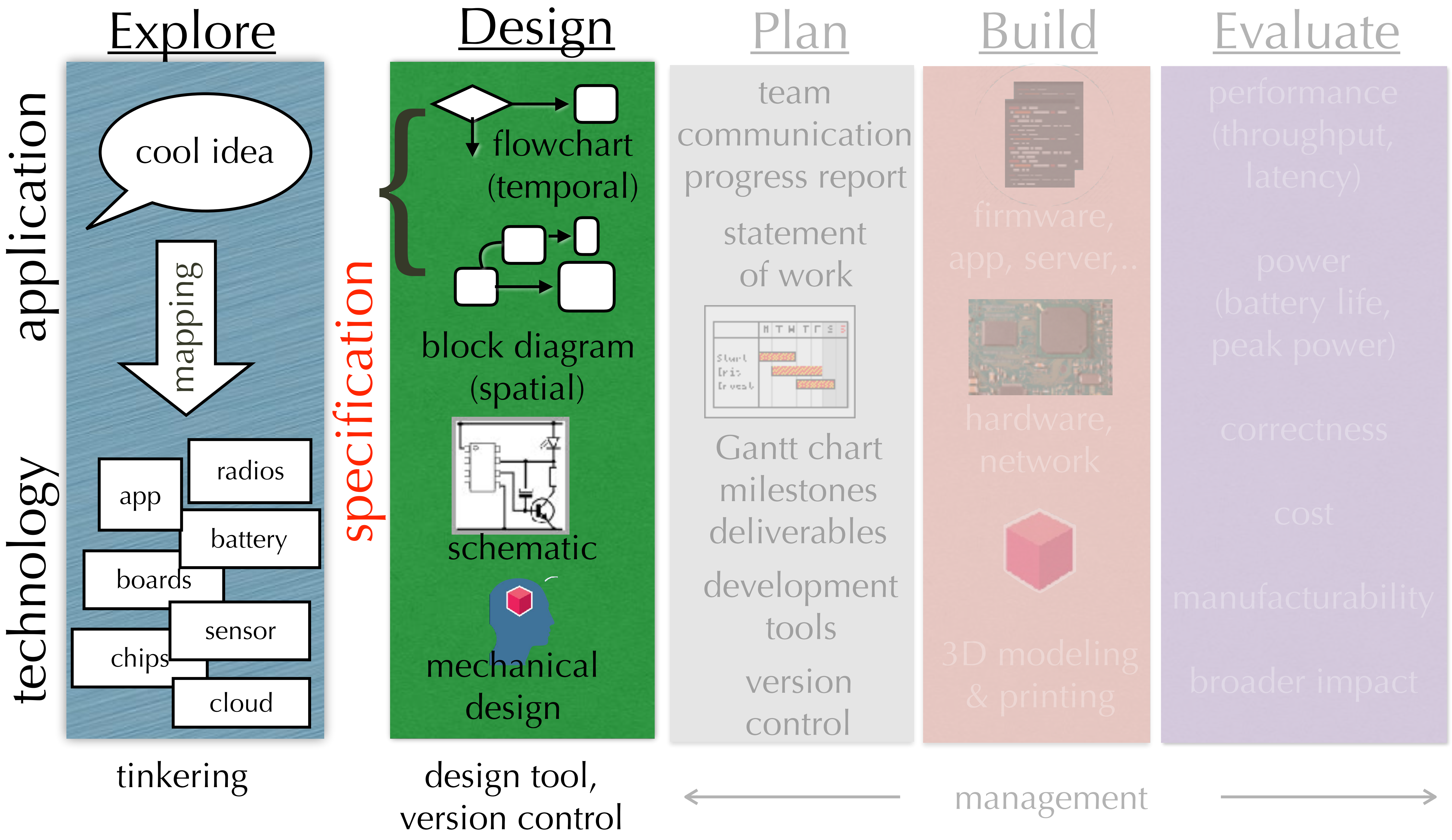
EECS 159A/CSE 181A:

Specification

Problem Statement

- Must know what problem you are solving!
  - may sound obvious, but painfully true
  - Figuring out the right problem statement can be half of the work!
- Refinement of problem statement
  - From the end-user's point of view
  - Translate into technical (engineering) specification
  - consider requirements, constraints, objectives

