CHALMERS |  UNIVERSITY OF GOTHENBURG

DIT045/DAT355
Requirements and User Experience

Creativity, Scoping, and Modeling (1)

Jennifer Horkoff
jenho@chalmers.se

Agenda

Requirements Creativity

Scoping

Modeling Part 1

- Context Diagram
- Goal Model Part 1

Course Representatives

- Anyone is super welcome to talk to me in person or via email with issues about the course
  - Sometimes it's too late to fix something this year, but I can fix it next year
- You can also talk to the course representatives
- I need volunteers!
  - 2-3 from Chalmers
  - 2-3 from GU
  - Email jenho@chalmers.se I will pick the first 2-3
- We'll meet mid-way through the course (end of November)

What should the system do?

- For now: Requirements are things people want the system to do
- How do we know what the system should do?
  - Ask people (good, we'll get to that later)
  - Come up with ideas on your own (can be good)
  - Come up with ideas with other people (users of the system)
- Want your ideas to be useful – solve some problem
- Want ideas to be novel – not to create a system like many others

Creativity for RE

- Stakeholders are limited by what they perceive to be possible and influenced by their experiences
  - I want the (current) system to do some extra functionality
  - The extra functionality is probably something they've already seen somewhere else
- Creative RE is about discovering requirements stakeholders were not aware of
- Creative requirements are those that are both novel and useful
- Innovation is the implementation of creativity

Creativity for RE

- The idea of studying creativity in RE is that there are methods to help average people who are not particularly creative come up with creative ideas
  - You don’t have to be a creative genius to be creative

Example: Parking Lot App

- Imagine you (or your students) must develop the requirements for a parking lot app, that lets you pay for parking using and app on your phone
- Some basic (but useful) requirements:
  - App should work on all phone types
  - App should accept all payments types (credit, paypal, contactless)
  - App should be able to email receipt
  - Register with facebook/google
  - Make it impossible to cheat app, take spot without paying

Example: Parking Lot App

- Some (more) creative requirements:
  - App should alert you when your time is almost up, allow top-ups
  - App should include a parking space finder
  - Promote app in garages
  - Should be able to “pay it forward” with parking credit

Example Brainstorming

- Brainstorming is the most famous and basic creativity activity
- Best to do this first
  - Get the obvious ideas out of the way

Example for Brainstorming

Museums and galleries want to have better outreach to potential “users”. Given COVID, the attendance rate is down, but even in the future, it is good to make their collections more accessible, e.g., for people who live far away. They want to share their collections online in a website or app. They want to educate people about history and art (or whatever the gallery is about), but they also want to make money in a way which is comparable to how they make money currently. They would like to attract new people!

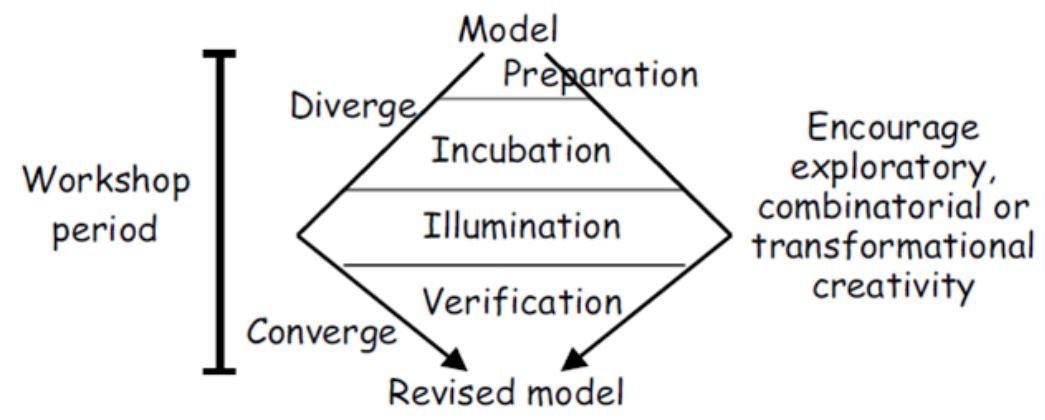
Ideas for the website/app: what should it do/not do?

Brainstorming for Virtual Museum/Gallery

- Open some tool for virtual post-its…

Creativity Theories

- Divergent and Convergent Creativity (Creative Problem Solving)
- Preparation, incubation, illumination, verification (Poincaré)
- Exploratory, combinatorial, or transformational creativity (Boden)
- RE Creativity Workshop conducted at City University London (Maiden et al.)



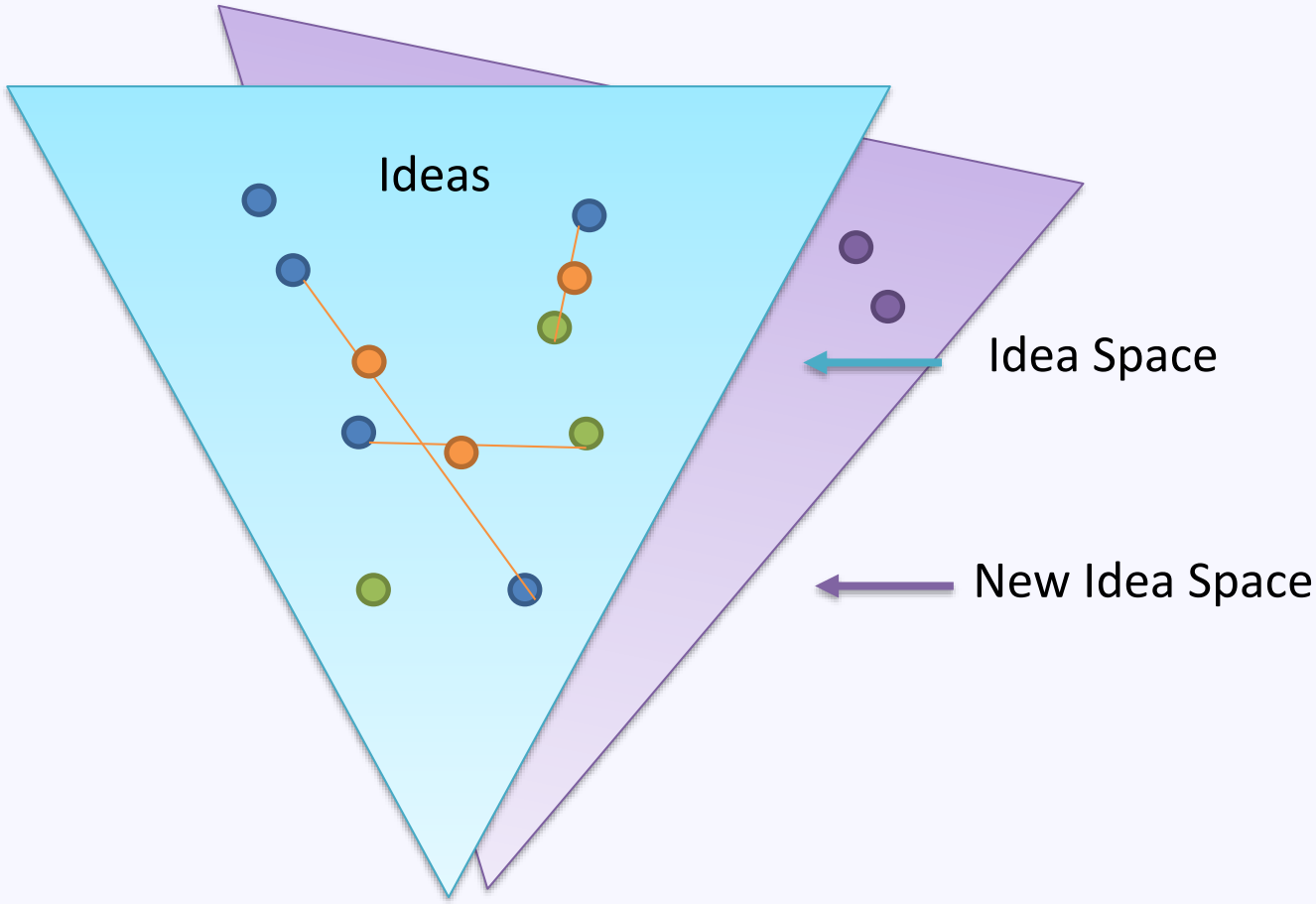
Exploratory, Combinatorial, Transformational Creativity

- (Presentation idea via N. Maiden)

Exploratory Creativity

Combinatorial Creativity

Transformational Creativity



Creativity Activities

- (Somewhat) structured techniques in order to generate ideas
- E.g., Brainstorming, Hall of Fame/Bright Sparks, Creative Search, Pairwise Comparison, Creativity Triggers, Assumption Busting, Roleplaying, ….

