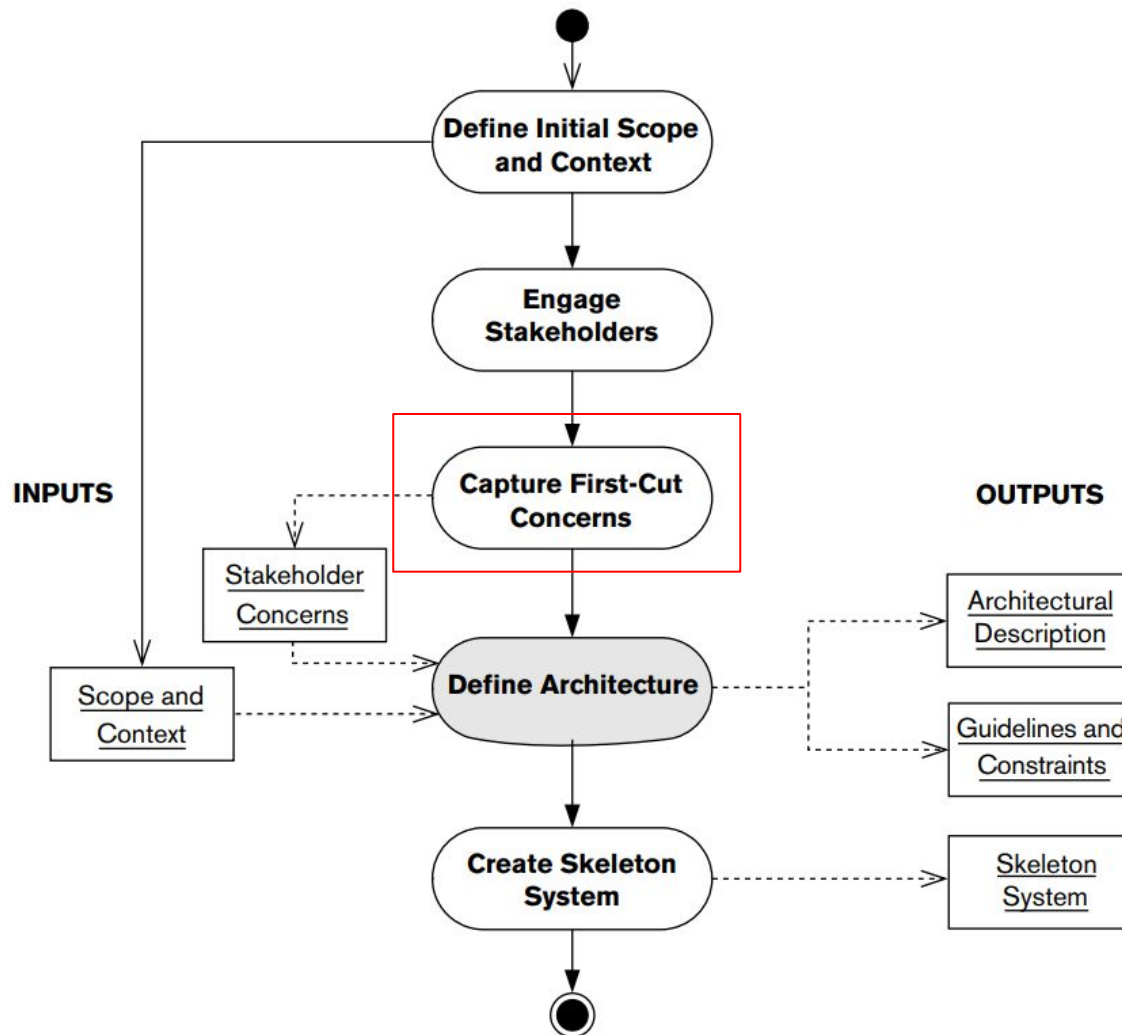# Concerns, Principles, and Decisions

CSCE 742 - Lecture 6 - 09/13/2018

The Architecture Definition Process

Today's Class

- Capturing concerns.
  - What policies, goals, and constraints must guide architecture definition?
- Establishing architectural principles.