From Exploration Specification before Design



Explore

application

cool idea

mapping

technology

app

radios

battery

boards

sensor

chips

cloud

tinkering

specification

Design

flowchart (temporal)

block diagram (spatial)

schematic

mechanical design

design tool, version control

Plan

team communication progress report

statement of work

Gantt chart milestones deliverables

development tools

version control

Build

firmware, app, server,..

hardware, network

3D modeling & printing

Evaluate

performance (throughput, latency)

power (battery life, peak power)

correctness

cost

manufacturability

broader impact

management

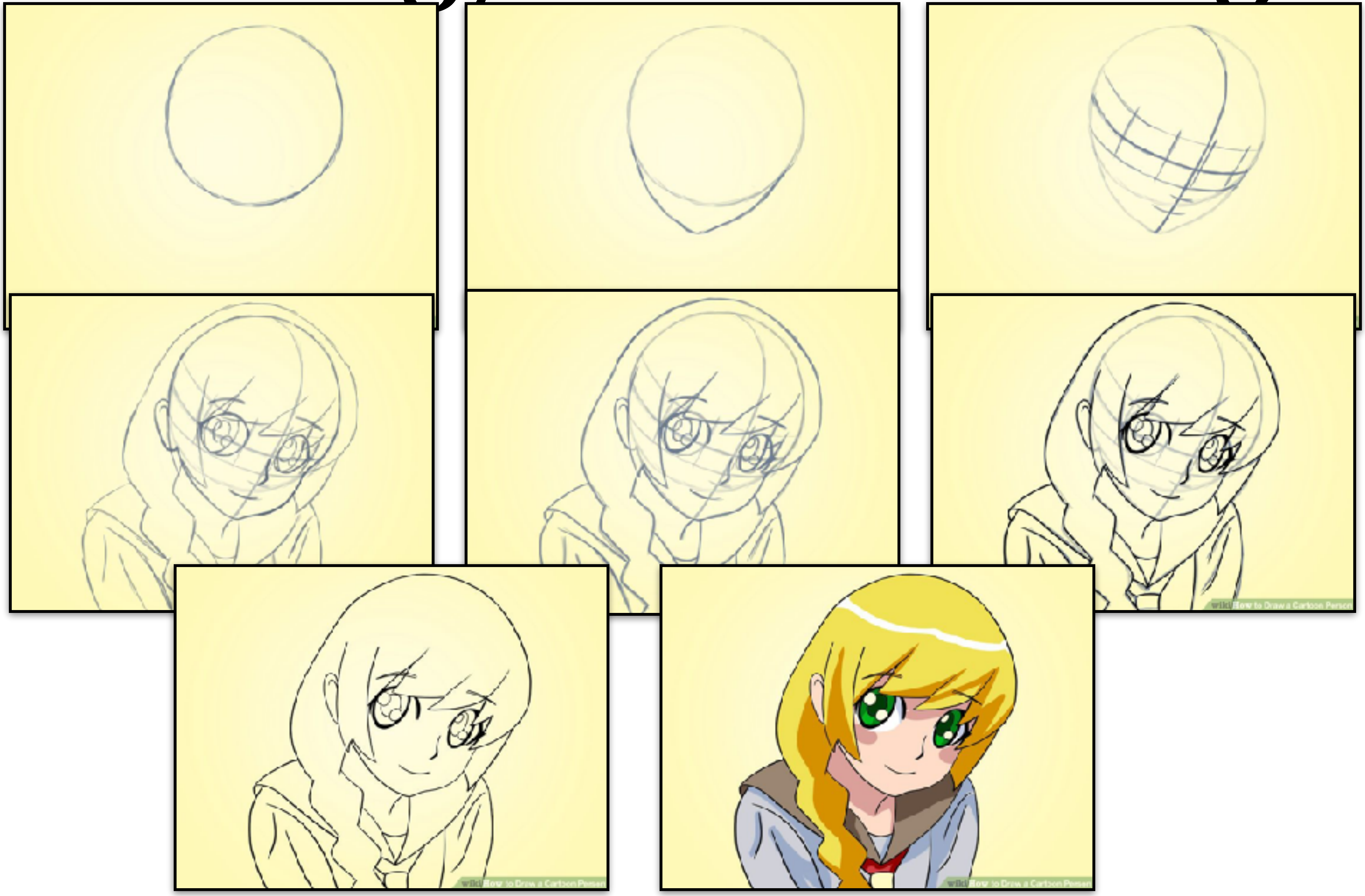
Design process: analogy with drawing

This is what you want to draw ("build")

- How do you start?
- From a color pen?
- Drawing all details?
- Add one feature at a time?

