声明：

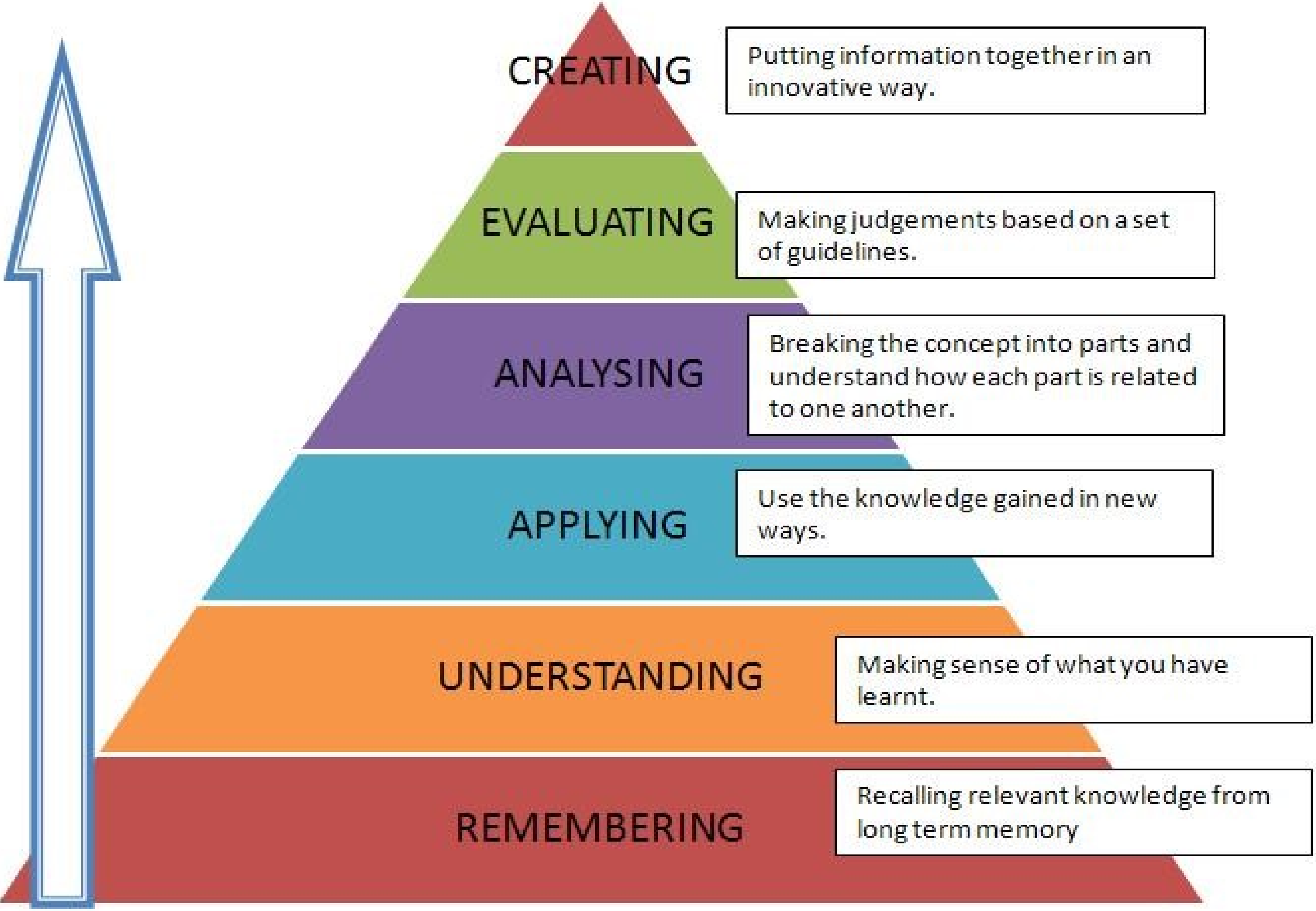
1.本课程所用PPT和作业内容源于edx：BerkeleyX:
Engineering Software as a Service。

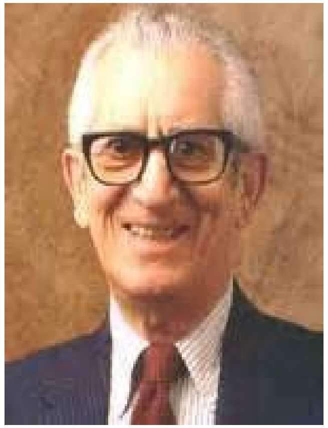
2.本课程所用PPT和作业内容经过了部分删除和修改。

3.本课程所用代码管理平台和实验部署平台涉及部分互联网开源内容。

Welcome to Advanced Software Engineering

“The purpose of education is to change the thoughts, feelings and actions of students”


CREATING
Putting information together in an innovative way.
EVALUATING
Making judgements based on a set of guidelines.
ANALYSING
Breaking the concept into parts and understand how each part is related to one another.
APPLYING
Use the knowledge gained in new ways.
UNDERSTANDING
Making sense of what you have learnt.
REMEMBERING
Recalling relevant knowledge from long term memory



Bloom, Benjamin S

What is Computing ?

