

Understanding Design Complexity

Rebecca Wirfs-Brock
www.wirfs-brock.com



Who I Am



Agenda

- Motivation
 - What makes software complex?
 - Agile design values
- Techniques for characterizing complexity
 - Commonality-Variability Analysis Conversations
 - Tools: Hot spot cards, whiteboards & decision tables
- Design Follow Through
 - Design strategies for managing complexity
 - Making design limitations explicit

3

What makes software complex?

- Intrinsic complexity:
 - Rich, intricate data
 - Many interconnections and relationships
 - Non-trivial algorithms, behaviors, rules
 - Special cases
 - Configurations and options
 - ...
- Long useful evolution:
 - Growth spurts
 - Complex connections
 - Technology shifts
 - Extensions, customizations, hacks, patches, adaptations
 - ...

4

Design Complexity

- Messy, irregular, volatile, unpredictable, sprawling
- or
- Orderly, predictable, relatively stable, compartmentalized?

5

How can you grow sustainable solutions that tackle complexity?

- Involve people skilled in both agile development practices *and* design *and* the problem domain
- For complex stuff, spend adequate time understanding inherent complexity.
- DO THIS INCREMENTALLY.
- Make time for exploring options
- Grow best practices to handle complexity

6

Agile Design Values

- Core values:
 - Design Simplicity
 - Communication
 - Teamwork
 - Trust
 - Satisfying stakeholder needs
- Constant learning



7

Design Principles and Best Practices

- ***Separate concerns.***
 - Don't mix application logic with data access or presentation code.
- ***Preserve the ability to grow your solution.***
 - ***Hide and organize details.***
 - Isolate changeable from stable parts.
 - Keep your implementation clean. Designs are only as good as their implementation details.
 - Develop tests along with code. Testable designs are easier to refactor.
- ***Avoid accidental complexity.***
 - Find the simplest solution that supports known variability and complexity.
 - Rework designs when the problem shifts to avoid technical debt.

8

An Example: Rental Fee Calculations

| Membership Category | Car Type | Insurance Type | Base Hourly Rate | Base Charge per Mile | Special Daily Rate Applies | Special Daily Rate | Damage Deductible |
|---------------------|-----------------|----------------|------------------|----------------------|----------------------------|--------------------|-------------------|
| Low | Small hatchback | Basic | \$3.95 | \$0.17 | No | | \$500 |
| Standard | Small hatchback | Basic | \$2.80 | \$0.17 | Yes | \$36 | \$400 |
| Low | Large hatchback | Basic | \$3.00 | \$0.18 | No | | \$500 |
| Standard | Large hatchback | Basic | \$3.00 | \$0.18 | No | \$38.50 | \$400 |
| Low | Small hatchback | Full | \$3.95 | \$0.17 | No | | \$100 |
| Standard | Small hatchback | Full | \$2.80 | \$0.17 | Yes | \$36 | \$0 |
| Low | Large hatchback | Full | \$4.95 | \$0.18 | No | | \$100 |
| Standard | Large hatchback | Full | \$3.00 | \$0.18 | Yes | \$38.50 | \$0 |

9

Aargh!!

How can you make sense out of tangled, complex requirements?



10

“The structuring tasks in design build on the deepest building blocks of human cognition: being able to distinguish what is common from what changes.”—James Coplien & Gertrud Bjørnvig, *Lean Architecture for Agile Software Development*

Find Commonalities and Variations

11

Spot Commonalities and Ask:

Do they make sense in the domain, or are they anomalies?

| Membership Category | Car Type | Insurance Type | Base Hourly Rate | Base Charge per Mile | Special Daily Rate Applies | Special Daily Rate | Damage Deductible |
|---------------------|-----------------|----------------|------------------|----------------------|----------------------------|--------------------|-------------------|
| Low | Small hatchback | Basic | \$3.95 | \$0.17 | No | | \$500 |
| Standard | Small hatchback | Basic | \$2.80 | \$0.17 | Yes | \$36 | \$400 |
| Low | Large hatchback | Basic | \$3.00 | \$0.18 | No | | \$500 |
| Standard | Large hatchback | Basic | \$3.00 | \$0.18 | No | \$38.50 | \$400 |
| Low | Small hatchback | | | | | | |
| Standard | Small hatchback | | | | | | |
| Low | Large hatchback | | | | | | |
| Standard | Large hatchback | | | | | | |

Is damage deductible amount based on insurance type & membership?

12

Meaningful Cluster Commonalities

| Membership Category | Car Type | Insurance Type | Base Hourly Rate | Base Charge per Mile | Base Hourly Rate | Special Daily Rate | Damage Deductible |
|---------------------|-----------------|----------------|------------------|----------------------|------------------|--------------------|-------------------|
| Low | Small hatchback | Basic | \$3.95 | \$0.17 | \$3.95 | | \$500 |
| Low | Large hatchback | Basic | \$4.95 | \$0.18 | \$3.95 | | \$500 |
| Low | Small hatchback | Full | \$3.95 | \$0.17 | \$4.95 | | \$100 |
| Low | Large hatchback | Full | \$4.95 | \$0.18 | \$4.95 | | \$100 |
| Standard | Small hatchback | Basic | \$2.80 | \$0.17 | \$2.80 | \$36 | \$400 |
| Standard | Large hatchback | Basic | \$3.00 | \$0.18 | \$3.00 | \$38.50 | \$400 |
| Standard | Small hatchback | Full | \$2.80 | \$0.17 | \$2.80 | \$36 | \$0 |
| Standard | Large hatchback | Full | \$3.00 | \$0.18 | \$3.00 | \$38.50 | \$0 |

