Autolab: An Online Autograding Service for the World

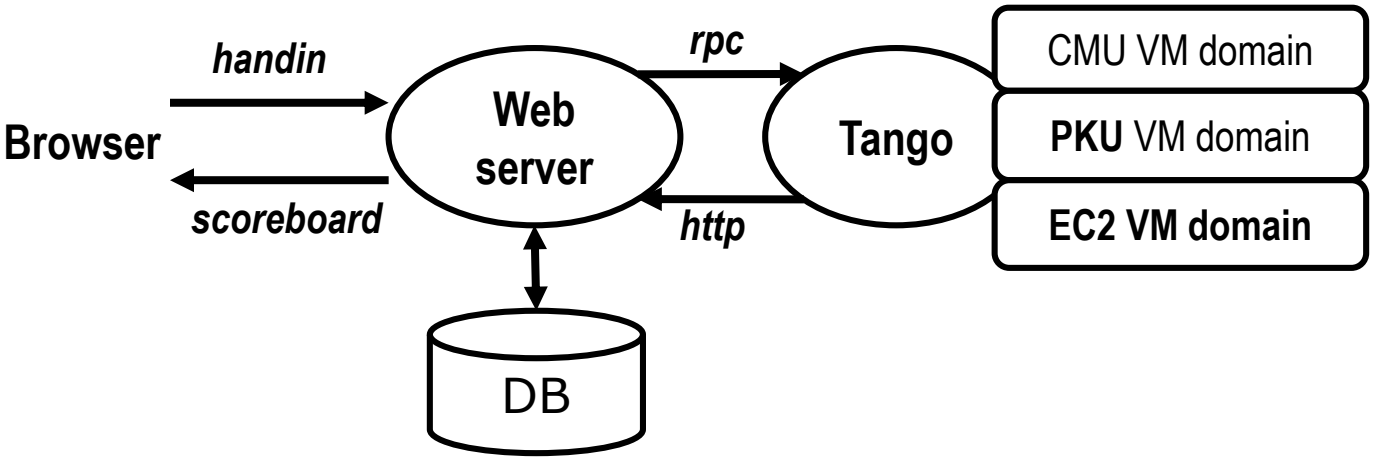
http://autolab.cs.cmu.edu

Autolab Introduction

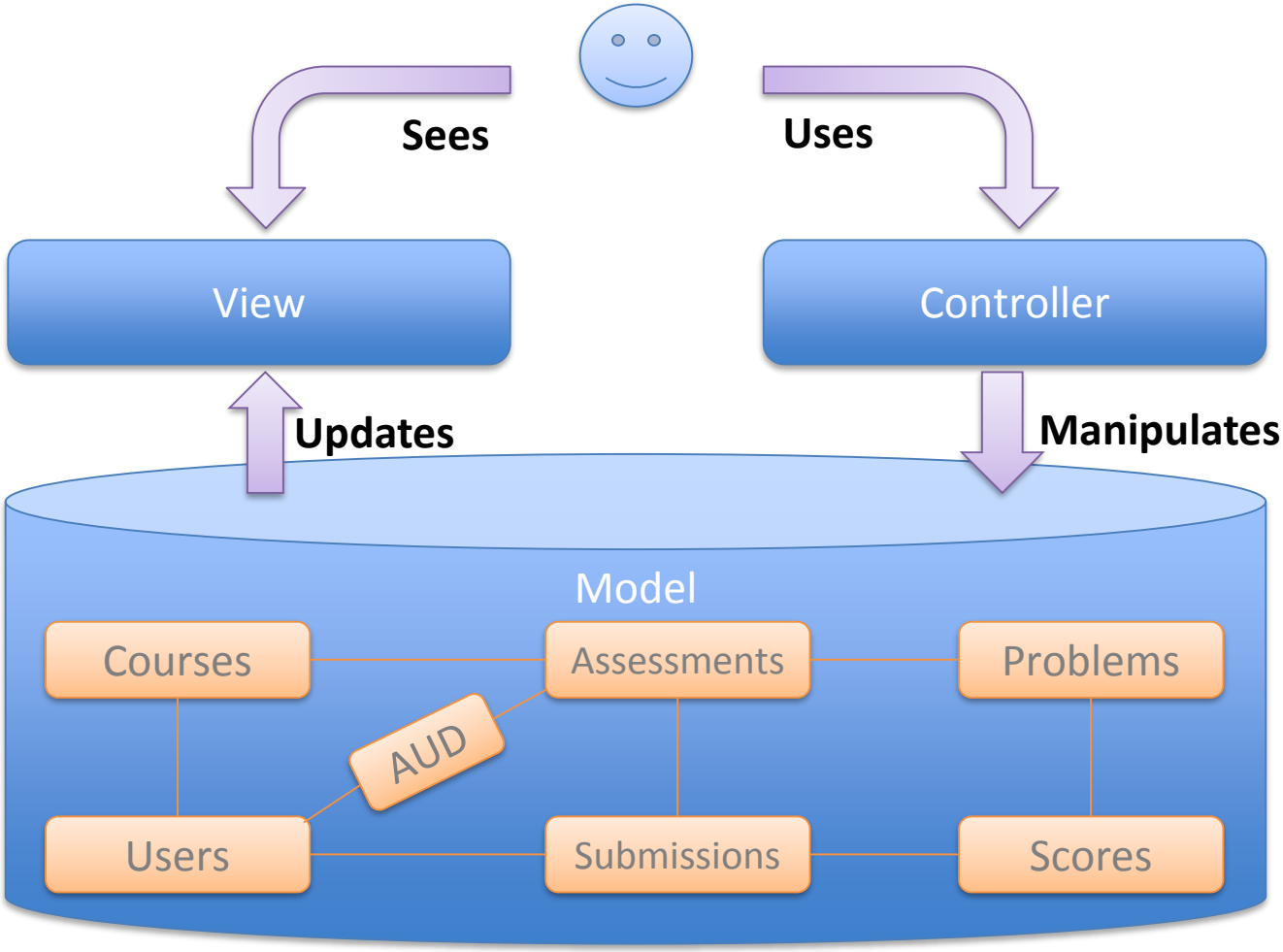
- **An online autograding service that allows instructors to offer programming assignments over the Internet.**
- **Two key ideas: autograding and scoreboards**
  - Autograding:
    - Programs evaluating the quality of other programs.
    - Student handins automatically autograded on secure VMs.
  - Scoreboard
    - Scores are posted in real-time on sorted class scoreboard.
    - Students anonymize themselves with nicknames.
      - "kill -9 15213", "213 makes me ANSI"!
- **With Autolab you can use your Web browser to:**
  - Download the lab materials
  - Handin your code for autograding by the Autolab server
  - View the class scoreboard
  - View the complete history of your code handins, autograded results, instructor's evaluations, and gradebook.
  - View the TA annotations of your code for Style points.