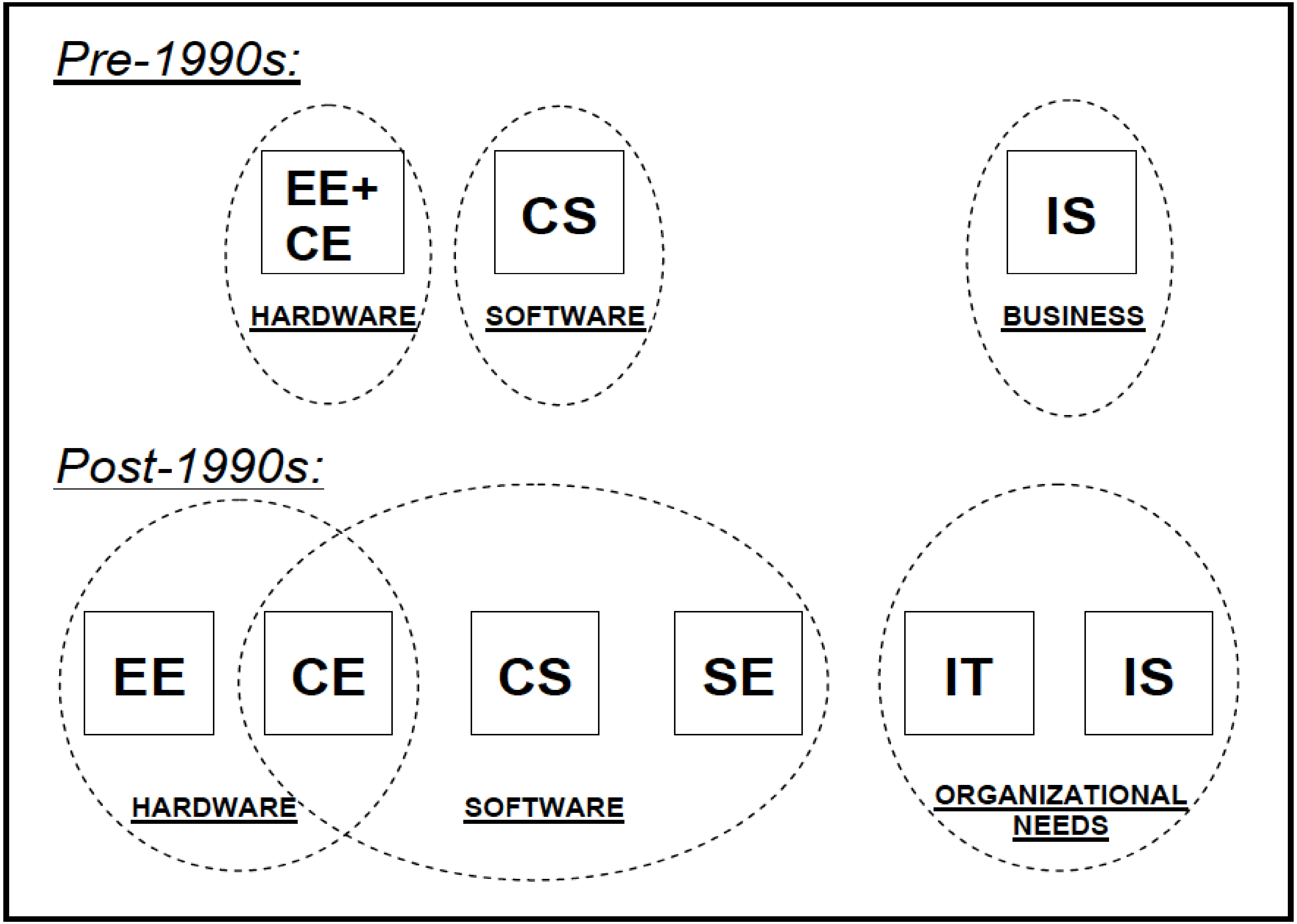
1. Designing and building hardware and software systems for a wide range of purposes;
2. Processing, structuring, and managing various kinds of information;
3. Doing scientific studies using computers;
4. Making computer systems behave intelligently;
5. Creating and using communications and entertainment media;
6. Finding and gathering information relevant to any particular purpose, and so on.

Post- 1990s

- EE / CE address computer hardware and related software issues involved in the design of digital device.
- CS is one of the family of academic disciplines; focus on creating new knowledge. Software is the currency in which ideas are expressed and a wide range of computing problems and applications are explored.
- SE address the important challenges inherent in building software systems that are reliable and affordable. Creating software that satisfies robust real-world requirements;
- IS: The problems of managing information became extremely complex, and the challenges of making proper use of IT to support org. efficiency and effectiveness became crucial issues. Focus on the generation and use of information.