Sketch it, then fill in the details

Analogy with Drawing


wikiHow to Draw a Cartoon Person
wikiHow to Draw a Cartoon Person
wikiHow to Draw a Cartoon Person

Technical Specification

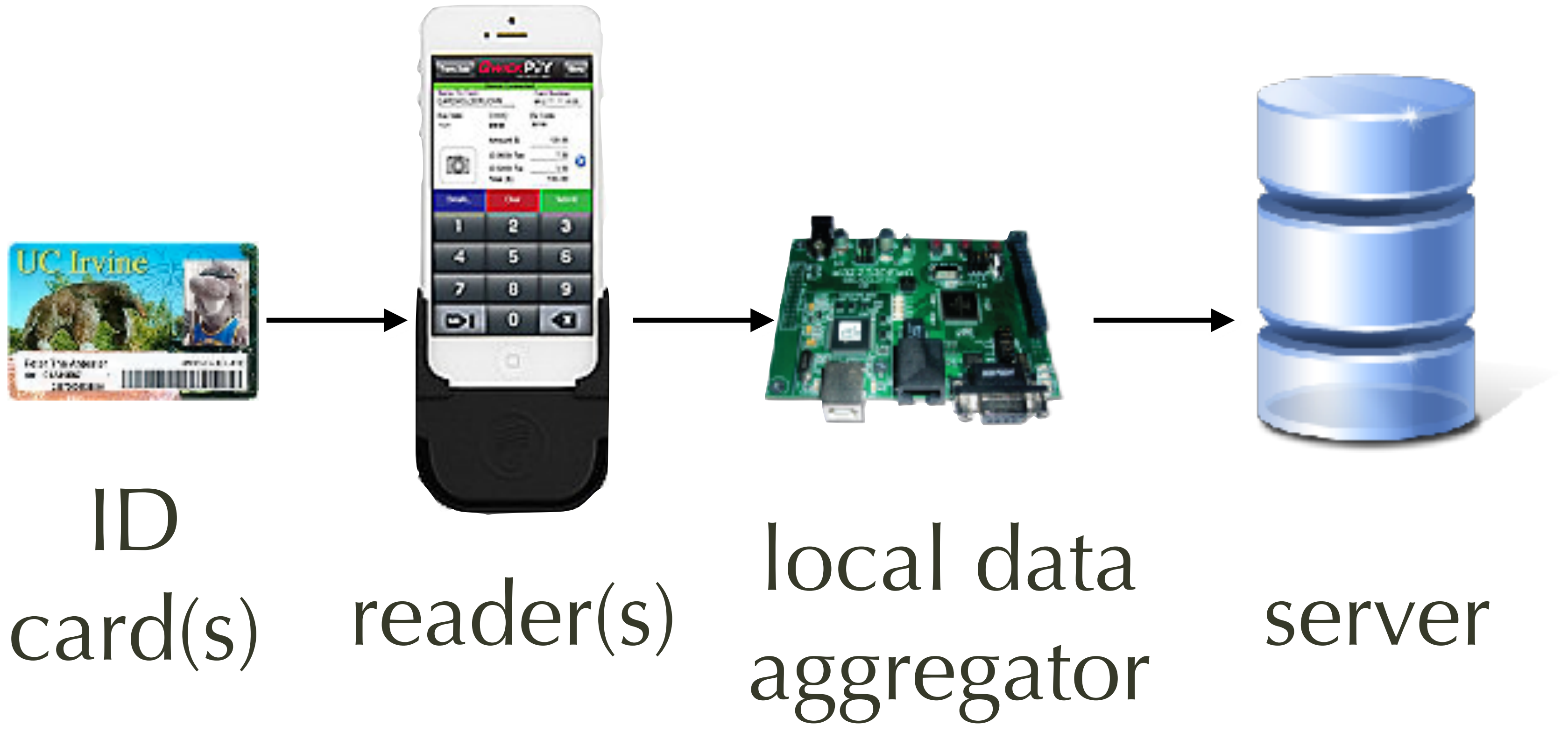
- What the boss tells the engineers to build
  - Could be a little or a lot of details
- Underspecification
  - Not a lot of details; very sketchy
  - Engineers have a lot of freedom to choose
  - Engineers may have to guess what the boss wants
- Overspecification
  - A lot of details all given
  - little freedom to explore potentially more elegant solutions