Autolab系统组成

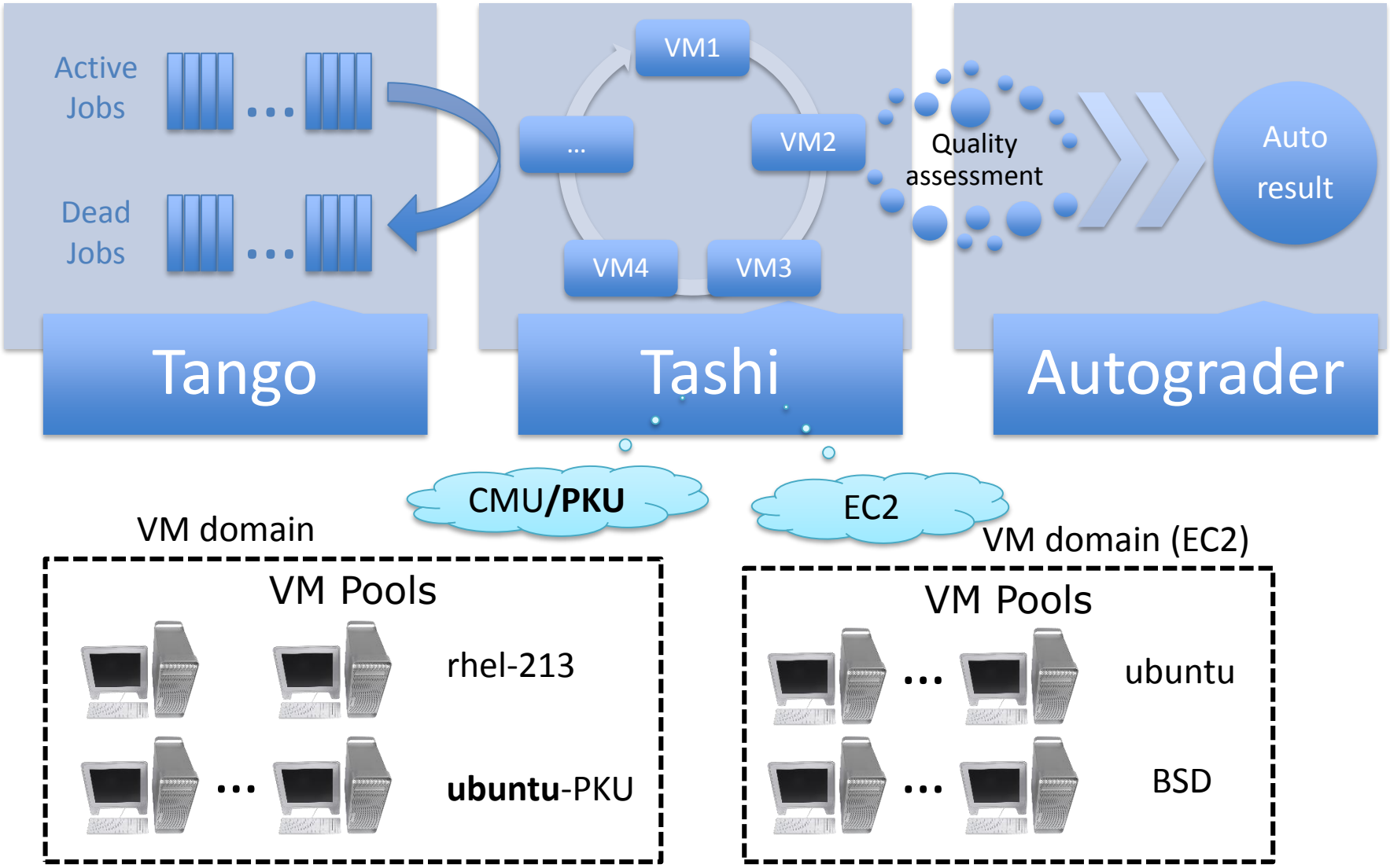
层次结构		基本功能
前端 Front-End	WebServer	用户交互、数据处理
	Database	数据管理、系统管理
后端 Back-End	Tango	任务管理、远程过程调用
	Tashi	集群管理器、节点管理器



前端（Front-End）


Sees
Uses
View
Controller
Updates
Manipulates
Model
Courses
Assessments
Problems
AUD
Users
Submissions
Scores

后端（Back-End）


Active Jobs
Dead Jobs
Tango
VM1
VM2
...
VM4
VM3
Tashi
Quality assessment
Auto result
Autograder
CMU/PKU
EC2
VM domain
VM Pools
rhel-213
ubuntu-PKU
VM domain (EC2)
VM Pools
ubuntu
BSD

Autolab accounts

- **Autolab网站：**
  - 浏览器登录：https://162.105.31.220
- **Autolab账号：**
  - 学生账号：学号@pku.edu.cn
    - 每个学生属于某个section：`S1801, S1802, ...`
  - 助教账号：学号@pku.edu.cn
    - 助教的权限是CA，可以对section进行管理
  - 密码：系统会自动给每个账号发信通知密码
    - Note 1：该信件可能被北大邮箱拒收，或者可能被放到垃圾邮箱里，请注意查询
    - Note 2：如未收到邮件或忘记密码，可登录Autolab网站，页面下方有个 Forgot password? 进去后输入自己的邮箱即可获得新密码的邮件

Linux accounts

- **Labs will use the Linux Server**
  - Linux服务器IP：162.105.31.232
  - ssh登录（登录端口为缺省端口22）
    - Windows下可以使用支持ssh协议的应用软件，如putty
    - Linux下可在终端上使用ssh命令
  - 学生账号：Section+学号
    - Section：S1801，S1802，…
    - 注意：Section和学号中间没有+号，举例：S18011700012345
  - 助教账号：Section+CA
    - Section：S1801，S1802，…
    - 举例：S1801CA
  - 密码：请询问助教

Change your password ASAP: `passwd`

Lab Rationale

- **Each lab has a well-defined goal such as solving a puzzle or winning a contest**
- **Doing the lab should result in new skills and concepts**
- **We try to use competition in a fun and healthy way**
  - Set a reasonable threshold for full credit
  - Post intermediate results (anonymized) on Autolab scoreboard for glory!
- **Work groups**
  - You must work alone on all lab assignments
- **Handins**
  - Electronic handins using Autolab (no exceptions!)