  - How should stakeholder interests and priorities be applied over the entire architecture?
- Making and documenting decisions.
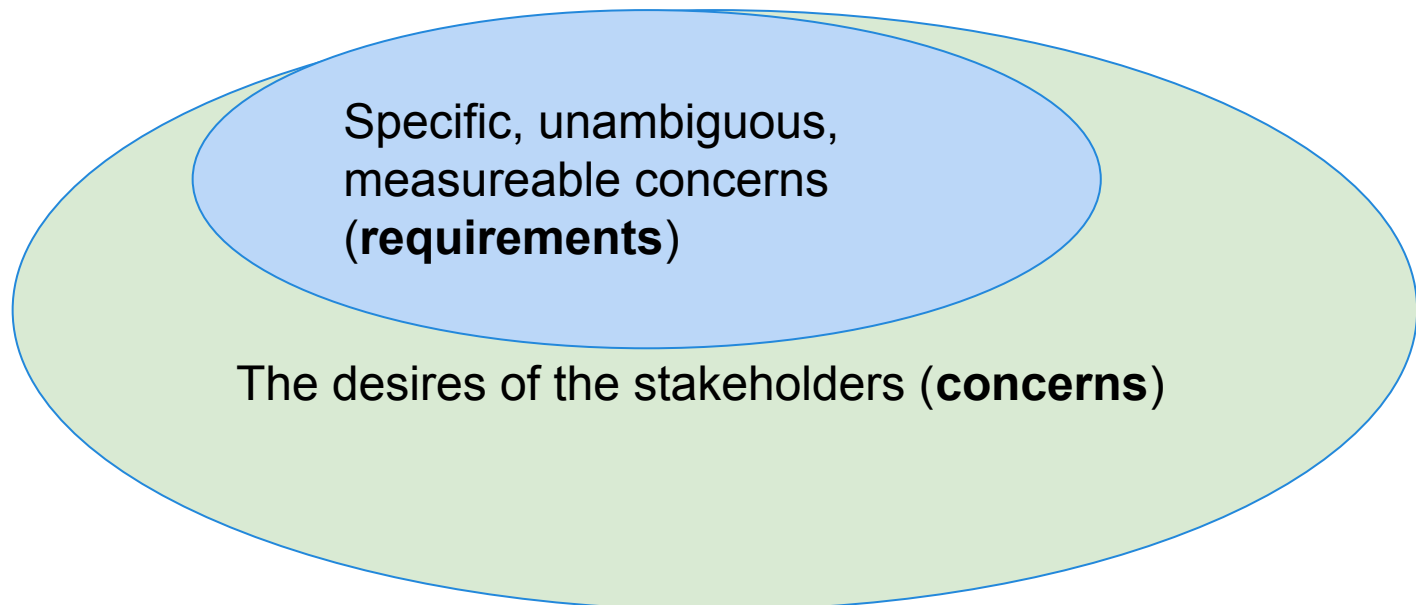 - Using principles to link concerns to decisions.

Capturing Concerns
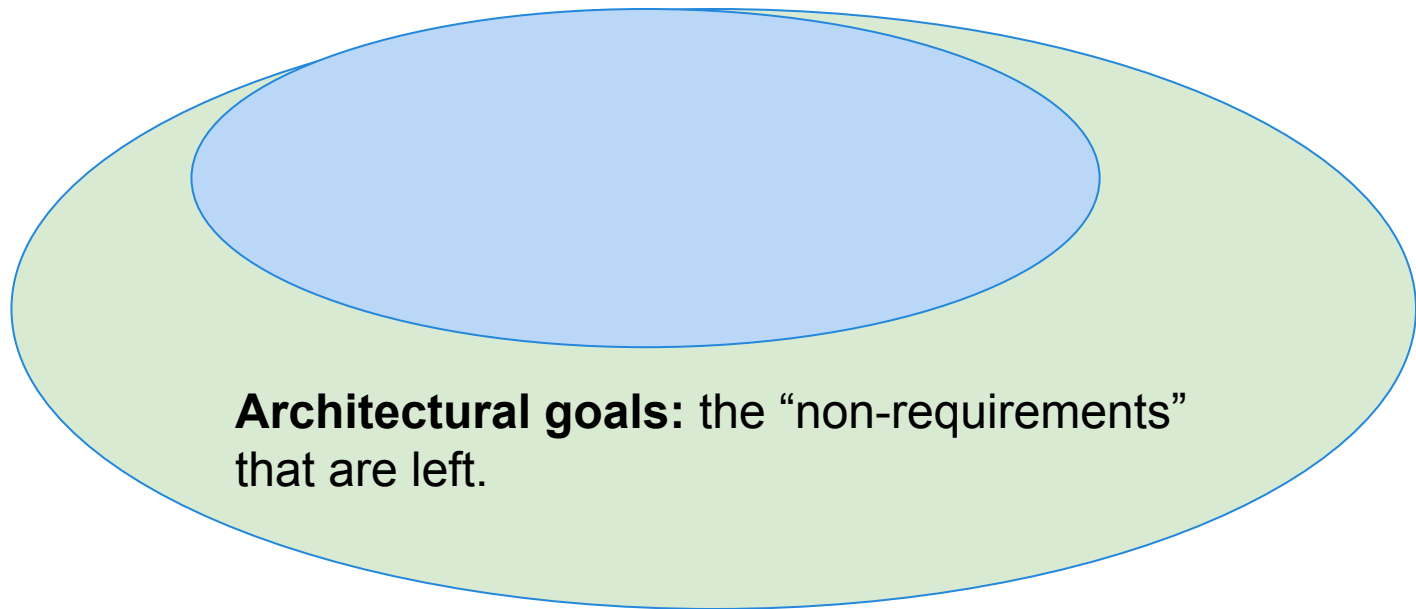
Input to the Architectural Definition