Scope of Specification

- Structural
  - The organization of the system as connected subsystems
- Behavioral
  - The way it interacts w/ human or other system
- Mechanical
  - Size, dimensions, shape, weight
  - Strength (drop test), waterproofness
- Performance
  - Speed, latency, throughput, battery life, energy consumption

Structural Specification (spatial view)

- Block diagram or schematic
- e.g.,: attendance system using ID card
  - Commits to using ID card, magstripe, iPhone
  - Underspecifies aggregator and server



ID card(s)

reader(s)

local data aggregator

server

Behavioral Specification (temporal view)

- State machine, flowchart, pseudocode
- Example: attendance system
  - Configuration
    - Administrator and user setup
    - Device setup
  - Deployment
    - Scan card
    - Upload data

Behavioral Spec: Global Scope

- Behavior across subsystems
- e.g., Scanning:
  - user swipes card, card reader converts magstripe data, sends to app, app adds time stamp and location, starts a transaction with gateway;
  - gateway logs local copy of data record, checks for duplicate records, starts transaction with database;
  - database receives data record, checks validity, sends acknowledgment

Behavioral Spec: Local Scope

- Behavior local to a subsystem
- Example local behavioral specs
  - App: potentially UI centric
    - Device configuration: driver, version check, ..
    - User administration: registration, assign rights
    - Scanning action: wait for user to swipe, local logging, upload data
  - Database: transaction centric

From Global to Local Behavior

- Global
  - necessary to capture application
  - stated to fulfill a requirement in application
- Local
  - necessary to realize each subsystem
  - Combine projection of global behavior with subsystem-specific tasks
- Principle: separation of concerns
  - Local behavior should be “modular”, parameterized

Example: Projecting Global Scanning behavior to Local

- App:
  - user swipes card, card reader converts magstripe data, sends to app, app adds time stamp and location, starts a transaction with gateway;
- Gateway:
  - logs local copy of data record, checks for duplicate records, starts transaction with database;
- Database
  - receives data record, checks validity, sends acknowledgment

Local Behavioral Spec

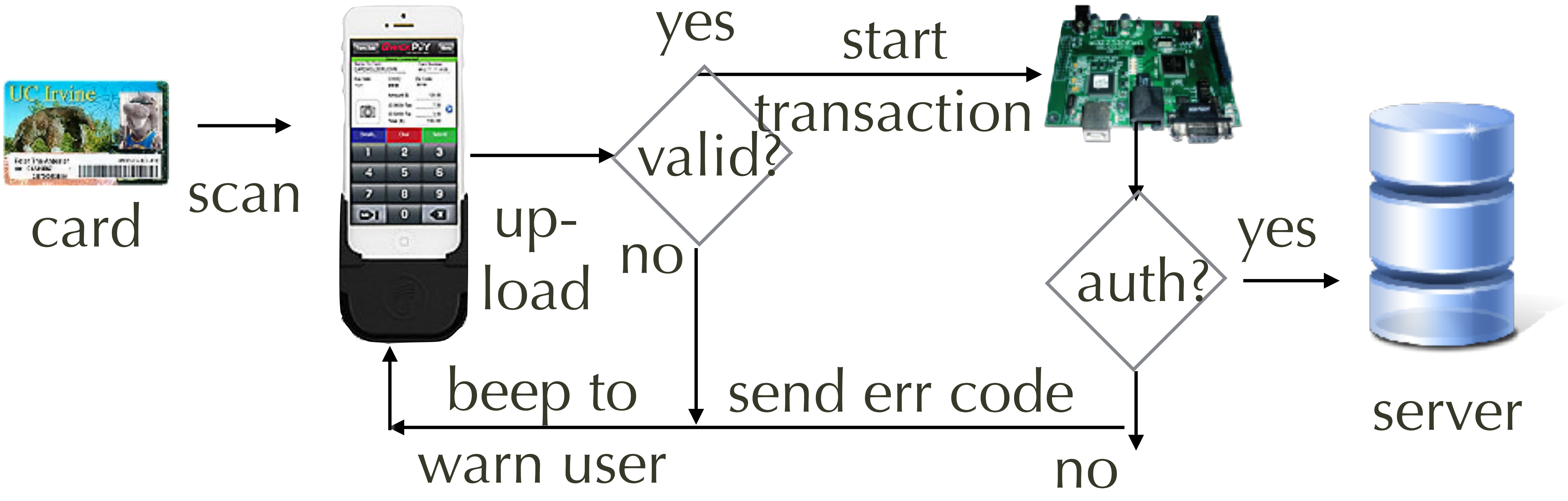
- Modular interface
  - matches the higher-level needs of application
  - underneath: a general, more generic subsystem
- Example: database
  - API for receiving structured data (ID, timestamp)
  - Underneath: generic database (does not know the meaning of ID or timestamp => just strings)
  - Why? easier to develop and test separately!