- Can be performed manually, or can be supported by tools
- Usually conducted on groups of people

- Long list found at:
  - http://becreative.city.ac.uk/

Creativity Triggers

Light

Try to simplify your solution, to make its structure **slighter**, more **lightweight**

Consider also to...

... remove parts of the solution to make it more **less busy, time consuming**

... revise your solution to make it looks **thinner or smaller**

Example

Apple initiated the trend of ultra-light portables by reducing the size and weight of its MacBook Air's structure, to make an extra-flat laptop.

Stimulating Stakeholders' Imagination: New Creativity Triggers for Eliciting Novel Requirements
Burnay, C., Horkoff, J. & Maiden, N.

Adaptable

Can you replace multiple products with one **adaptable** product?

Consider also to...

... add a new feature to your solution to make it able to **change**

... try to make your solution more **malleable**, more **flexible** for the user

Example

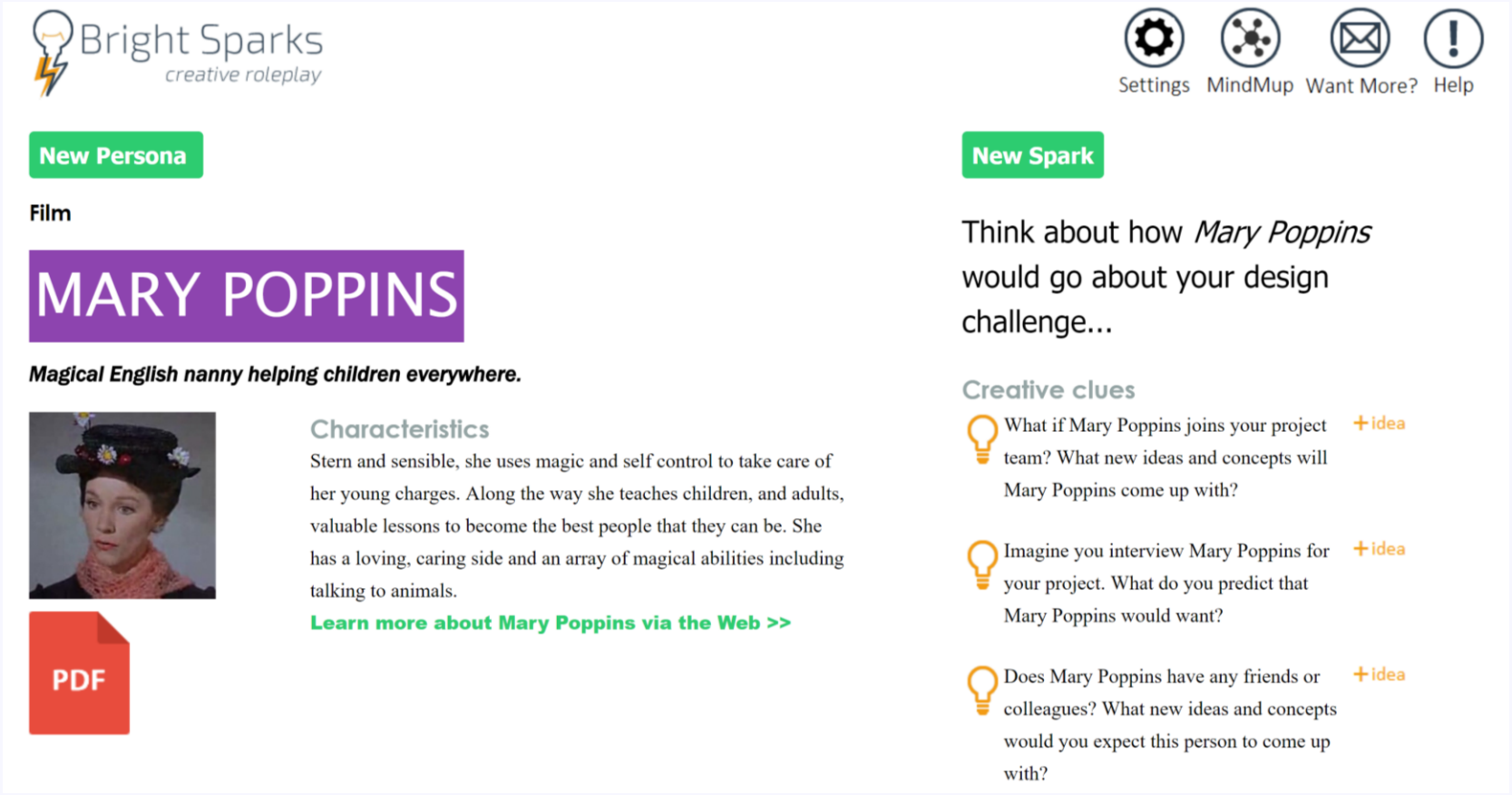
Microsoft's Surface is a tablet that turns easily into a fully functional laptop. It can adapt to multiple contexts of use, and satisfy with various user needs.

Stimulating Stakeholders' Imagination: New Creativity Triggers for Eliciting Novel Requirements
Burnay, C., Horkoff, J. & Maiden, N.

https://prezi.com/t9ggh7ymljlc/creativity-triggers/

Example Activity: BrightSparks

- Personas, but… with famous people
- http://brightsparks.city.ac.uk/



Another Exmample

- Bike share system
- Can rent pedal bikes in urban areas

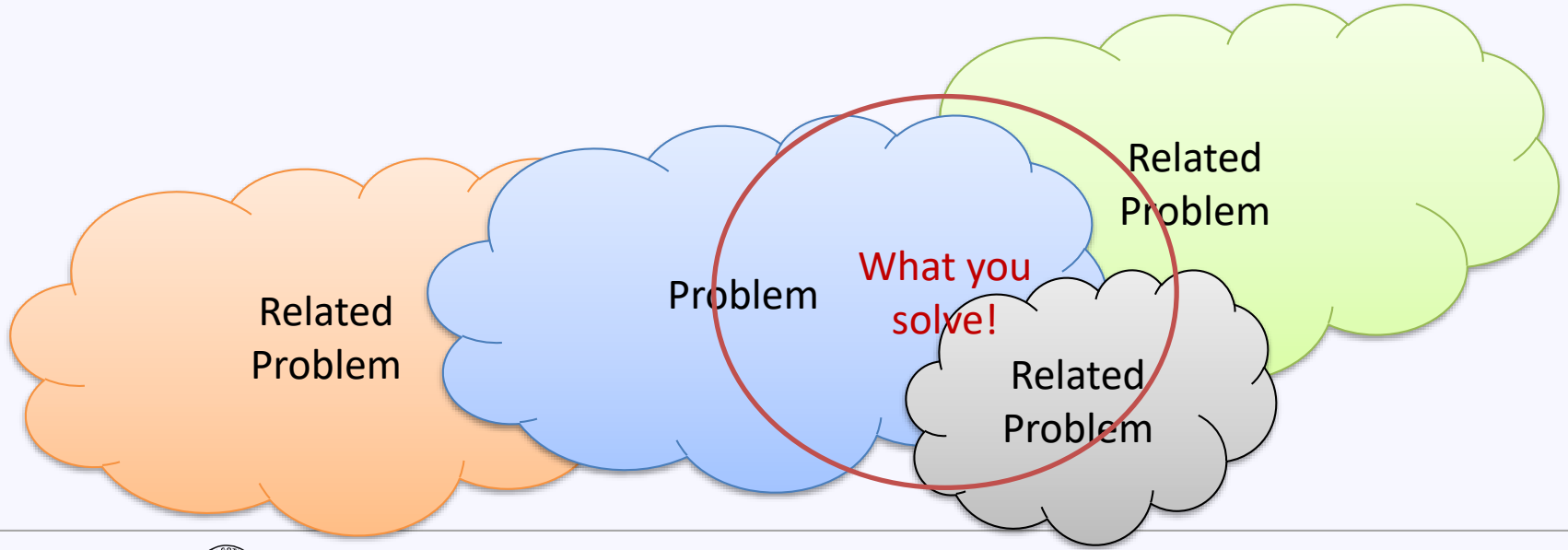
Requirements for Bike Rental

1. The bike system should make available a wide variety of bike sizes, specifically, bikes for smaller people, larger people, and kids. <see X guide>
2. The bike system should make available a wide variety of bike types, specifically, city bikes and mountain bikes.
3. The system shall allow users to rent helmets.
4. The system should provide helmets of a variety of standard sizes < see X guide>
5. The system should provide an option to rent and electric bike.
6. The bike racks should have charging stations available.
7. Bikes should be water resistant.
8. Bike tires should be suitable for driving through water.
9. It should be possible to return a bike without access to a mobile phone.