EE- Electronic Engineering / CE-Computer Engineering / CS- Computer Science
SE- Software Engineering / IS – Information System

Big Picture of Computing Curricula



Computer Science

Three Major Tasks:

1. Design and implement software.

- Programming job, supervise programmers, keeping them aware of new approaches

2. Devise new way to use computers.

- Net, DB, HCI → WWW; make robots became practical and intelligent aides. DB create new Knowledge, decipher the secrets of our DNA

3. Develop effective ways to solve computing problems.

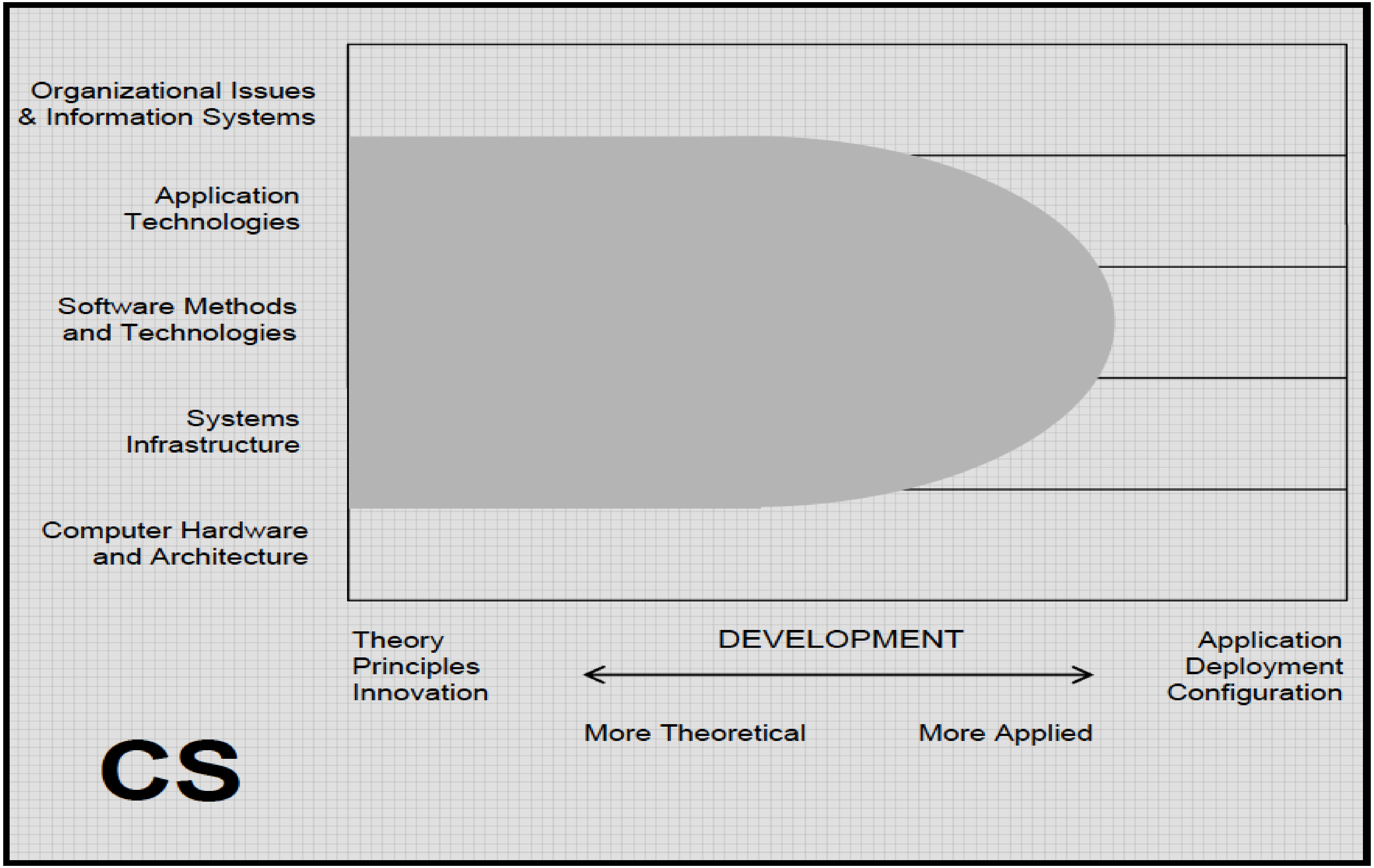
- develop the best possible ways to store information in databases, send data over networks and display complex images. to determine the best performance possible, and their study of algorithm helps them to develop new approaches.

