomous Vehicles
Internet of Things for Government
IT/OT Integration
Context Delivery Architecture
iBeacons and Bluetooth Beacons
Intelligent Business Operations
Sensor Fusion
Smart Transportation
Home Energy Management/Consumer Energy Management
Internet of Things
Low-Cost Development Boards
Consumer Smart Appliances
Internet of Things for Manufacturing Operations
Wearable User Interfaces
Wireless Power Charging
Bluetooth 4
Complex-Event Processing

Sliding Into the Trough

Big Data
Facilities Energy Management
Operational Technology Security
Asset Performance Management
Advanced Metering Infrastructure
Machine-to-Machine Communication Services
Mobile Health Monitoring
Operational Technology Platform Convergence
Vehicle-to-Infrastructure Communications
Enterprise Manufacturing Intelligence
AMQP

Climbing the Slope

RFID for Logistics and Transportation
ZigBee
Consumer Telematics

Entering the Plateau

Wireless Healthcare Asset Management

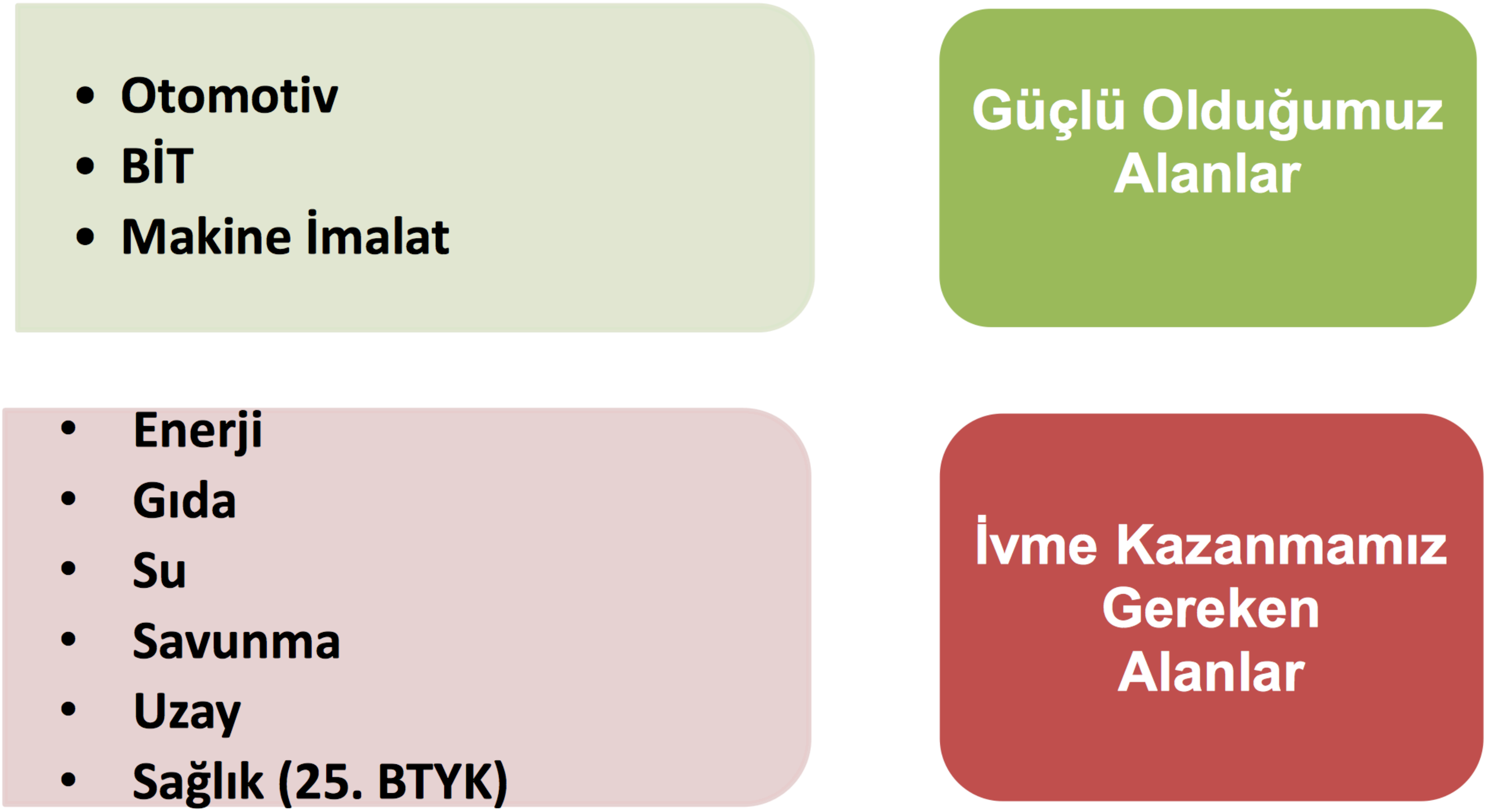
Large Data Sets

“Future computer science research is believed to employ, analyze, and interpret large data sets…. As computing pervades every facet of our lives and data collection becomes increasingly ubiquitous, feasible algorithms for solving these problems are becoming more and more necessary to analyze and understand the vast quantities of available information. In order to rigorously develop these algorithms, a mathematical foundation must be established for large data sets, including the theory of large graphs, high-dimensional data, sparse vectors and so on. These innovative studies discover striking results that reveal a fundamental change in computer science that will reshape our knowledge of the world.”

http://www.cs.cornell.edu/~lwang/Wang11IJSI.pdf
Hopcroft, J. E., Soundarajan, S., & Wang, L. (2011). The future of computer science. *International Journal of Software and Informatics*, *5*(4), 549-565.

Tubitak Teydeb Öncelikli Alanlar

Ulusal Bilim Teknoloji Yenilik Stratejisi 2011-2016' belgesinde ülkemiz öncelikli alanları tanımlanmıştır.



- **Otomotiv**
- **BİT**
- **Makine İmalat**

Güçlü Olduğumuz Alanlar

- **Enerji**
- **Gıda**
- **Su**
- **Savunma**
- **Uzay**
- **Sağlık (25. BTYK)**

İvme Kazanmamız Gereken Alanlar

http://www.tubitak.gov.tr/sites/default/files/1511-programtanitimi.pdf