Requirements Scoping

RE Scoping

- Problems are usually boundless
  - Can always find more people with more related problems
  - Can always think of more ideas
- Need to focus on a reasonable part of the problem and set of users - set boundaries



(Again) Bike Share Example

- Bike share system
- Can rent pedal bikes in urban areas

Bike Share System Scoping

- What is the problem?
  - Want to get somewhere
  - Want to cycle, but don't want to bring a bike
  - Want to cycle, but don't want to own a bike
- Why?
  - My house is far from campus
  - Buses are too expensive
  - I'm visiting Gothenburg and don't want to travel with my bike
  - Owning a bike is too expensive, dangerous
- Solutions?
  - Move nearer to campus
  - Reduce transit costs
  - Buy a foldable bike, make oversized baggage less expensive
  - Make affordable bike buying options, or bike sharing
  - Reduce bike crime
  - ....

Another Example: The user wants

- I would like something to help with my holiday shopping. I want it to remember my good ideas for gifts that I find. It should keep a list of all the people I need gifts for, and the things they like. It should make some suggestions based on what I've bought for them in the past. It should keep track of what others have given me so I buy them gifts of roughly the same value. It should keep track of my overall budget, so I don't spent too much, and update the budget when I change my lists. Of course it should keep track of what I've bought so far. And it should be easy to use, and fun, and work on my phone. And I should be able to tell people what I've bought, but of course not the person I've bought the gift for, unless we agree, then we can share. Oh and it should be ready by December, or at least by April.
- What are you going to make? What functions? By when?

The user wants: Example Scoping

- What should the system do?
  - Keep track of presents given
  - Keep track of presents ideas
  - Keep track of present prices
  - Keep track of overall budget
    - Core functionality, must do basic
    - These functions are very related
  - Work on phone
  - NFR: Fun, easy to use
  - Constraints: ready by December
- What is not included?
  - Keep track of presents you received, with cost
  - Help to give presents of equal value
    - Not core functionality, extra feature, difficult, requires other people to use app
  - Share presents with other users (but not recipient, unless explicitly indicated)
    - Bonus functionality, interesting, but not core
  - Keep list of people to give gifts to
    - Bonus functionality, interesting, but not core

Back to the Virtual Museum/Gallery

- What is the problem?
- Why?
- What should the system do?
- Not do?

What Ideas are in/out?

Scoping

- Many ways to scope the problem
- Make sure you are solving the core problem
- Make sure the system is really the best way to solve the core problem
  - Could just buy an off-the-shelf system
  - Could not build a system at all!
- Make sure you have picked a reasonable scope to create a first version of the system
- Focus on MVP: minimal viable product
  - Add more stuff once the core is working

Requirements Modeling (part 1)

Why Modeling?

- Visual summary
- Forces you to think of specific concepts and relationships
- Forces you to think in different ways, depending on the type of model
  - Context model: think of system scope and actors
  - Goal model: actors, goals, dependencies, qualities….
  - Use Case model: actors, functions
  - Customer Journey map: actions, orders, experience

Requirements Modeling

- Many types of requirements models, we pick 4 types
- Most of UML can be used to model interactions between the world and the system
  - Class diagram: entities in the world
  - Activity diagram: user activity flows
  - Sequence diagram: user sequence of actions
  - State diagrams: states of entities in the world

- We won't do (very much) UML modeling in this course
  - You'll see in in other courses later
  - (Just remember you can also use these models for requirements)
  - Use cases are part of UML
  - Context diagrams are not (they are related to Data Flow Diagrams)
  - Goal models are not (they are more specific to RE)
  - Customer journey maps are not (RE/UX technique)

Tools

- Microsoft Visio (not free, academic license)
- PowerPoint/Word
- Draw.io (free, no templates)
- Creatively (4 free, diagrams public, templates)
- Lucidchart (free for some time, templates?)
- Many more….

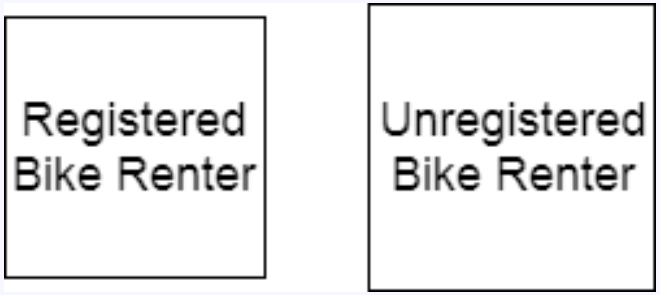
Context Diagrams (helpful for scoping!)

- Two main components:
  - Entities (actors):
    - One in the middle representing the system
    - Others around the outside representing actors providing input or receiving output from the system
  - Relationships:
    - Show the inputs and outputs to and from the system
    - Must be data or information, not actions!
- Optional helpful component:
  - Scoping tiers via concentric circles (examples later)
- Same as a level 1 data flow diagram
  - You will learn about this with databases next semester

Context Entities

- System actor

- Other actors

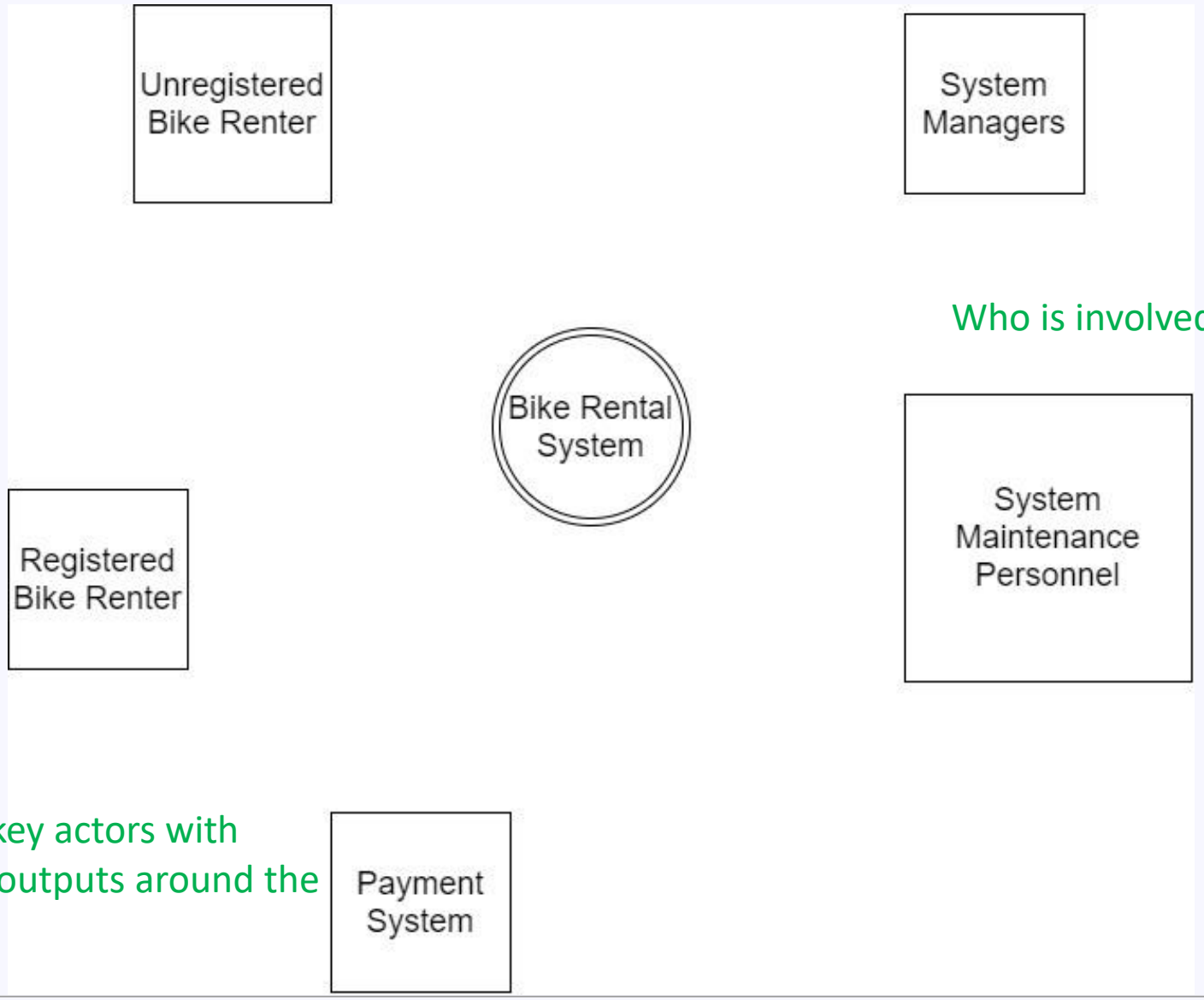


- Note: can use different shapes, just provide a legend
  - Tell us what each shape is supposed to be