Software Engineering

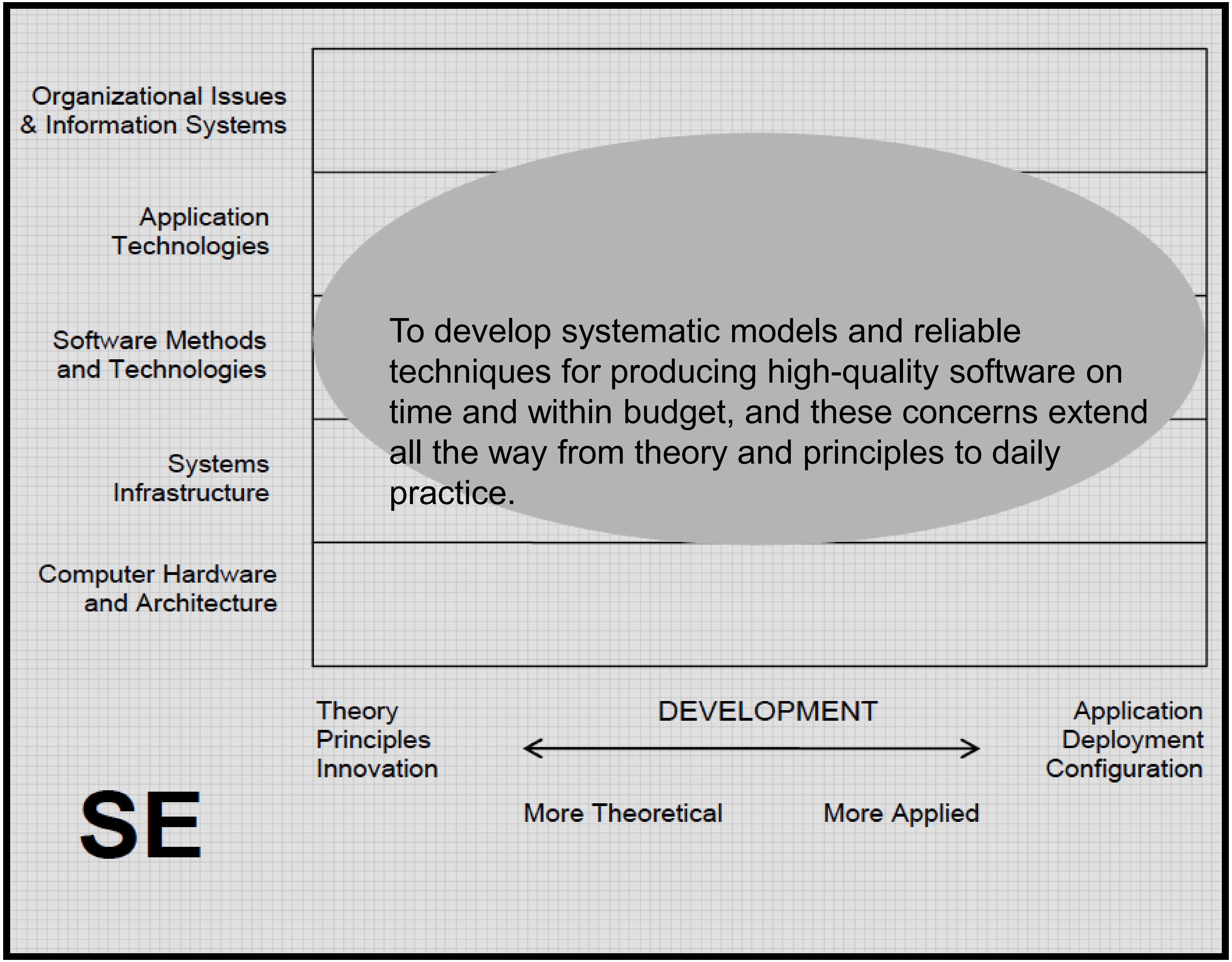
SE is the discipline of developing and maintaining software systems that behave reliably and efficiently, are affordable to develop and maintain, and satisfy all the requirements that customers have defined for them.

1. SE is different from other engineering disciplines due to:
   - Intangible nature of software
   - Discontinuous nature of software operation
2. SE students learn how to assess customer needs and develop usable software that meets those needs. Knowing how to provide genuinely useful and usable software is of paramount important.
3. Don’t confuse the discipline of SE with the ambiguous use of the term software engineer as used in employment advertisements and job titles

Computer Science Focus



Software Engineering Focus



Organizational Issues & Information Systems

Application Technologies

Software Methods and Technologies

Systems Infrastructure

Computer Hardware and Architecture

To develop systematic models and reliable techniques for producing high-quality software on time and within budget, and these concerns extend all the way from theory and principles to daily practice.

Theory
Principles
Innovation

DEVELOPMENT

Application
Deployment
Configuration

More Theoretical

More Applied

SE

Hard Problems in SE

- A software developers’ goal is to provide suitable solutions for users that consist of quality code and that don’t cost too much. Unfortunately, they usually don’t succeed. This problem has three facets:

（1）A Suitable Solution for Users means a software feature set that is both useful enough and usable enough for solving the user’s need at the time of delivery and afterwards.

（2）Quality Code is code that not only functions correctly, but also can be maintained and extended, as necessary.

（3）Cost has both money and time components, and must be both predictable and controllable. Tension exists between these three facets because whenever you try to improve one facet, it may adversely affect the other facets.