- *Business and IT strategies* set organizational priorities and establish a roadmap.
- *Goals and drivers* are the issues that drive stakeholders to initiate the project.
- *Standards and policies* mandate aspects of how the organization does business.
- *Real-world constraints* such as time, budget, and knowledge dictate project limitations.

Concerns

- A **concern** about an architecture is a requirement, an objective, a constraint, an intention, or an aspiration a stakeholder has for that architecture

Specific, unambiguous, measureable concerns (**requirements**)

The desires of the stakeholders (**concerns**)

Concerns

Architectural goals: the "non-requirements" that are left.

Online Shopping Concerns

- The values, ethos, and reputation of the retailer must be reflected in the appearance and operation of the online store and its supporting processes.
- At all times, the Web site should try to present a "human" face to the customer (even those portions of it that are fully automated).
- The online store must be easy to use by customers who have limited experience with computers and e-commerce.
- The online store must be responsive (quick to load and respond to customer actions) whether or not the customer has a fast Internet connection.
- The online store must cover all aspects of the shopping experience, including an up-to-date, browseable catalog; a secure online purchasing system; order tracking; and returns handling.

Goal Characteristics

- Often expressed using imprecise language.
- Unlikely to be quantifiable or measurable.
  - No objective criteria for judging whether or not they have been met.
- Because of strong business focus, unclear how to translate into an architectural solution
- **You cannot ignore goals and drivers.**