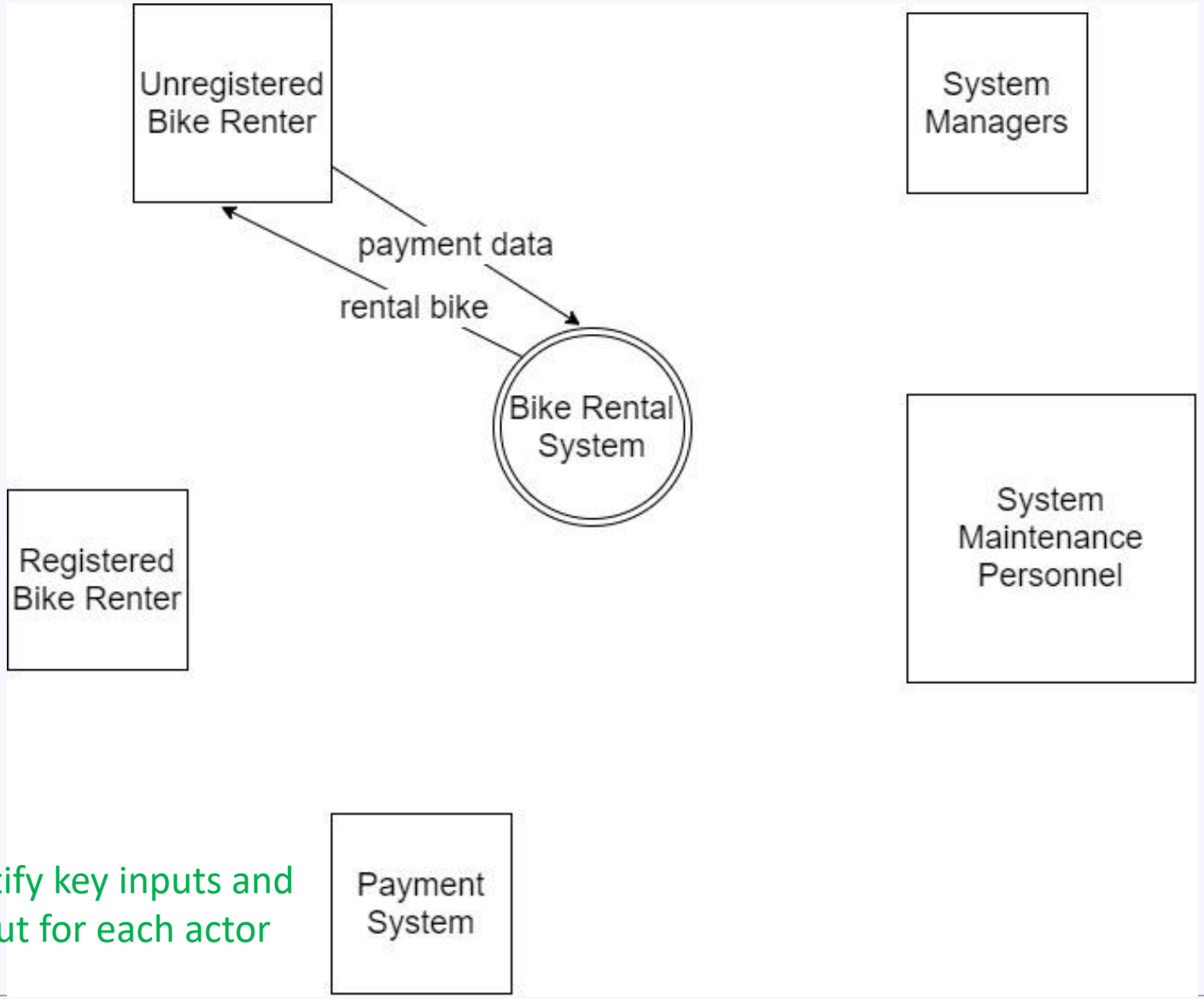
Context Example: Bike Rental

System actor in the middle

Context Example: Bike Rental

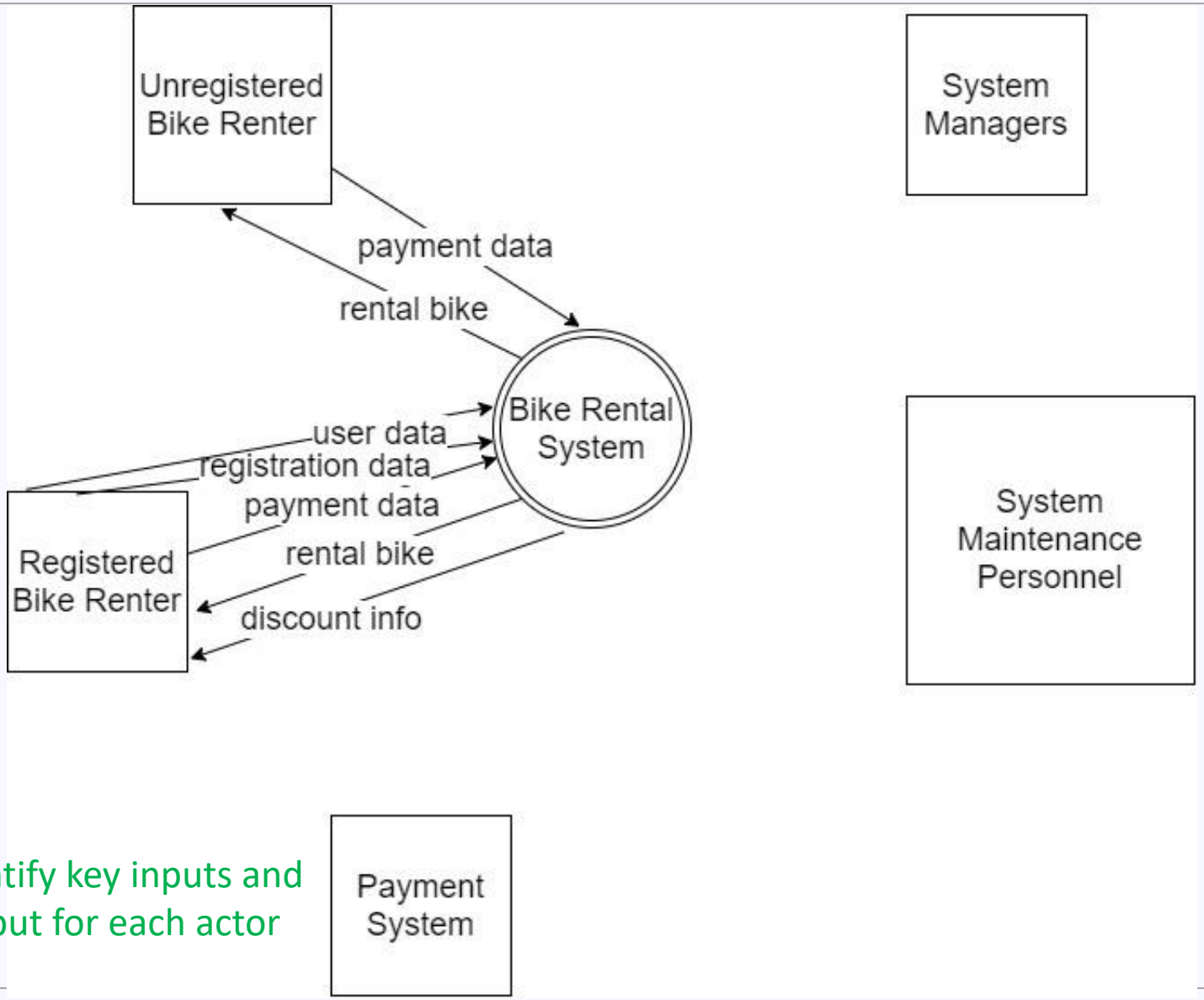


Context Example: Bike Rental