Course Goal

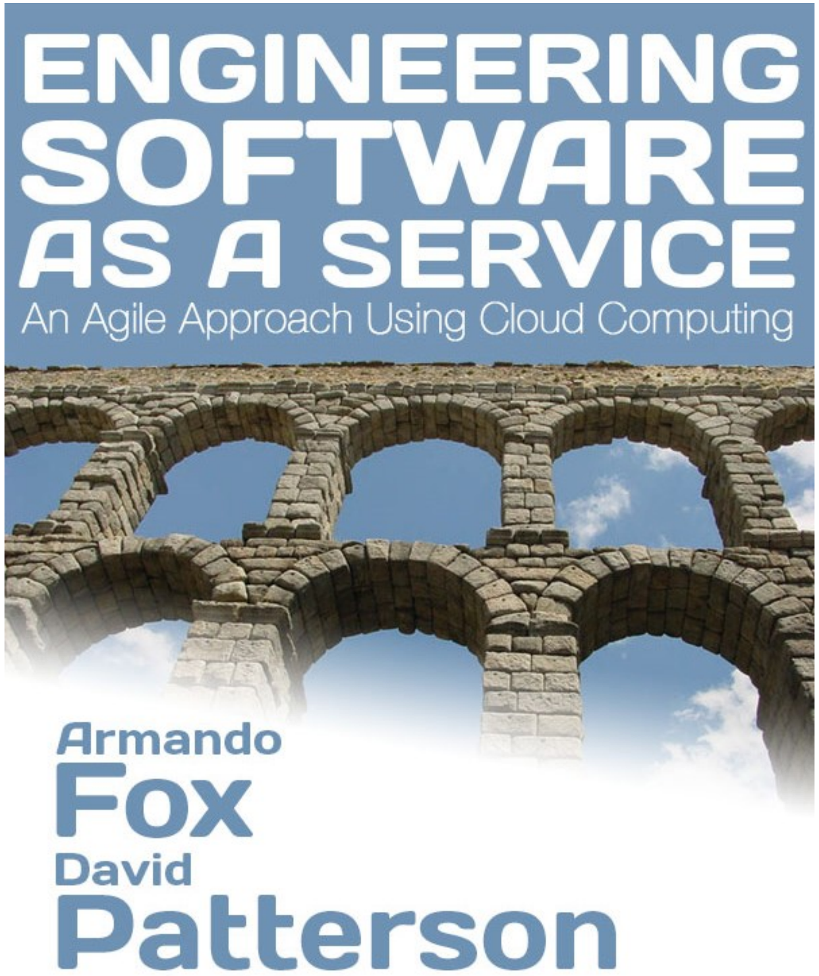
1. Develop the ability to distinguish between good and bad Internet service ideas;
2. Develop the ability to deal with the complexity in designing and implementing applications;
3. Learn how to create a good software models for a specific application requirement, especially for SaaS on Cloud;
4. Learn how to make a assessment for a software system, especially for SaaS on Cloud;
5. Learn team-working for conceiving and designing a useful application.

Course Organization

- 课堂讲授：9月14日（周四）开始至1月18日，共18次课（去除国庆节、元旦和考试），课堂提问和讨论，课堂参与考核参考。
- 实践环节
  - 提供候选项目列表，供选择（语言Ruby, Rails 框架）
  - 使用PaaS平台（自建或者选用Heroku）部署项目的SaaS
  - 代码托管平台
    - www.github.com / www.trustie.net
  - BDD：用例工具 Cucumber（如何用一组清晰、可执行且团队中任何人都能读懂的规格说明来表达用户的想法）
  - TDD：测试用例工具Rspec,（描述我们对系统执行某个样例(example)的期望行为(behavior) ）
  - 项目管理：www.github.com
- 考核要求：
  - 考试 40% （课堂闭卷）时间：估计1月25日
  - 作业+平时 20% （作业每周一次，平时抽查）
  - 项目 40% （内容：开题时间-估计10月12日、演示和报告检查时间-估计1月25日）

Textbook

- http://saasbook.info
- ***First Edition*** (≥ 1.1.0)