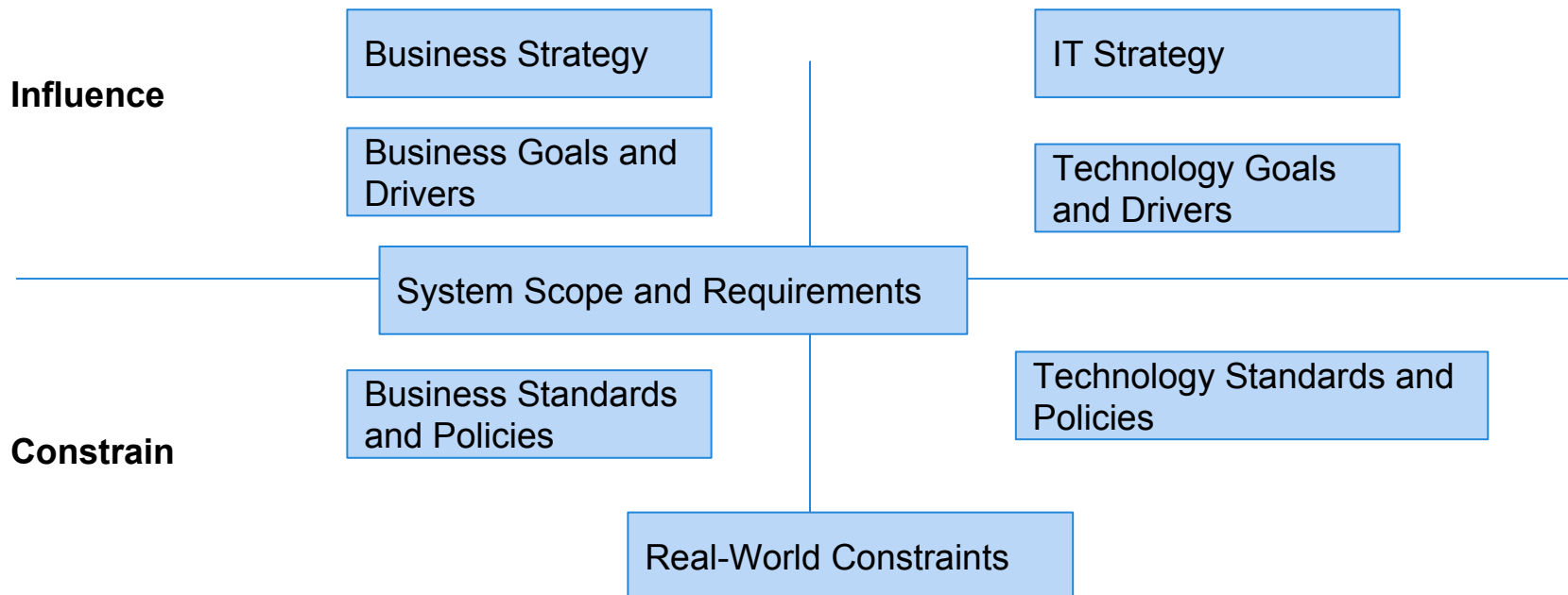# Importance of Goals

Instead Of

How to Work With Goals

- Try to turn them into requirements.
  - "System should be responsive"
    - Quantifiable response time, throughput, etc.
  - "System should be easy to use"
    - XX% of users can complete a transaction within YY time.
- Manage stakeholder expectations.
- Develop *architectural principles* that translate goals into features and qualities of the architecture.
  - "Ease of use" translates to common look and feel, common exception handling procedures, interface standards.

Concern Categories

Problem-Focused Concerns: constrain the problem the system is solving.

Solution-Focused Concerns: constrain solutions to that problem.

Influence

Business Strategy

Business Goals and Drivers

IT Strategy

Technology Goals and Drivers

System Scope and Requirements

Constrain

Business Standards and Policies

Technology Standards and Policies

Real-World Constraints

Problem-Focused Concerns

- Influence or constrain the problem the system is trying to solve.
  - May mandate a capability of the system.
  - May shape the nature and details of capabilities.
  - May constrain how the system behaves in certain circumstances.
- Address "why" and "what" questions.
- Include business strategy, business goals and drivers, scope and requirements, and business standards and policies.

Business Strategy

- Defines the direction for the business as a whole or for some part of it.
  - What goods and services does it provide?
  - Who are its customers?
  - How is this organization different from competitors?
  - How does it structure and organize itself?
- Establishes a roadmap for future business.
- Drives requirements.
  - Understanding strategy enables prioritization.
- Enables justification of architecture decisions

Business Goals and Drivers

- A goal is a specific aim the organization has.
  - "Achieve 85% of sales through online purchases"
- A driver is a force acting on the organization that requires it to behave in a particular way.
  - "Losing market share to a competitor with a better web presence"
- Work with stakeholders to translate goals and drivers into concrete requirements or architectural principles.

Business Standards and Policies

- Mandate how the organization does business or operates internally.
  - Driven by regulation, best practices, or by the organization's ethos and ways of working.
- Constrain aspects of the architecture.
  - Data retention policies, security policies.
  - Translate into architectural features such as archiving capabilities and security controls.
- Influence requirements, but aspects are often forgotten.

Solution-Focused Concerns

- Constrain the solution to the problems defined in the problem-focused concerns.
- Influences architecture by suggesting or mandating a specific approach to building the system.
- Address "how" and "with what" questions.
  - "How should it be built?"
  - "What components or technology should be used?"
- Includes IT strategy, technology goals and drivers, technology standards and policies.

IT Strategy

- Defines long-term direction for IT.
  - IT as a business unit, providing services to the rest of the organization and to customers.
- Concepts and ideas have implications on the architecture.
  - May drive technology requirements and constraints.
  - For example, systems may need to use central services or data stores.

Technology Goals and Drivers

- Goals are aims the IT department has.
  - Retailer has a technology goal to be able to scale its customer-facing systems at short notice.
- Drivers are forces acting on the project or IT department that requires particular behaviors
  - Retailer is driven by unpredictable and volatile patterns of online system usage.
- Should be translated into requirements and principles when possible.