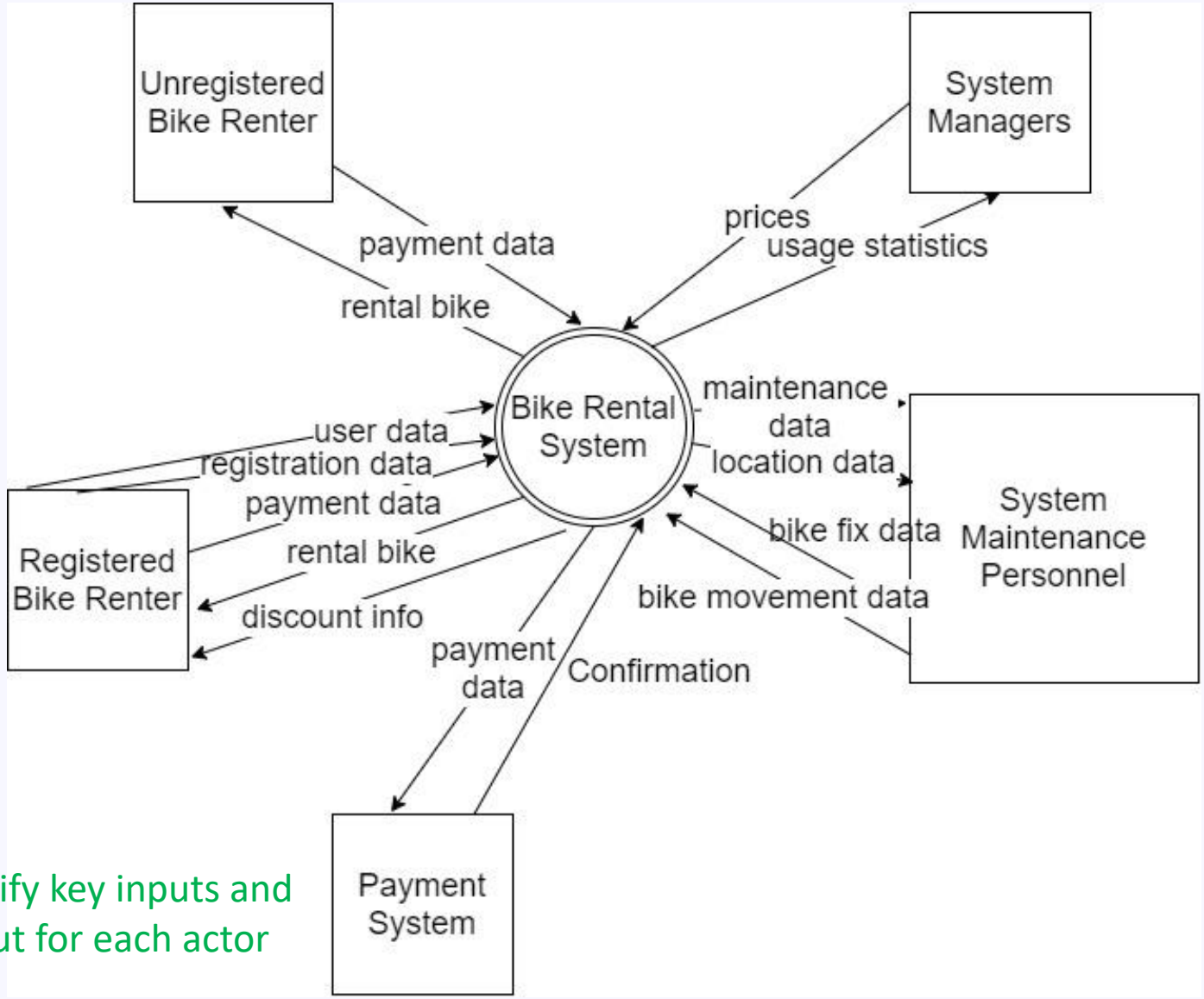
Identify key inputs and output for each actor

Context Example: Bike Rental



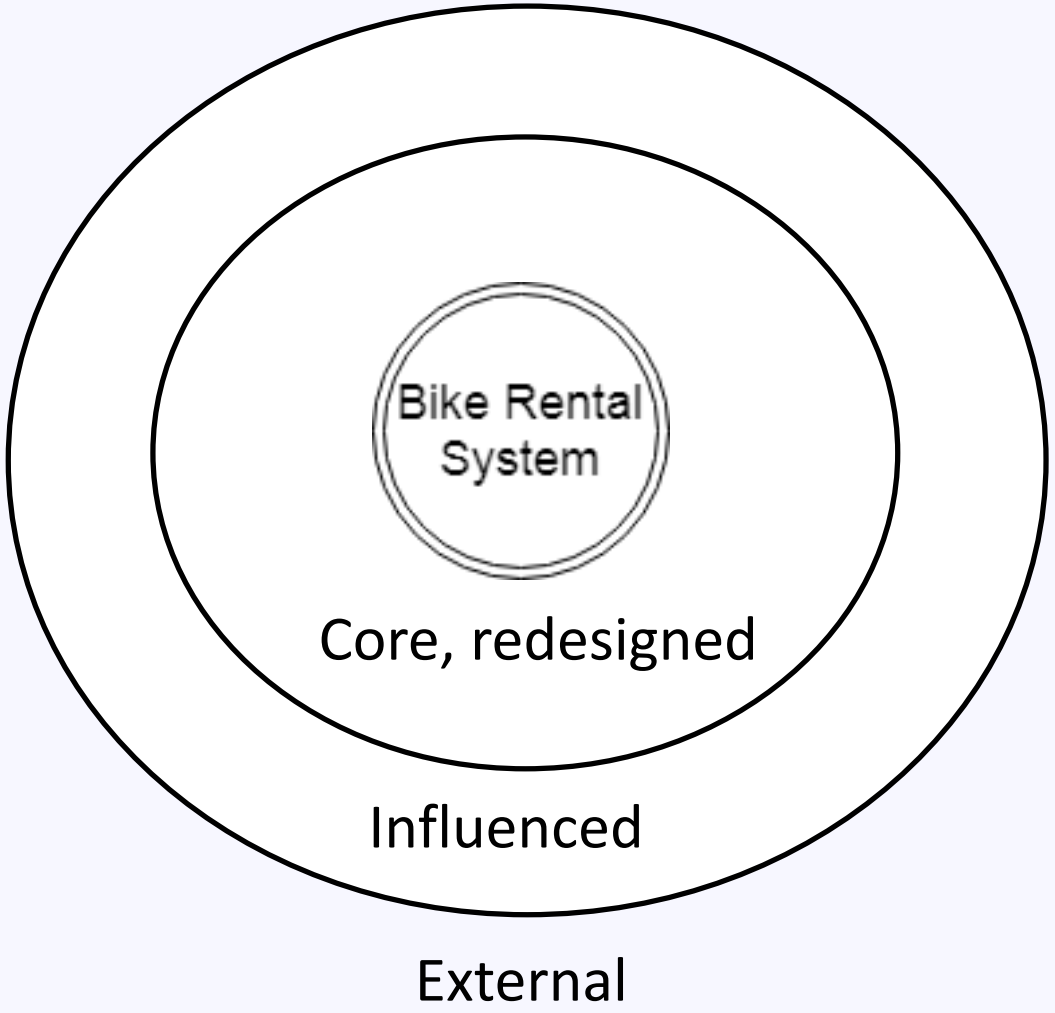
Identify key inputs and output for each actor