Behavioral Refinement

- Translating high-level behavior into detailed actions
  - This is essentially just programming
  - Generalized to software (and hardware to some extent)
- Steps involved for programming
  - Write high-level pseudocode first, using high-level functions
  - Write the functions or subroutines to invoke, using lower-level functions
  - Write the lower-level functions…

Behavioral vs. Structural Specification

- They are separate views!
  - Behavioral: pseudocode, flowchart, FSM
  - Structural: block diagram, schematic, etc
- Common mistake is to mix them



Underspecification: Not enough details

- Example: boundary cases
  - What if you scan a non-UCI card or ID? Should it beep? give warning? fail silently?
  - What if the memory card is full?
- Why is underspecification a problem?
  - Undocumented, difficult to track
  - Could be very costly to change, breaking a lot of code

Overspecification: Dictates too much

- Use a very specific platform or technology
  - e.g., assumes a smartphone =>
    rules out embedded system implementation
  - e.g., "Uses an Arduino with Xbee module …"
    => rules out many other embedded platforms, such as Intel Edison, Raspberry Pi, etc
  - e.g., must use a 9V battery => why not other types?
- Why can this be a problem (in early stage)?
  - Limits ability to satisfy constraints and objectives
  - Not a problem if constraints, objectives, requirements can all be met

From Exploration to Specification

- Exploration
  - Lists technology options and criteria
  - Rank-order the “selling points”
- Generating the specification
  - Identify & translate **constraints** from different levels
  - Translate rank-ordered “selling points” into **objectives**
  - Translate application scenarios into **requirements**
- Outcome: specification for the project

Definitions

- **Constraints**:
  - Prescriptive *limits* on the system
- **Objectives (or objetive functions)**:
  - Rank-ordered criteria for being *good*
- **Requirements**:
  - Descriptive criteria for being *correct*

Constraints

Prescriptive limits on the system

Constraints: limits what you are allowed to do, use, or build

- weight (max, maybe min)
- volume (min, max)
- cost: (max) bill of materials (BOM)
- latency: (max, maybe min)
- throughput: (usually min)
- RF power level: (min, max)
- heat generation: (max)
- and more…

Constraints imposed by

- Application
  - customer, boss, marketing team, etc.
- Regulatory agencies
  - FCC - Federal Communications Commission (food)
  - FDA - Federal Food & Drug Administration (med. dev)
  - FAA - Federal Aviation Administration (quadcopters)
  - EPA - Environmental Protection Agency (disposal)
  - NHTSA - National Highway Traffic Safety Admin. (car)
  - TSA - Transportation Security Administration (in-flight)

Constraints: Various Budgets

- Power and energy budget:
  - How many watts (or mW) average? peak? standby?
  - How many watt-hours, mWh, mAh total energy?
  - How do these translate into heat dissipation?
- Cost budget:
  - How many $$ costs in parts to prototype
  - How many $$ in BOM cost? Manufacturing cost?
- Size and weight budget:
  - How many mm^2 area or mm^3 volume?
  - How many grams in weight? (how much is battery?)