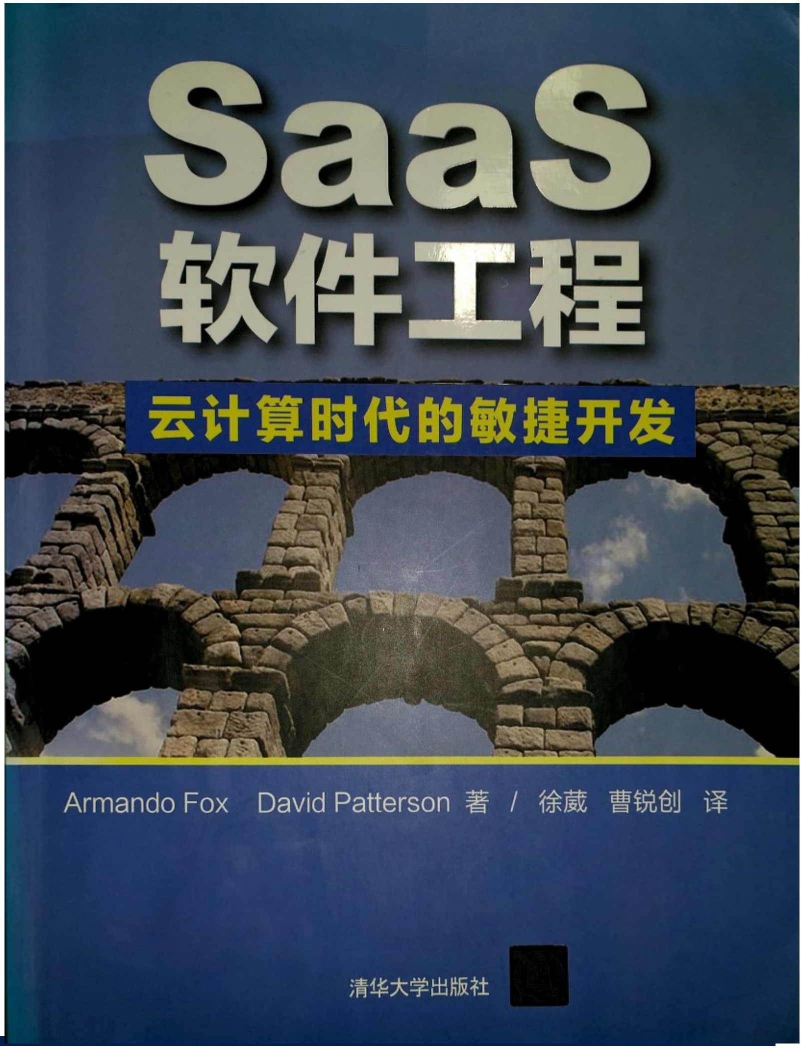


ENGINEERING SOFTWARE AS A SERVICE

An Agile Approach Using Cloud Computing

Armando Fox

David Patterson



SaaS 软件工程

云计算时代的敏捷开发

Armando Fox David Patterson 著 / 徐葳 曹锐创 译

清华大学出版社

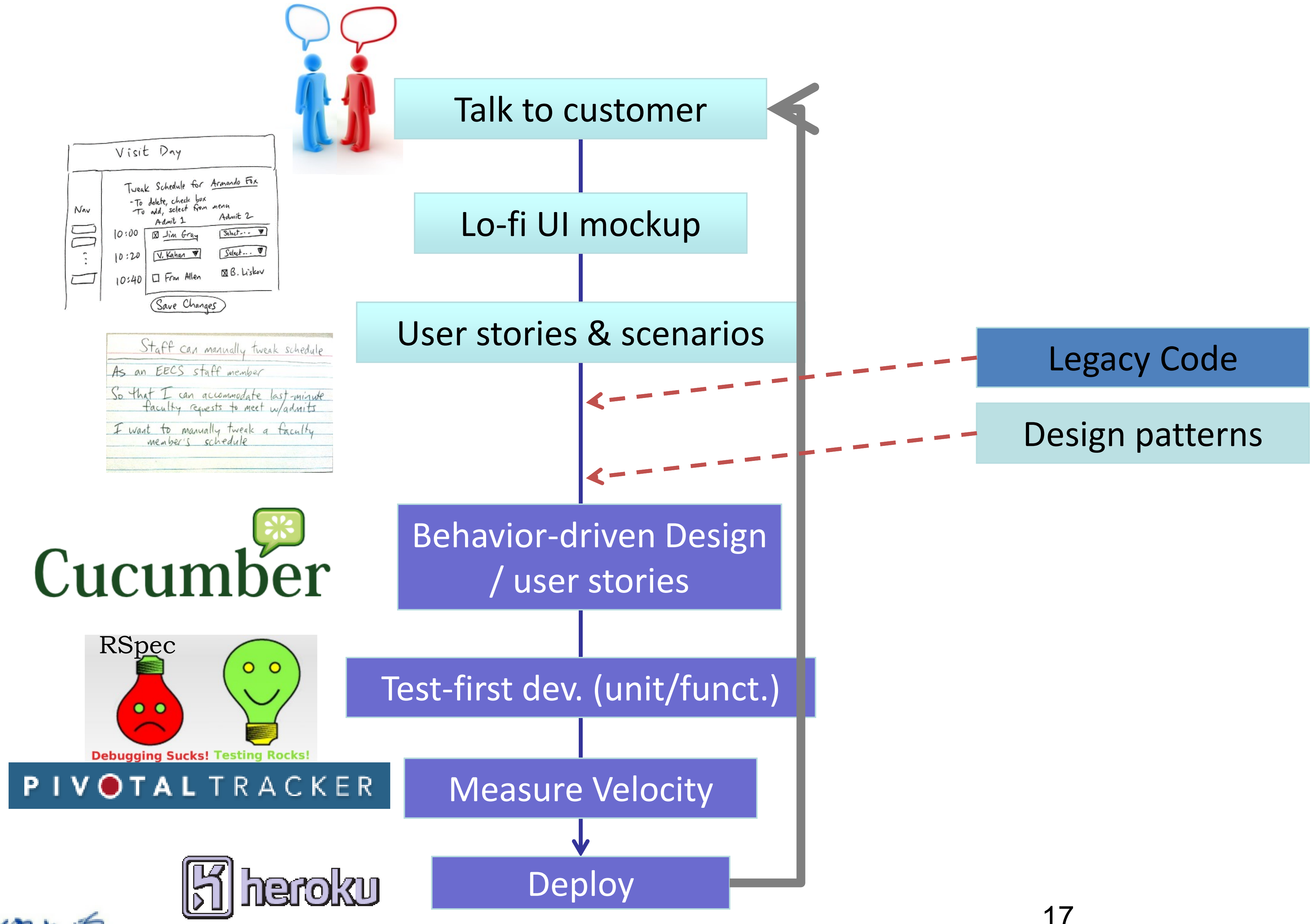
About Authors

- **Armando Fox** : Professor of Computer Science at UC Berkeley and the Faculty Advisor to the UC Berkeley MOOCLab. During his previous time at Stanford, he received teaching and mentoring awards from the Associated Students of Stanford University, the Society of Women Engineers, and Tau Beta Pi Engineering Honor Society.
- **David Patterson** : the Pardee Professor of Computer Science at UC Berkeley. In the past, he served as Chair of Berkeley's CS Division, Chair of the CRA, and President of the ACM. His best-known research projects are Reduced Instruction Set Computers (RISC), Redundant Arrays of Inexpensive Disks (RAID), and Networks of Workstations (NOW)

Contents

- **Introduction to SaaS and Agile Development**
- **The Architecture of SaaS Applications**
- **SaaS Framework**
  - **Introduction to Ruby**
  - **Introduction to Rails, Advanced Rails**
- **SaaS Client Framework: JavaScript Introduction**
- **Requirements: BDD and User Stories**
- **Testing: Test-Driven Development**
- **Maintenance: Legacy, Refactoring, and Agile**
- **Project Management: Scrum, Pairs, and VCS**
- **Design Patterns for SaaS Classes**
- **Performance, Releases, Reliability, and Security**

Course in 2 slides



Talk to customer

Lo-fi UI mockup

User stories & scenarios

Behavior-driven Design / user stories

Test-first dev. (unit/funct.)