Gaps Between Computer Science Education and Profession

- There will be a gap between what you have learned at the university and what you will need at the workplace…..

What needs to be included in the curriculum

“Research into the computer science curriculum has suggested **various phases of software engineering** are needed to improve students’ knowledgebase when entering the workforce.

...extreme programming, testing methodology, and communication skills.

Patterson [7] **open source software** into the curriculum, as it provides graduates with the ability to help

improve potential organizations. ….Eclipse…..

Douglas et al. [1] **human computer interaction** into the computer science curriculum, as 50% of code written for software applications is designed for the user interface.

Fitzpatrick [2] introduced the Software Quality Star, a model for improving software quality motivated by ISO/IEC 12207, which attempts to integrate human-computer interaction (HCI) and software engineering to improve curriculum regarding customer usability.

Havill and Ludwig [4] suggested three approaches to improving CS undergraduate programs, which include **improving communication skills**, **exposing students to research early**, and **increasing exposure to mathematics**. This is a valid point for computer scientists moving towards a graduate level degree, but lacks the increasing need of how to effectively work with customers and understand what is being requested.

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

"What positions does your company usually fill with computer science graduates (not MIS or business information systems majors)?

There is a consensus from the interviews that a wide range of positions are available for computer scientists. Positions consisted of programmer analyst, software engineer, test engineer, network admin, telecommunication, application development, quality engineer, systems engineer, requirements engineer, project engineer, operation support, help desk, technical advisor, software architect, technical analyst, and business analyst."

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

"How important is it for computer science undergraduates to know how to gather and elicit customer requirements to fulfill IT positions in your company?