Example BOM costs of different designs

Product	AntScan	ZotScan
Cost	$50	$175
Weight	100 grams	125 grams
Battery life	200 h / 1 AAA	3 h / 6 AA
Formats	MagStripe	1D, 2D QR
Simultaneous	100 scanners	3 scanners
RF Range	20 m LOS	100 m
Rx Unit	extra	WiFi AP or PC

- Which is better?
- What does it mean to be “good”?

Objectives

(or objective functions)

Rank-ordered criteria for being “good”

Objectives

- The word “objective” is heavily overloaded
  - Could mean your goal, direction, etc., but could be qualitative
- This class defines objectives as
  - Rank-ordered criteria for being “good”
  - Correctness is a given (i.e., requirements and constraints are already satisfied)

Possible Objectives

- Functional:
  - Feature-richness,
  - battery life,
  - responsiveness,
  - robustness,
  - scalability
- Nonfunctional:
  - Price
  - weight
  - aesthetics
  - tech-support?

Objectives determine which one is better

- if cost or weight is more important,
  => AntScan is better
- if convenience or RF range is more important,
  => ZotScan is better

Product	AntScan	ZotScan
Cost	$50	$175
Weight	100 grams	125 grams
Battery life	200 h / 1 AAA	3 h / 6 AA
Formats	MagStripe	1D, 2D QR
Simultaneous	100 scanners	3 scanners
RF Range	20 m LOS	100 m
Rx Unit	extra	WiFi AP or PC

Rank-ordered Objectives

- What if you want multiple objectives?
  - => need to prioritize them!
  - e.g., primary: battery life, secondary: power

Product	AntScan	ZotScan
Cost	$50	$175
Weight	100 grams	125 grams
Battery life	200 h / 1 AAA	3 h / 6 AA
Formats	MagStripe	1D, 2D QR
Simultaneous	100 scanners	3 scanners
RF Range	20 m LOS	100 m
Rx Unit	extra	WiFi AP or PC

Trade-offs based on Objectives

- Cannot always meet all objectives
  - Need to prioritize objectives
  - Give priority to certain features over others => making a trade-off
- Worst case: go back to the specification
  - Check which part is overspecified => relax the overspecification, have more flexibility to consider other options

Requirements

- Properties that the system must have in order to be considered “correct”
- Several kinds of requirements
  - functional
  - performance: timing, throughput, scalability…
  - form factor
  - materials

Requirements from two different points of view

- End-User
  - Description of system operation
  - "Use Cases": scenarios to help illustrate requirements
- Technical
  - Functional: what does it do in technical terms
  - Nonfunctional: form factor, ruggedness, weight, …

From User's Point of View

- Purpose: What can you use it for?
  - Taking attendance? Store checkout?
- Operation: (functional)
  - Any install step? Setup? Configuration?
  - How does the user know it's ready to use? Does it have auto-sleep mode? wake up?
- Form factor
  - Should it be portable? How much should it weigh?
  - Should it be shaped in a way that's easy to grab? How big should it be?

User’s view cont’d: Feedback vs. Access

- Feedback
  - How does user know if scanning is ok or failed?
  - How does the user know if Data has been sent successfully to the server? Or logged locally?
- Access
  - Who is allowed to view what part of data? Read vs. Read/Write?
  - How does the user view data? (browser, proprietary GUI, SD card in card reader?

End-User: Interoperability

- Assumption on user-provided setup:
    - Gateway? PC? Server? PDA/Phone?
- Standards Compliance
    - What kind of barcode can it read? 2D? What are the barcode standards?
    - Privacy? protection against hackers?

Functional Requirements

- What are the essential subsystems?
- What does it do?
  - input-output “transfer” characteristics
  - stateful vs. stateless behavior?
- How do different parts interact with each other?
- Key point:
  - Try to stay at the level of “what the system should do”, while allowing many implementation options

1. Economic Constraints

- Cost
  - Bill of material (BOM) at a given quantity
  - Manufacturing, packaging, shipping costs
- Market
  - Who pays for the system? the support? (e.g., cloud)
  - Do you envision a company be selling and supporting this system?