Context Example: Bike Rental

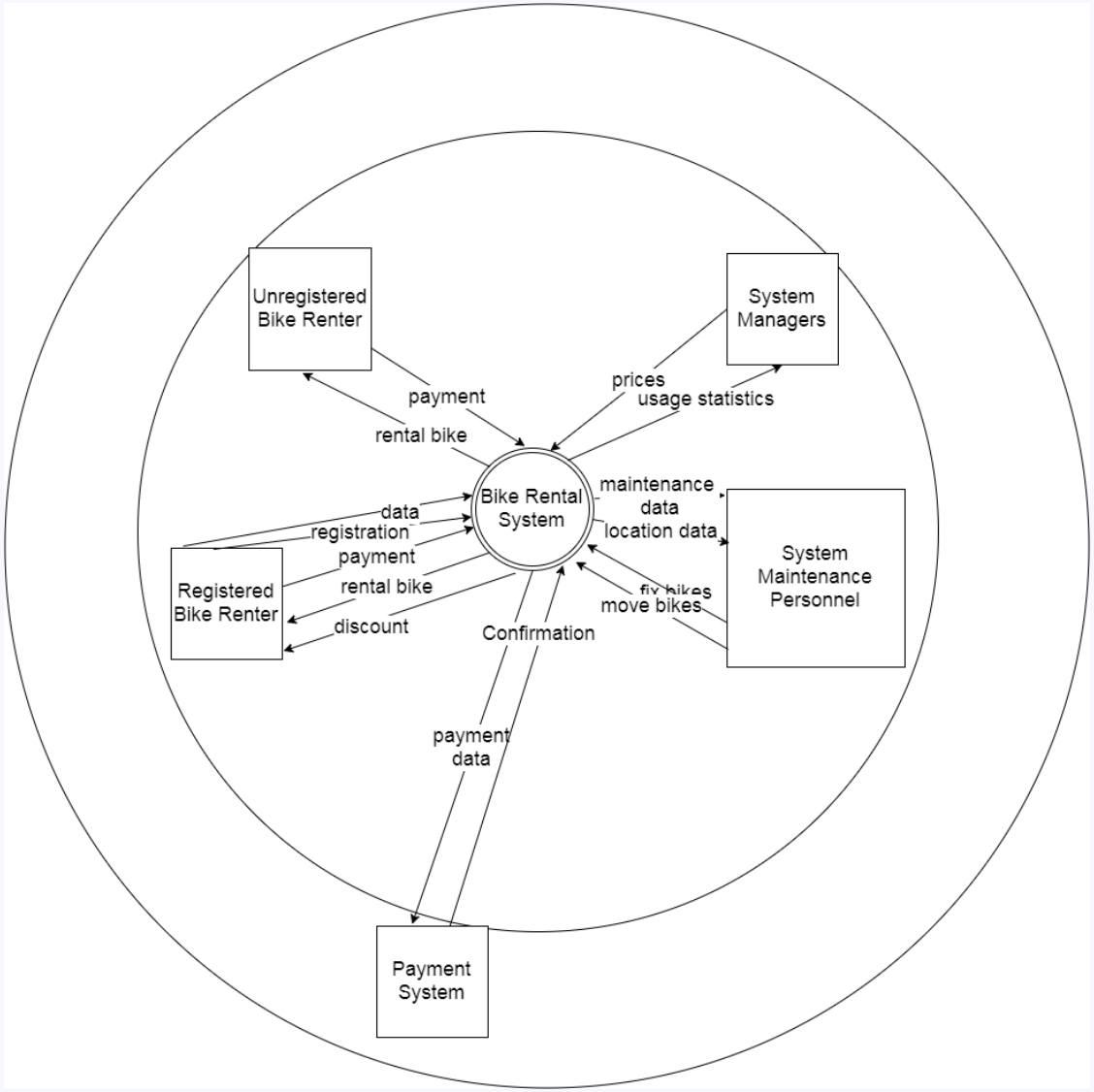


Identify key inputs and output for each actor

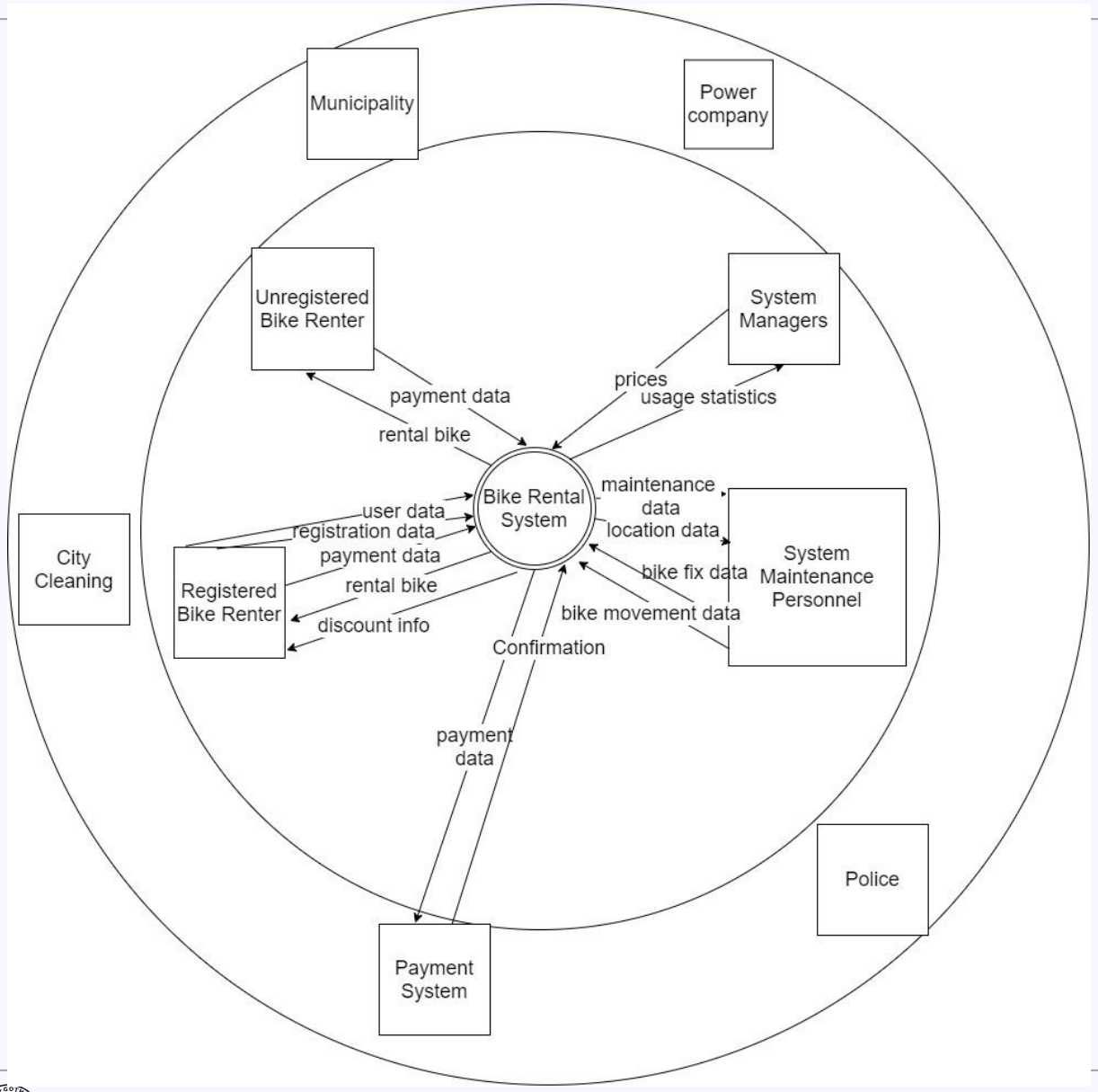
Scoping



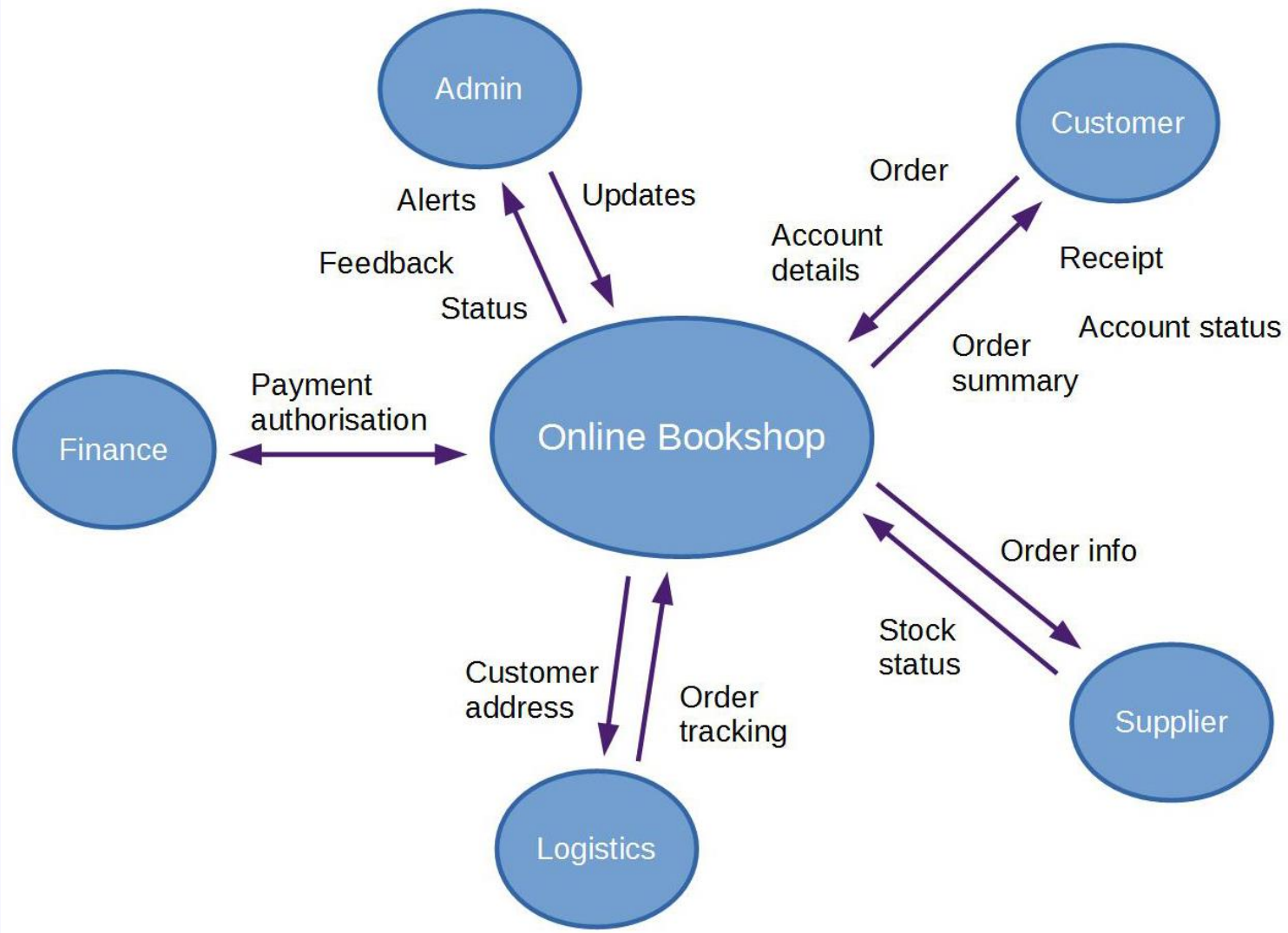
Bike Rental



Bike Rental



Context Diagram Example



https://krakenknowledge.wordpress.com/2014/10/14/design-workshop-week-3-system-context-and-component-diagrams/

For Virtual Museum/Gallery?

- Draw.io
- If time: start now, finish in video

Goal Models (part 1)

Why Goal-Orientation?

- Most systems today are socio-technical
  - E.g., online commerce, healthcare, government
- Complex web of relationships among systems and stakeholders
  - Help each other achieve what they want
- Help stakeholders understand their needs:
  - E.g., security, privacy, trust, profitability, market positioning, strategic alliances, intellectual property, …
- Understanding “why?”, not just “what?” or “how?”

Example Application: Strategic Requirements Analysis for Kids Help Phone

- Kids Help Phone (KHP) is a not-for-profit organization which provides counseling for Canadian children and youth.
- Traditionally, KHP has provided counseling via phone