From the survey we find that gathering and eliciting customer requirements is very desirable skill set for computer science graduates. 80% of professionals surveyed stated the importance was high, where 20% stated the importance was medium to low. Those in the minority state a senior level person performs this job, but this further solidifies the need for computer science curriculum to include requirement elicitation techniques to create a well-rounded graduate when entering corporate America."

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

“What do computer science undergraduates need to have in their undergraduate programs in terms of soft/ behavioral skill sets?

Participants suggested computer science curricula need to include training on effective writing skills for documentation and status reporting and communication and presentation skills. Team building, the ability to be flexible and how to deal with hostile personalities were highly emphasized. Negotiation skills, software requirement gathering, project management, and information management skills were also mentioned as important soft skill sets.”

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

"What do computer science undergraduates need to have in their undergraduate program in terms of technical skill sets (e.g. programming languages)?

Our interviewees discussed the importance of object oriented programming, decoupling understanding, analytical skills, and integrity in programming. Specific languages such as Java, JavaScript, Ruby, Perl, HTML, CSS, SQL, Python, Junit, .Net, C++, and UML were identified. Other concepts such as web services, network administration, server administration, database administration, and requirement engineering were stated."

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

"What do computer science undergraduates need to have in their undergraduate program to help them when they are required to work on project teams (e.g. process improvement teams or new product development teams)?

Interview participants recommend the need for CS students to be familiar with process improvement frameworks (e.g. 6 Sigma and CMMI). Significant group project related skills, good communication, time forecasting, and how to ask the right questions were some of the most prominent comments for this questions. The systems development life cycle was described as a fundamental piece of CS curriculum but Microsoft Project, RUP, and PMBOK, placed in the curriculum could enable success when entering the workforce in any position."

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Gaps in the computer science curriculum:

an exploratory study of industry professionals

"In general, what do computer science undergraduates need to know, to enable them to have a successful career in an IT department?

Participants mentioned that CS graduates need to know how their work affects the bottom line. Doing so could help them realize that their work matters and their best effort should always be put forth. Some characteristics mentioned include being flexible, being able to meet short and long term goals, communicating with upper management, having integrity and a great attitude. A wide knowledge of systems and technology integration, certificates on the latest technology, open source technologies, as well as being a good researcher, and obtaining a good mentor would lead to success in today's IT department."

Simmons, C. B., & Simmons, L. L. (2010). Gaps in the computer science curriculum: an exploratory study of industry professionals. *Journal of Computing Sciences in Colleges*, *25*(5), 60-65.
http://www.researchgate.net/publication/234805197_Experiences_with_active_learning_in_CS_3/file/60b7d528ba0fbf38d6.pdf#page=73

Educational and on the Job Experiences

"Discrepancies were found particularly in areas related to practical skills (such **as testing, maintaining code over time, use of source code control and development tools**), **communication, critical thinking and problem solving**, and **strategies used to continue learning on-the-job**. Participant suggestions for improving university programs focused largely on the **use of large scale, complex, authentic projects of significant duration**. The author recommends further consideration be given to **explicitly teaching the type of self-learning skills and strategies** used by experienced professionals."

Exter, M. (2014, March). Comparing educational experiences and on-the-job needs of educational software designers. In *Proceedings of the 45th ACM technical symposium on Computer science education* (pp. 355-360). ACM.

See also:

Radermacher, A., & Walia, G. (2013, March). Gaps between industry expectations and the abilities of graduates. In *Proceeding of the 44th ACM technical symposium on Computer science education* (pp. 525-530). ACM.

Suhaimi, A., & Sukiman, S. A. SKILLS NEEDED BY IT GRADUATES AS PERCEIVED BY MALAYSIAN IT PROFESSIONALS.