2. Environmental Constraint

- Constraints imposed by intended operating environment
  - availability of Wi-Fi network?
  - available space for installing the reader?
  - use of attendance system during a field trip?
- Other environmental constraint
  - Use of RoHS (restriction of hazardous substance)
  - Use of disposable supplies (e.g., fingerprint)

3. Social Constraints

- Privacy issues
  - What is public info vs private info?
  - which part needs protection (encrypt, etc)
- Is the usage socially accepted?
  - how disruptive it is to the class?
  - is wearing an AR headset acceptable?
  - is wearing Google Glass acceptable socially?
- Physical size, possibly weight
  - or else looks strange

4. Political Constraints

- Country-specific policies
- Example: Google Map
  - not accessible in certain countries
- Export control laws
  - The Export Administration Regulations (“EAR”) regulate exports of commercial items with potential military applications (so called “dual-use” items).

Ten Categories of Commerce Control List

- 0 Nuclear Materials, Facilities & Equipment and Miscellaneous
- 1 Materials, Chemicals, “Microorganisms” and Toxins
- 2 Materials Processing
- 3 **Electronics**
- 4 **Computers**
- 5 **Telecommunications and Information Security**
- 6 **Sensors and Lasers**
- 7 **Navigation and Avionics**
- 8 Marine
- 9 Propulsion Systems, Space Vehicles and Related Equipment

5. Ethical Constraints

- Governed by code of ethics
- Example: IEEE Code of Ethics
  - *We, the members of the IEEE, in recognition of the importance of our technologies in affecting the quality of life throughout the world, and in accepting a personal obligation to our profession, its members and the communities we serve, do hereby commit ourselves to the highest ethical and professional conduct and agree:*

IEEE Code of Ethics (1/2)

1. to accept responsibility in making decisions consistent with the safety, health, and welfare of the public, and to disclose promptly factors that might endanger the public or the environment;

2. to avoid real or perceived conflicts of interest whenever possible, and to disclose them to affected parties when they do exist;

3. to be honest and realistic in stating claims or estimates based on available data;

4. to reject bribery in all its forms;

5. to improve the understanding of technology; its appropriate application, and potential consequences;

IEEE Code of Ethics (2/2)

6. to maintain and improve our technical competence and to undertake technological tasks for others only if qualified by training or experience, or after full disclosure of pertinent limitations;

7. to seek, accept, and offer honest criticism of technical work, to acknowledge and correct errors, and to credit properly the contributions of others;

8. to treat fairly all persons and to not engage in acts of discrimination based on race, religion, gender, disability, age, national origin, sexual orientation, gender identity, or gender expression;

9. to avoid injuring others, their property, reputation, or employment by false or malicious action;

10. to assist colleagues and co-workers in their professional development and to support them in following this code of ethics.

6. Health and safety

- Health issues
  - e.g., shared fingerprinting sensor: sanitation?
  - e.g., harmful radiation?
- Safety issues
  - e.g., fire hazard due to excessive heat?
  - e.g., tripping hazard of running long cords?