Measure Velocity

Deploy

Legacy Code

Design patterns

Visit Day

Nav

Tweak Schedule for Armando Fox
- To delete, check box
To add, select from menu

	Admit 1	Admit 2
10:00	☒ Jim Gray	Select... ▼
10:20	V. Kahan ▼	Select... ▼
10:40	□ Fran Allen	☒ B. Liskov

Save Changes

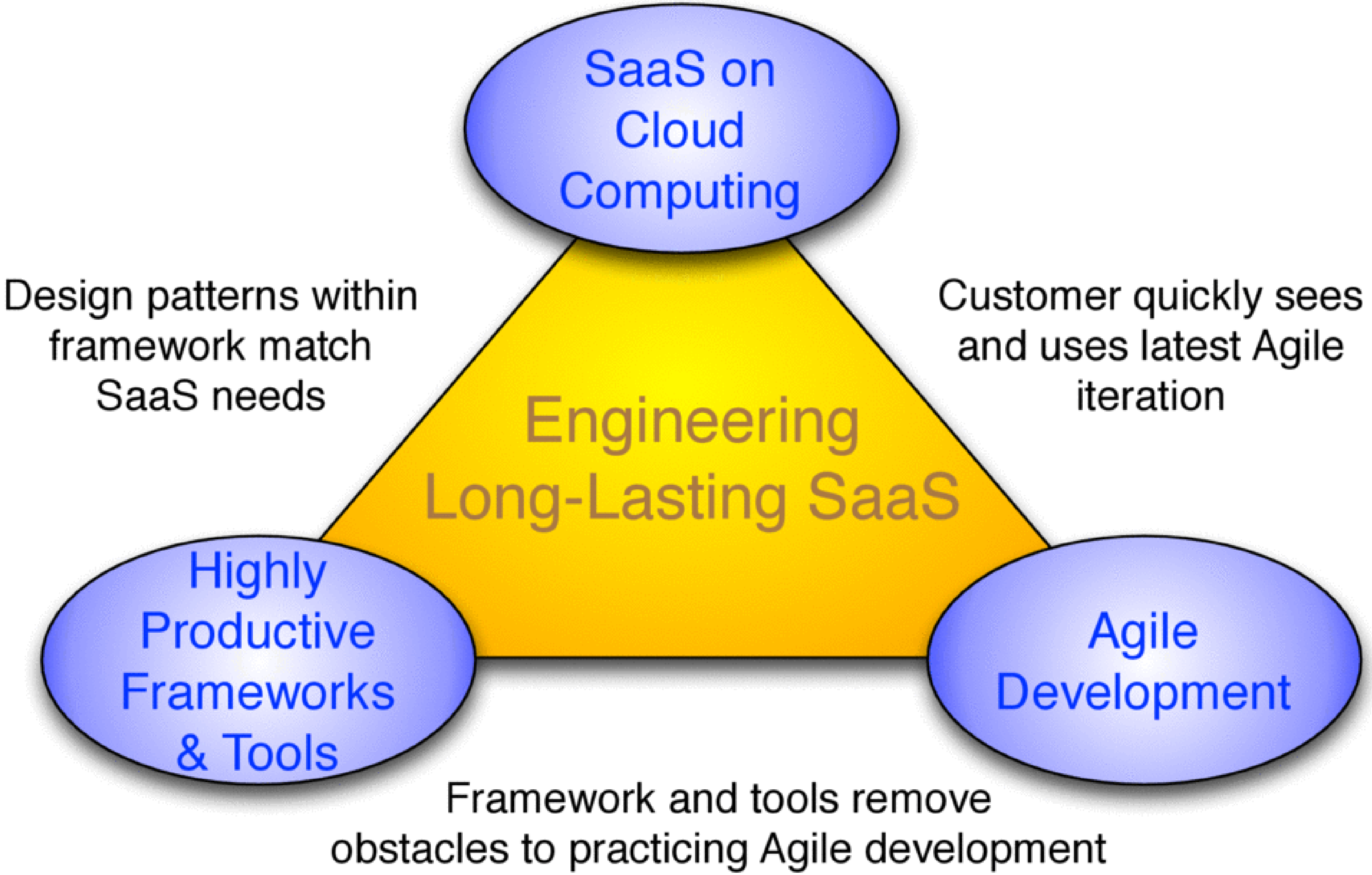
Staff can manually tweak schedule

As an EECS staff member

So that I can accommodate last-minute faculty requests to meet w/admits

I want to manually tweak a faculty member's schedule

Course in 2 slides



SaaS on Cloud Computing

Design patterns within framework match SaaS needs

Customer quickly sees and uses latest Agile iteration

Engineering Long-Lasting SaaS

Highly Productive Frameworks & Tools

Agile Development

Framework and tools remove obstacles to practicing Agile development

Online Resources

http://www.saasbook.info/students

- Online "get started" tutorials on ruby in case want to get feet wet before Ruby lectures
- Pointers to online HTML/CSS tutorials in in case want to get feet wet before Rails material
- Tutorials on other tools (GitHub, etc.)
- Feel free to suggest more on Piazza