Technology Standards and Policies

- Standards may have a technology or business focus.
  - How computers are networked versus syntax of internal messages.
  - Adopting standards may ease design and make it easier to integrate the system.
- Policies define processes that must be followed to meet stakeholder needs.
  - Often preexisting policies (security).

Technology Standards and Policies

- Open standards (ISO, IEEE, W3C) accepted by community and apply to range of HW/SW.
- Proprietary standards imposed by client or dominating companies.
- Organizational standards dictate certain HW/SW or infrastructure components.
- Ad-hoc "best practices"
- Must ensure compliance can be tested.

Other Constraints

- Web shop may use a third party service to validate credit cards.
  - Could impose a wait time when using your system.
  - Could cause customers to accidentally submit an order twice or to give up.
  - Need to ensure your site can work around this delay and prevent issues.
    - Perform authorization in the background. Send e-mail once payment confirmed.
- "Customers should not be left waiting while slow background processes complete"

Other Constraints

- Technical constraints
  - Limits in functionality, scale, security.
- Time
  - Deadlines constrain scope of solution, testing cycle, deployment windows, design time.
- Cost
  - Limit ability to hire staff, tool support, hardware choice, complexity of solution.
- Skills
  - Do your staff/users know a technology, know a development approach, or have domain expertise?

Other Constraints

- Operational Constraints
  - Need to provide service at particular times.
  - Need to operate system in compliance with organizational standards.
  - Need to fit existing operational schedules (backups, network maintenance).
- Physical Constraints
  - Distance between clients and servers, time zones.
- Organizational Constraints
  - Preferred development approaches, outsourcing.

"Good" Concerns

- Should be clearly stated and lack jargon.
- Quantified and measurable (as much as possible).
- Testable in a way that demonstrates whether it has been achieved.
- Traceable back to strategy or goals and can be traced forward to architectural or design features.

Establishing Principles

Relationship Between Concepts
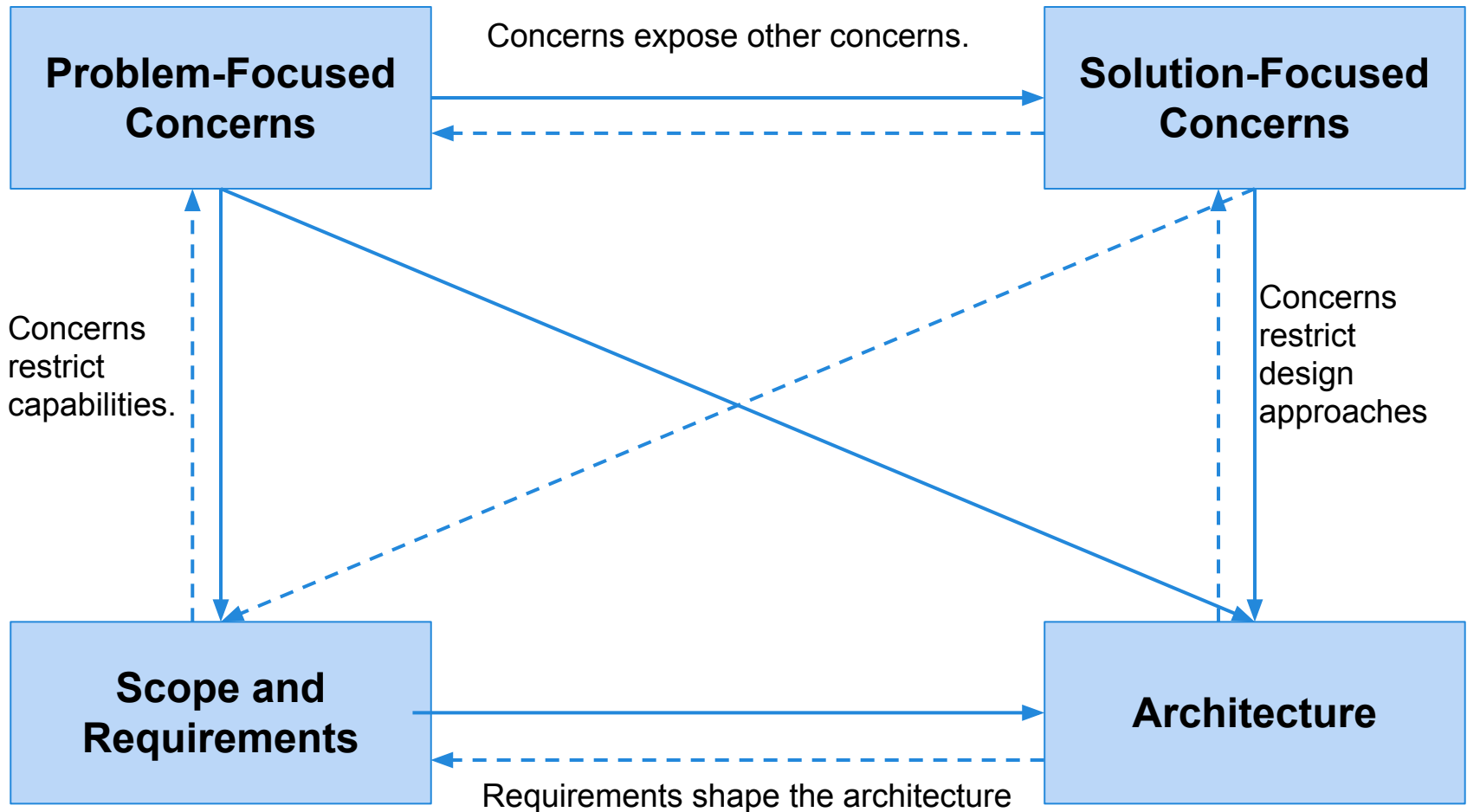
Principles