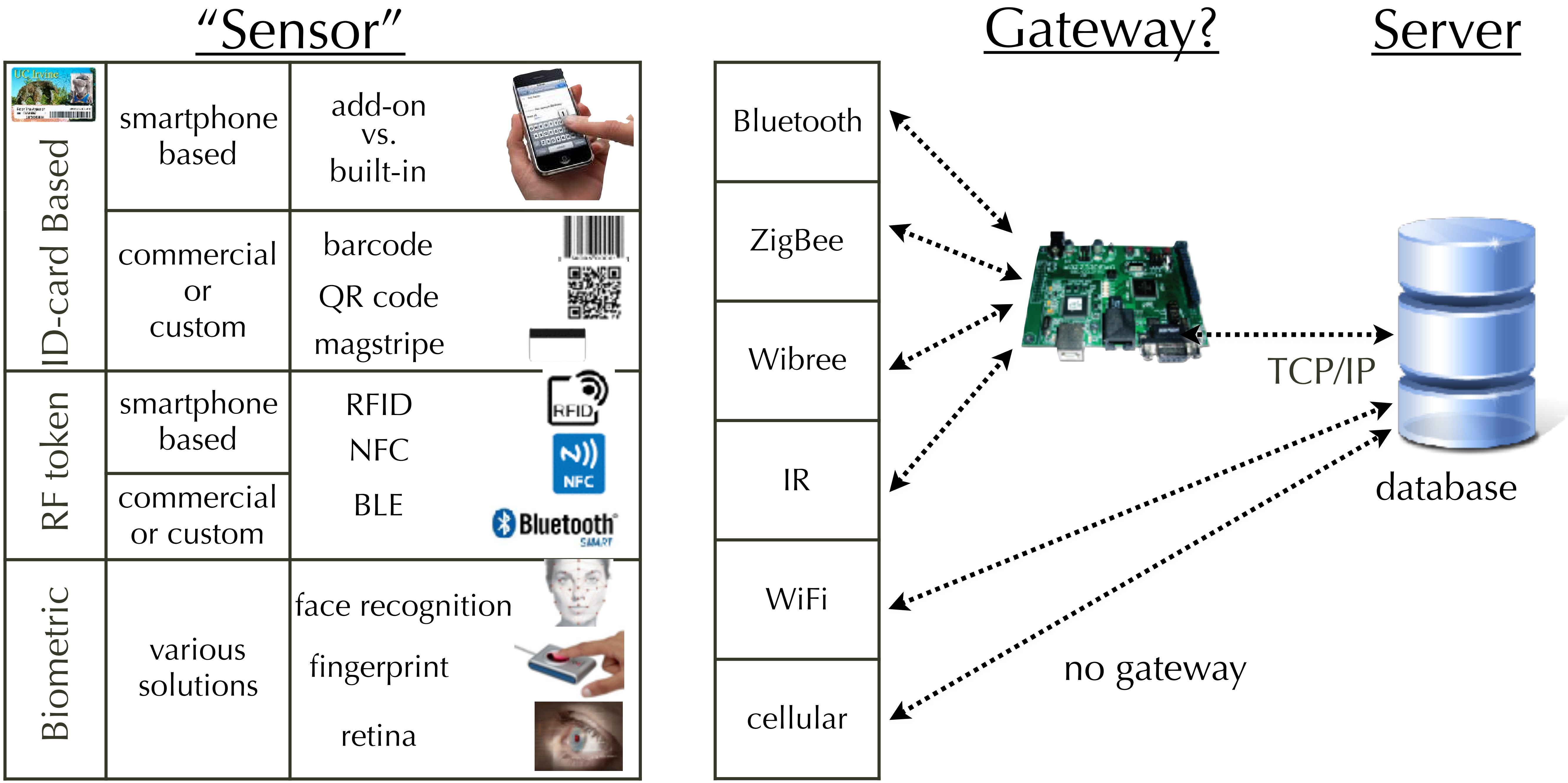
7. Manufacturability

- Printed circuit board
  - number of layers, rigidity, single side or both,
  - surface mount assembly
- Mechanical parts (incl. enclosure)
  - subtractive: etching, drilling, milling, carving,
  - additive: 3D printing, SLA (stereolithography)
    => each may limit size of model or shape

8. Sustainability

- The ability for the process to continue indefinitely on its own
  - socio-ecological, environmental, economic,
- Concepts
  - Scale in space and time
  - Consumption vs resource availability
- What is the sustainability constraint on the attendance system?
  - continued use over different courses

Attendance System: high-level subsystems



“Sensor”

ID-card Based	smartphone based	add-on vs. built-in
	commercial or custom	barcode QR code magstripe
RF token	smartphone based	RFID NFC BLE
	commercial or custom	
Biometric	various solutions	face recognition fingerprint retina

Bluetooth
ZigBee
Wibree
IR
WiFi
cellular

Major Subsystems

- Sensor subsystem (essential)
  - Inputs detected tag or biometric data
  - local vs. offloaded processing to map to studentID
  - data transmission upstream
  - what else? location sensing? real-time clock?
  - administrator authentication?
- Gateway (for non-IP wireless protocols)
- Server (essential)