Timeliness and version limit

- **Grace days**
  - **5 grace days for the semester**
  - **Limit of 0, 1, or 2 grace days per lab used automatically**
  - Covers scheduling crunch, out-of-town trips, illnesses, minor setbacks
  - Save them until late in the term!
- **Lateness penalties**
  - Once grace day(s) used up, get penalized **10% or 20% per day**
  - No handins later than **3 days after due date**
- **Late Slack: 3600**
  - This is the number of seconds after a deadline that the server will still accept a submission and not count it as late.
- **Default Version Threshold: 16**
  - If a submission's version is greater than this threshold, it is penalized according to the version penalty.
- **Version Penalty: 10% or 20% points**
  - The penalty applied to submissions with versions greater than the version threshold.

Cheating/Plagiarism: Description

- **Unauthorized use of information**
  - Borrowing code: by copying, retyping, **looking at** a file
  - Describing: verbal description of code from one person to another.
  - Searching the Web for solutions
  - Copying code from a previous course or online solution
  - Reusing your code from a previous semester (here or elsewhere)
    - Arrange meeting with instructor before reusing your old solutions

Cheating/Plagiarism: Description (cont.)

- **Unauthorized supplying of information**
  - Providing copy: Giving a copy of a file to someone
  - Providing access:
    - Putting material in unprotected directory
    - Putting material in unprotected code repository (e.g., Github)
      - Or, letting protections expire
  - Applies to this term and the future
    - There is no statute of limitations for academic integrity violations
- **Collaborations beyond high-level, strategic advice**
  - Anything more than block diagram or a few words
  - Code / pseudo-code is NOT high level
  - Coaching, arranging blocks of allowed code is NOT high level
  - Code-level debugging is NOT high level

Cheating/Plagiarism: Description

- **What is NOT cheating?**
  - Explaining how to use systems or tools
  - Helping others with *high-level* design issues
    - High means very high
  - Using code supplied by us
    - Starter code, class examples
  - Using code from the CS:APP web site
- **Attribution Requirements**
  - Starter code: No
  - Other allowed code (course, CS:APP): Yes
  - Indicate source, beginning and end

Cheating: Consequences

- **Penalty for cheating:**
  - Best case: -100% for assignment
    - You would be better off to turn in nothing
  - Worst case: failing the course
  - Loss of respect by you, the instructors and your colleagues
  - If you do cheat – come clean asap!
- **Detection of cheating:**
  - We have sophisticated tools for detecting code plagiarism
- **Don't do it!**
  - Manage your time carefully
  - Ask the staff for help when you get stuck

Some Concrete Examples:

- **This is Cheating:**
  - Searching the internet with the phrase 15-213, 15213, 213, 18213, malloclab, etc.
    - That's right, just entering it in a search engine
  - Looking at someone's code on the computer next to yours
  - Giving your code to someone else, now or in the future
  - Posting your code in a publicly accessible place on the Internet, now or in the future
  - Hacking the course infrastructure
- **This is OK (and encouraged):**
  - Googling a man page for fputs
  - Asking a friend for help with gdb (but not with your code)
  - Asking a TA or course instructor for help, showing them your code, …
  - Using code examples from book (with attribution)
  - Talking about a (high-level) approach to the lab with a classmate

Why It's a Big Deal

- **This material is best learned by doing**
  - Even though that can, at times, be difficult and frustrating
  - Starting with a copy of a program and then tweaking it is very different from writing from scratch
    - Planning, designing, organizing a program are important skills
- **We are the gateway to other system courses**
  - Want to make sure everyone completing the course has mastered the material
- **Industry appreciates the value of this course**
  - We want to make sure anyone claiming to have taken the course is prepared for the real world
- **Working in Teams and Collaboration is an Important Skill**
  - But only if team members have solid foundations
  - This course is about foundations, not teamwork

How to do with the labs

- **Start early**
- **Don’t rely on marathon programming sessions**
  - Your brain works better in small bursts of activity
  - Ideas / solutions will come to mind while you’re doing other things
- **Plan for stumbling blocks**
  - Assignment is harder than you expected
  - Code doesn’t work
  - Bugs hard to track down
  - Life gets in the way
    - Minor health issues
    - Unanticipated events