- An **architectural principle** is a fundamental statement of belief, approach, or intent that guides the definition of an architecture.
- It may refer to current circumstances or to a desired future state.
- Provide a decision-making framework for the architecture, define rules of design, and are derived from the needs and priorities of stakeholders.

Example - Weather Data

- Weather monitoring service for mobile apps.
  - Data captured around the world.
  - Published over XML and HTTP.
  - Architecture must effectively manage and distribute captured data.
- Several solutions that meet the requirements
  - A: Collect data locally and transfer to a central managed database. Simple and easy, but central point of failure and limited scalability.
  - B: Copy data to three locations, direct requests to closest.
  - C: Three locations, but each only hosts local data. Requests routed to location with the requested data.

Example - Weather Data

- Each option meets functional requirements.
  - Each has advantages and disadvantages in flexibility, performance, scalability, time, cost.
  - Each has important implications for users in timeliness, consistency, and completeness of data.
- Which is correct? Talk to the stakeholders.
  - In this case, leads to a principle:
    - **"Data required for processing a request should be held as close to that user as possible, even if this means it has to be replicated or redistributed."**
  - Rules our A and C, argues for B.

Example - Customer Care

- Retailer wants new customer care center.
- Hardware-sizing exercise reveals that cost is way over budget.
  - High availability attained via clustering, hardware replication, online backup.
- How can we reduce cost?
  - Talk to the stakeholders.
  - Reveals that different parts of the system are of varying importance to the business.
  - This can be taken into account in planning.

Example - Customer Care

- **Principle:** "While availability of the customer care system is of paramount importance, parts of the system that support customer-facing staff should be prioritized over parts such as workflow monitoring and management reporting. If tradeoffs need to be made, the parts that support the customer-facing workflow should always take precedence."
- Use high-availability technology in the parts of the system supporting customer interactions. Reduce use of that technology in the rest of the system.

"Good" Principles

- Constructive
  - Highlights issues, drives decisions, and establishes the right architectural framework.
- Reasoned
  - Motivated by business drivers and goals.
- Well Articulated
  - Can be understood by all stakeholders and is not open to misinterpretation.
- Testable
  - Must be possible to determine objectively whether they are being adhered to.
- Significant
  - Can the opposite meaning ever be applied?

Fixing a Bad Principle

- "The online store must be easy to use by customers who have limited experience with computers and e-commerce"
    - Significance test: "The store must be hard to use" will never be true, so this fails significance test.
    - Ease of use is still clearly important.
- Break down into:
    - "We should minimize the amount of data that is collected from customers during early interactions. This minimizes data protection concerns and avoids slowing down browsing and product selection. "

Fixing a Bad Principle

- "We should minimize the amount of data that is collected from customers during early interactions. This minimizes data protection concerns and avoids slowing down browsing and product selection."
  - More reasoned (rationale), constructive (highlights a specific need), and signficant (a sign-up that requires more data may be necessary).
  - Still not testable. Correct by setting a target (less data than competitors' sites)

Defining Principles

- Often seem obvious, but can cause huge arguments.
    - Get differences out in the open and resolve them early in the process.
    - State clear principles to stakeholders, argue with them, and revise until consensus is reached.
- Do not make principles too specific.
    - Should be overarching, capture spirit and motivation.
- Do not write too many principles.
    - Need to be memorable and to the point.