Homework

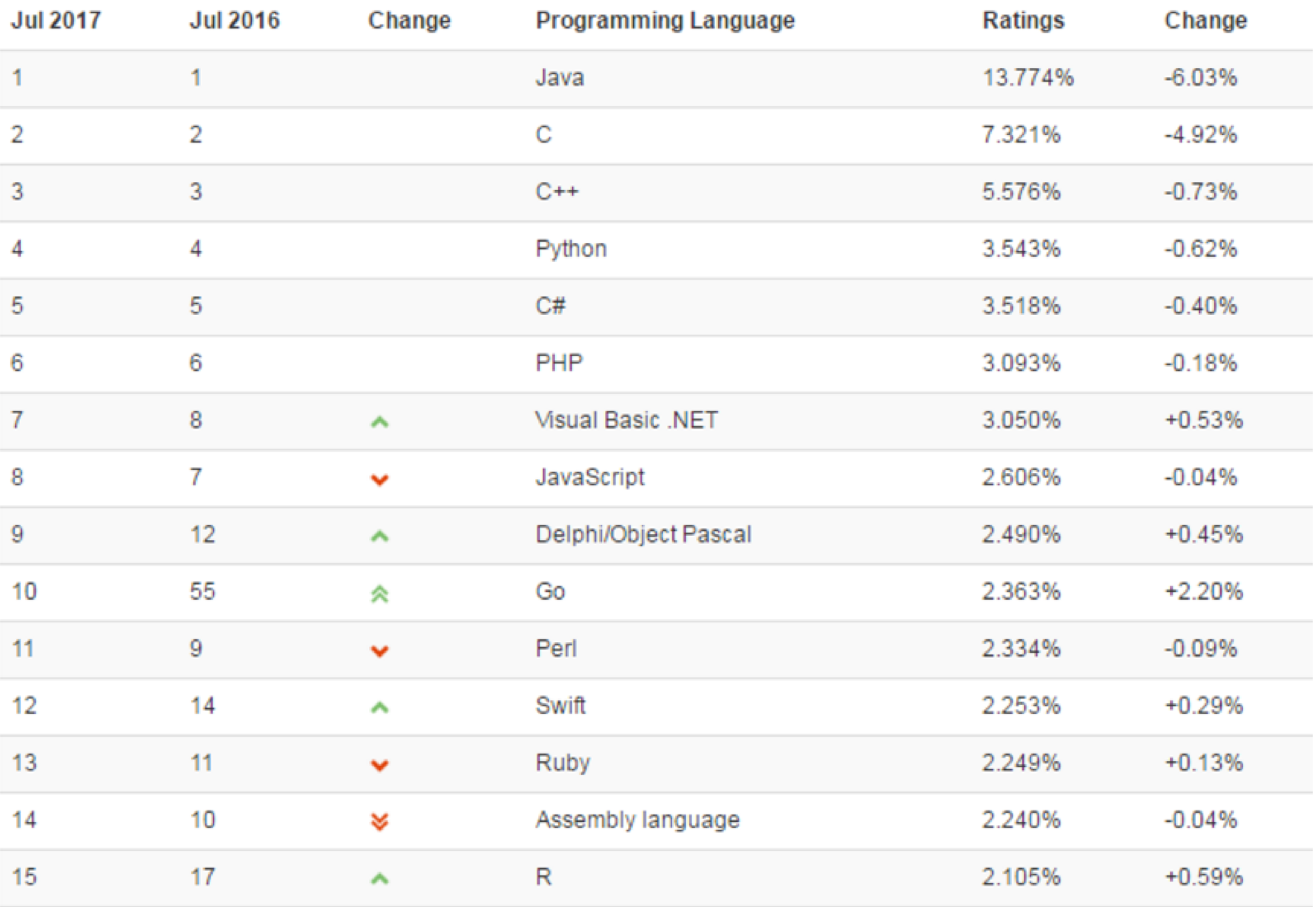
- 学习《Ruby基础教程》，熟悉Ruby编程

- 学习《Web开发敏捷之道--应用Rails进行敏捷Web开发》——”Agile Web Development with Rails”，熟悉基于Ruby/Rails 开发Web应用

- 学习 JavaScript

编程语言的状况

编程语言排行榜 TOP 20 榜单：



Jul 2017	Jul 2016	Change	Programming Language	Ratings	Change
1	1		Java	13.774%	-6.03%
2	2		C	7.321%	-4.92%
3	3		C++	5.576%	-0.73%
4	4		Python	3.543%	-0.62%
5	5		C#	3.518%	-0.40%
6	6		PHP	3.093%	-0.18%
7	8	↑	Visual Basic .NET	3.050%	+0.53%
8	7	↓	JavaScript	2.606%	-0.04%
9	12	↑	Delphi/Object Pascal	2.490%	+0.45%
10	55	⇈	Go	2.363%	+2.20%
11	9	↓	Perl	2.334%	-0.09%
12	14	↑	Swift	2.253%	+0.29%
13	11	↓	Ruby	2.249%	+0.13%
14	10	⇊	Assembly language	2.240%	-0.04%
15	17	↑	R	2.105%	+0.59%