- As new technology is introduced, KHP wanted to go where the kids are: the web
- How can counseling services be effectively provided on-line?
- How can the organization continue to ensure:
- Anonymity? Confidentiality? Quality of Service?

[Easterbrook, Yu, Aranda, Horkoff, Strategic Requirements Analysis for Kids Help Phone]

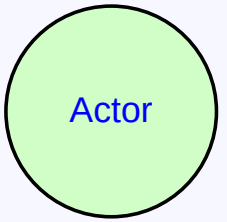
Example Application: UK Air Navigation Service Provider

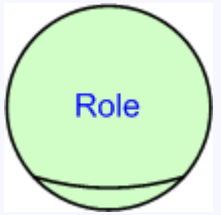
- A new design is needed for the Controlled Airspace Infringement Tool (CAIT)
  - provides air traffic controllers with timely warnings of airspace infringements
- How do we know the new requirements are safe?
- How do we know how the new requirements effect

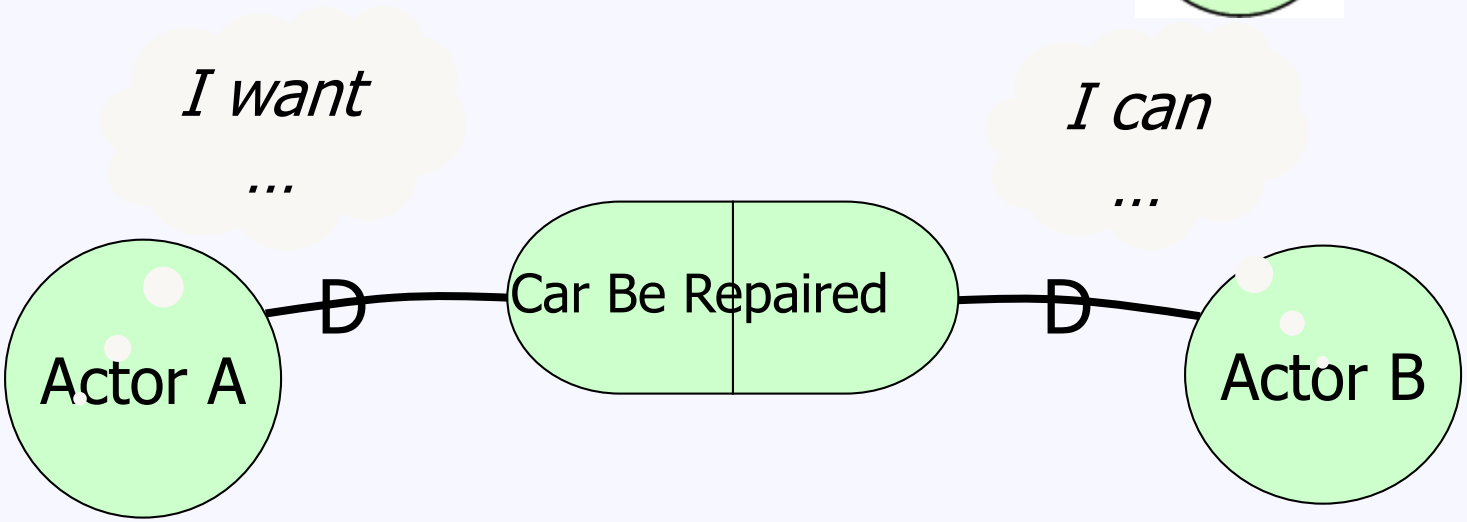
J. Lockerbie, N. A. Maiden, J. Engmann, D. Randall, S. Jones, D. Bush, REJ'11

Example Goal Modeling Framework: i* (Distributed Intentionality)

- Actors: General type
  - Agent: software or social agent:
    - E.g. Counseling System, NATS, Web Service, KHP
  - Role: collection of responsibilities
    - E.g. Client, Kid, Counselor, Pilot

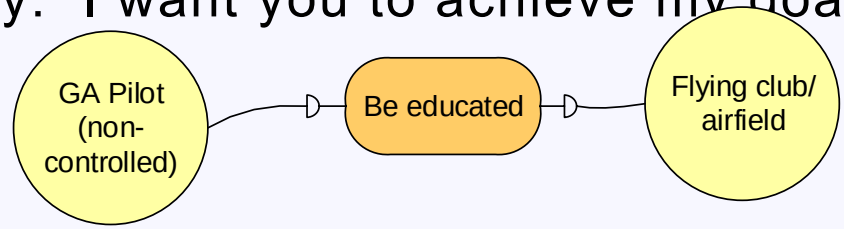






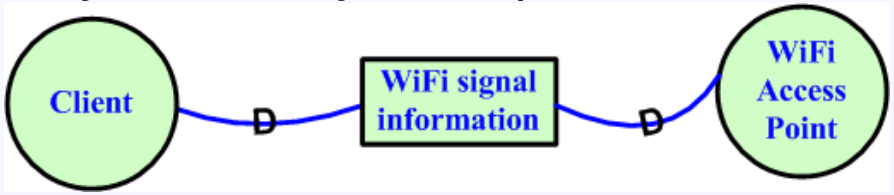
i* Strategic Dependencies (SD)