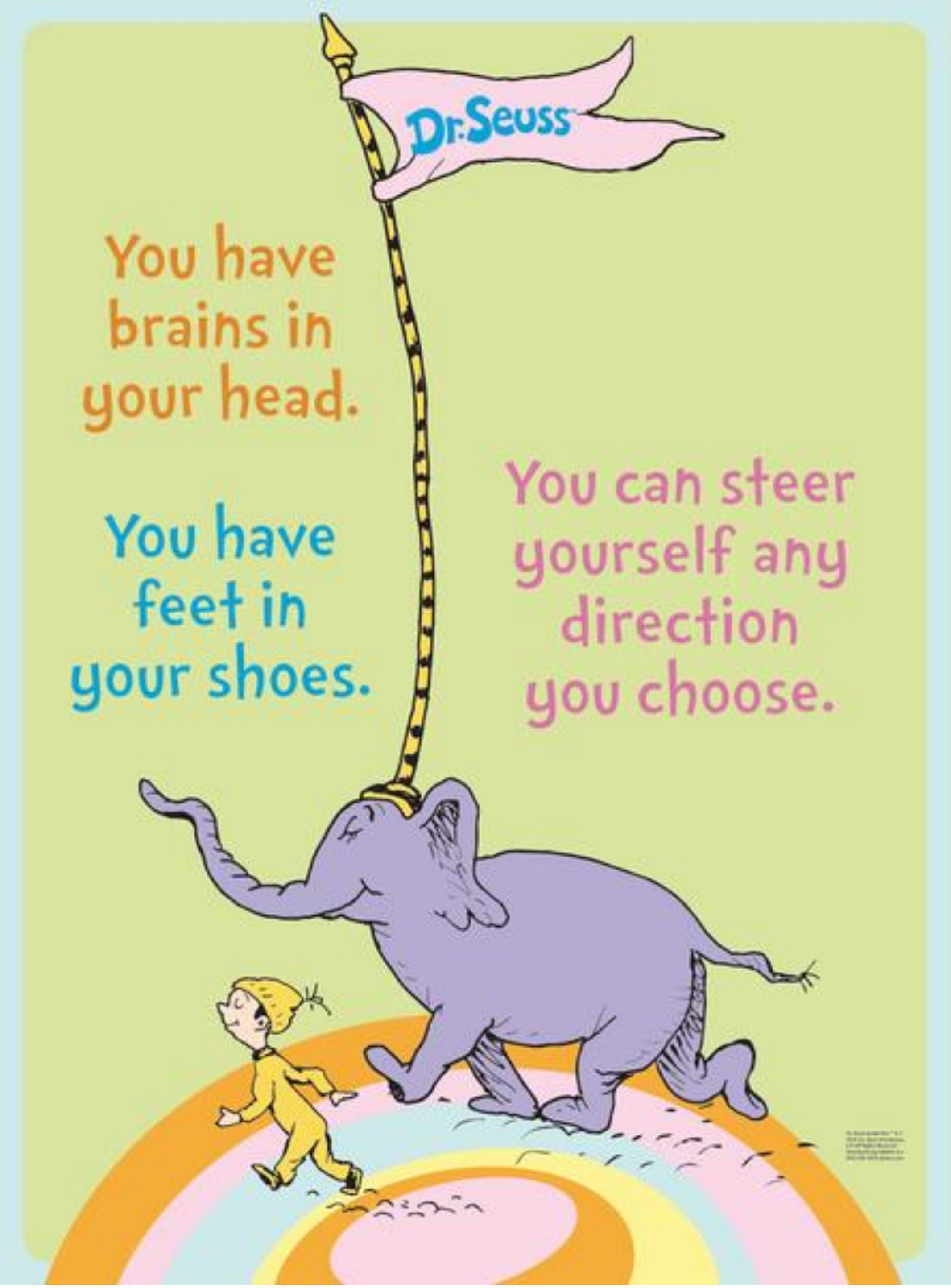
Advice?

- Take good care of yourself and your family
- Have fun
- Know how to communicate (writing, speaking)
- Think analytically, use math
- Have friends and a professional network
- Always learn and follow the new trends, things change very quickly
- Don’t give up
- Know when to make changes

Good luck!

Questions?

cataltepe@itu.edu.tr


Dr. Seuss
You have brains in your head.
You have feet in your shoes.
You can steer yourself any direction you choose.