Making and Documenting Decisions

Decisions

- Very important to get decisions right. May be impossible to fix once code is written.
- Decisions should be made explicit (documented and discussed).
  - Engages stakeholders.
  - Ensures developers understand the architecture.
  - Provides traceability back to business drivers and goals. Gives visibility into implications of decisions.

Architectural Significance

- Does this decision answer a "what", "how", or "with what" question?
  - "What": Map out functional components of the architecture, data stores, concurrency, deployment platforms, operational tools, and element relations.
  - "How": Drive how elements are constructed.
    - I.e., which database schema to use.
  - "With what": What software and hardware will be used to build the decision?

Example - Event Booking

- Event booking service offers online ticket sales. There is concern that card authorization could be slow.
  - "What" decision: include a service to collect payment after the order is accepted, and issue a notification if there is an issue.
  - "How" decision: credit card payments will be authorized using the service operated by Chase.
  - "With What" decision: service call to Chase using their standard API over HTTPS.

Linking Concerns and Decisions

- Principles provide traceability for architectural decisions by associating rationale and its implications.
  - Use business drivers to develop business principles.
    - Principles that focus on "why" questions, with drivers as rationale.
  - Use business principles to develop technology principles.
    - Focus on "how" and "with what" questions, with business principles as rationale.
  - Use technology principles as rationale for developing architectural decisions.

Example - Online Shopping

Goal: To interact with customers in a "joined-up" manner, making it easy to browse our entire portfolio.

Business Principles:
1. All customers will gain access to any information, service, or catalog via a single point of entry
2. Customer data items (name, address, e-mail address) should be replicated across all sites.

Rationale: G1

Technology Principles:
1. Each item of customer data will be held only once, updated in real time, and used as the authoritative source of the truth.
2. Any system that requires access to a core data item will retrieve it from the central data store at the time of use.
3. Noncore data (i.e., data that applies only to a specific service or catalog) will be managed by the system to which it pertains.

Rationale: BP1, BP2

Decisions:
1. A central consolidated data store will be used that manages all core customer data
2. A messaging framework will be used to synchronously retrieve or update all core data

Rationale: TP1, TP2, TP3

Activity - Airport Parking

- You will develop a new automated parking system at the CAE airport.

- In this new system:
  - A user can insert their card into the card reader at the ramp entrance.
  - This will record the time they entered airport parking.
  - They then can use the same credit or debit card to pay at an exit lane.
  - The system should be fully automated.
  - The system should also support ticketed parking
    - User receives a ticket and pays either by credit card or cash on exiting.

Activity - Airport Parking

The system needs to interact with a number of entities and systems, including:

- Customers parking in the ramp
- Airport police and emergency responders
- Ramp managers
- External systems for validating credit card details and submitting payments
- The airport's accounting system
- External physical gate systems with basic controllers (raise / lower)
- External physical systems for signage
- An existing personnel system for staffing exit kiosks

Activity - Airport Parking

1. Who are the stakeholders?
    a. What questions should you ask them?
    b. What information is missing?
2. What are their architectural concerns?

Activity - Airport Parking

1. Who are the stakeholders?

Customers, Police, Managers, Accountants, Personnel Managers. Anyone else?

2. What are their architectural concerns?

Customer: Types of credit cards supported, ease of use, availability of service, speed of service

Airport Police: Predictability of alarms, speed of alarms, information captured in alerts

Could these concerns contradict?

Key Points

- A wide range of factors shape architecture.
  - Business and IT strategy, goals and drivers, scope and requirements, standards and policies, and real-world constraints.
- We call these **concerns**
  - Problem-focused concerns guide the problem that the system is trying to solve.
  - Solution-focused concerns guide the possible solutions to that problem.

Key Points

- Architectural **principles** are statements of belief, approach, or intent that guide architecture definition.
- These principles provide a framework for the architecture and link concerns to the final **decisions**.

Next Time

- Identifying and Using Scenarios
  - Sources:
    - Rozanski & Woods: ch. 10
    - Bass, Clements, and Kazman: ch. 4, pg. 79-86, 131-134, 147-150, 175-176

- Homework:
  - Project 1, due 9/18
  - Assignment 1, due 9/25
    - Questions?