- Goal Dependency: I want you to achieve my goal, I don't care how



- Task Dependency: I want you to achieve this task, in an agreed upon way

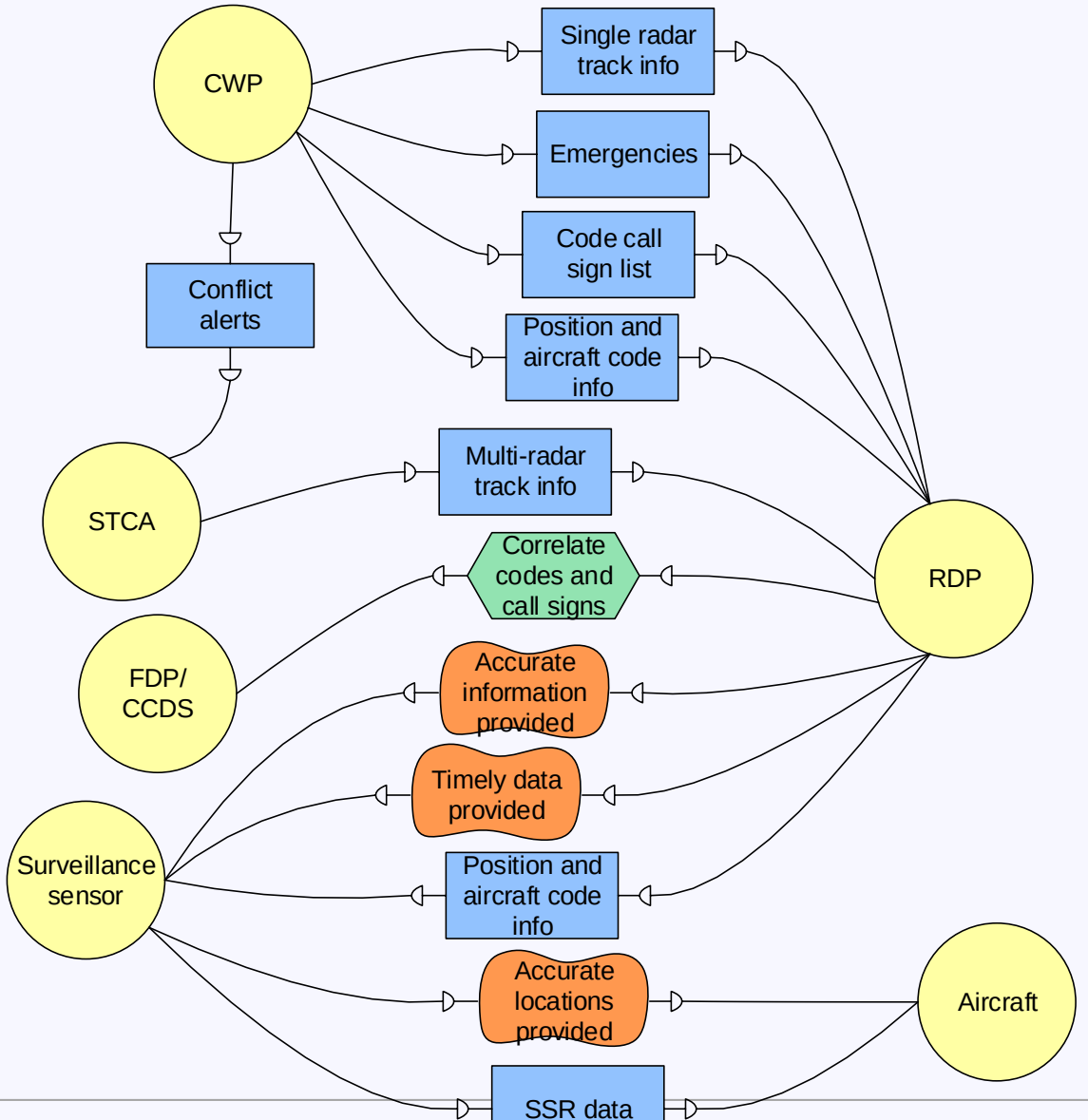
- Resource Dependency: I want you to provide this thing (entity)



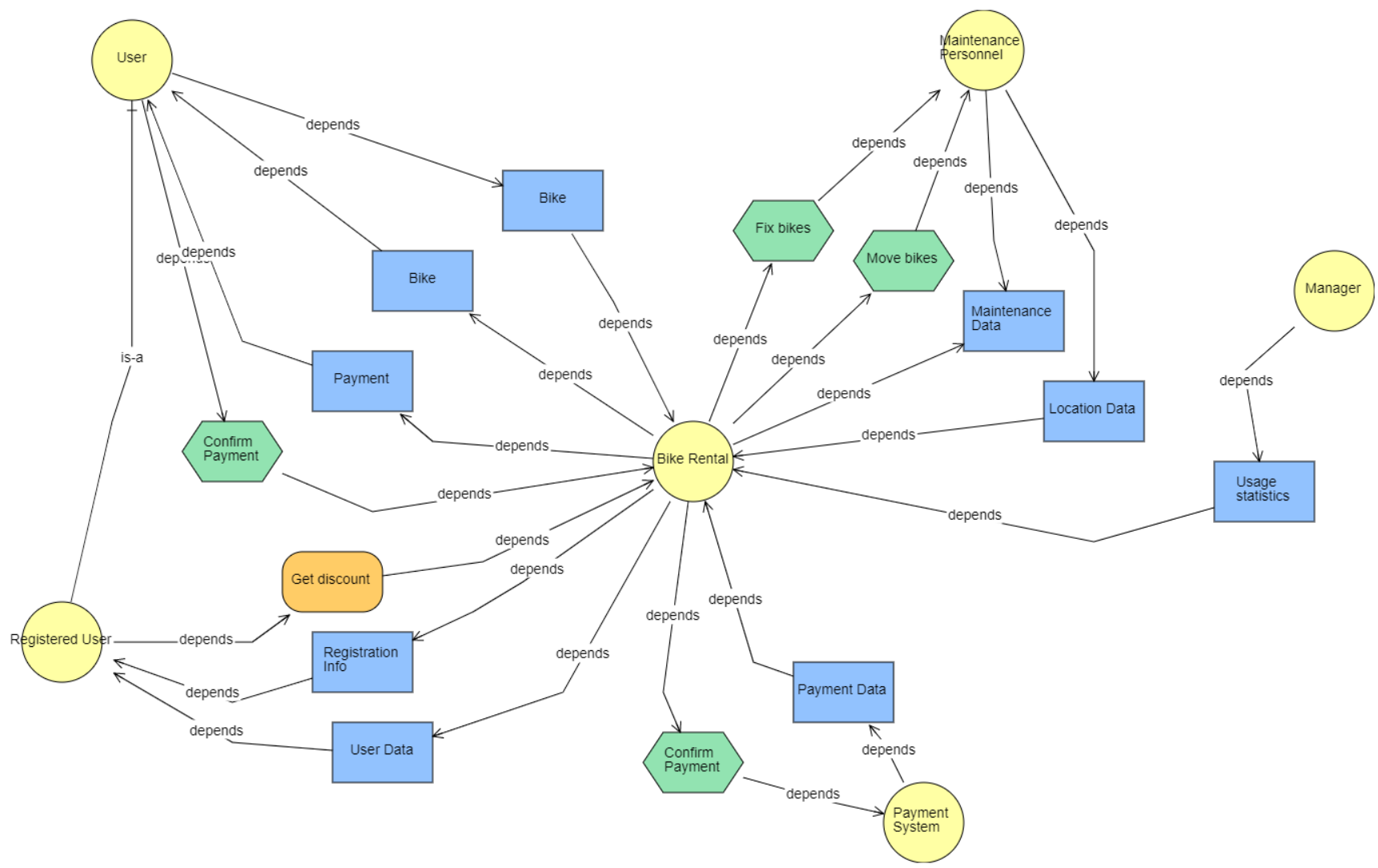
- Quality Dependency: I want you to achieve my quality

SD Example: NATS

- CWP = Controller Working Position
- STCA = Short Term Conflict Alter
- FDP/CCDS = Flight Data Processor/Code Callsign Distribution System
- RDP = Radar Data Processing



Bike Rental



Virtual Museum/Gallery SD?

- Start if time
- Finish in video
  - Uploaded in 1-2 days

Optional Readings

- Extra iStar help:
  - iStar cheatsheet
  - iStar2
  - 2016-12-istar_Supplimentary

Questions?