13

Meaningful Cluster Commonalities

| Membership Category | Car Type | Base Hourly Rate | Base Charge per Mile | Special Daily Rate Applies | Special Daily Rate | Insurance Type | Damage Deductible |
|---------------------|-----------------|------------------|----------------------|----------------------------|--------------------|----------------|-------------------|
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Basic | \$500 |
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Full | \$100 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Basic | \$500 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Full | \$100 |
| Standard | Small hatchback | \$2.80 | \$0.17 | Yes | \$36 | Basic | \$400 |
| Standard | Large hatchback | \$3.00 | \$0.17 | Yes | \$38.50 | Basic | \$400 |
| Standard | Small hatchback | \$2.80 | \$0.18 | Yes | \$36 | Full | \$0 |
| Standard | Large hatchback | \$3.00 | \$0.18 | Yes | \$38.50 | Full | \$0 |

14

Organize Variations

Used to compute rental cost

Insurance Fees

| Membership Category | Car Type | Base Hourly Rate | Base Charge per Mile | Special Daily Rate Applies | Special Daily Rate | Insurance Type | Damage Deductible |
|---------------------|-----------------|------------------|----------------------|----------------------------|--------------------|----------------|-------------------|
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Basic | \$500 |
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Full | \$100 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Basic | \$500 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Full | \$100 |
| Standard | Small hatchback | \$2.80 | \$0.17 | Yes | \$36 | Basic | \$400 |
| Standard | Large hatchback | \$3.00 | \$0.17 | Yes | \$38.50 | Basic | \$400 |
| Standard | Small hatchback | \$2.80 | \$0.18 | Yes | \$36 | Full | \$0 |
| Standard | Large hatchback | \$3.00 | \$0.18 | Yes | \$38.50 | Full | \$0 |

15

And Repeat...

Only for Standard Members

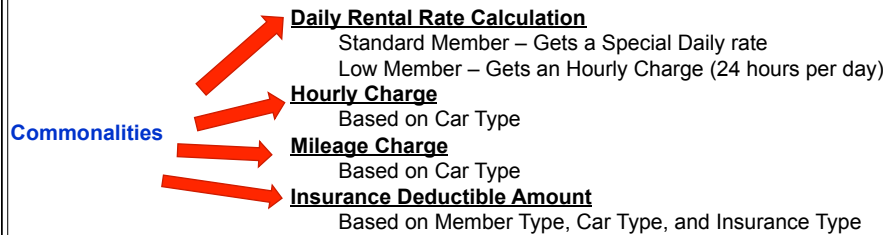
| Membership Category | Car Type | Base Hourly Rate | Base Charge per Mile | Special Daily Rate Applies | Special Daily Rate | Insurance Type | Damage Deductible |
|---------------------|-----------------|------------------|----------------------|----------------------------|--------------------|----------------|-------------------|
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Basic | \$500 |
| Low | Small hatchback | \$3.95 | \$0.17 | No | | Full | \$100 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Basic | \$500 |
| Low | Large hatchback | \$4.95 | \$0.18 | No | | Full | \$100 |
| Standard | Small hatchback | \$2.80 | \$0.17 | Yes | \$36 | Basic | \$400 |
| Standard | Small hatchback | \$2.80 | \$0.17 | Yes | \$36 | Full | \$0 |
| Standard | Large hatchback | \$3.00 | \$0.18 | Yes | \$38.50 | Basic | \$400 |
| Standard | Large hatchback | \$3.00 | \$0.18 | Yes | \$38.50 | Full | \$0 |

16

Hmm...price seems to be based on car type

Analyze Variations

- Commonality-variability analysis isn't design. It is preparation for creating an appropriate solution



17

Technique: Informally Describing
Variations on **Hot Spot** Cards

Variation name

**general description of variable
behavior**

at least two specific examples

18

Hotspot Cards: An Informal Technique to Describe Variations

- Write just enough detail so you can discriminate important differences.

Rental Rate Calculation

The cost of a car rental is based on fees assessed based on type of car, mileage, and customer membership category.

Standard member = 3 days * daily rate + 6 hrs * hourly rate + (cost per mile * miles).

Low member = 56 * hourly rate + (cost per mile * miles)

Hourly Rental Rate

A rental rate is charge per hour based on car type and membership category.

Standard member, small hatchback = \$2.80/hr
Standard member, large hatchback = \$3.00/hr
Low member, small hatchback = \$3.95/hr
Low member, large hatchback = \$4.95/hr

19

Design to support variations

1. Establish the scope of the variation—how much of the design will be considered.
2. Identify what is common and how it varies.
3. Bound the degree of variability that will be supported. Place limits on how much variation can be supported by your solution. Explain those limits.
4. Exploit commonalities in a design solution; while
5. Accommodating the variability.

20

How Much Variability to support?

- Designers spot commonalities that business people do not. Are they meaningful? What other factors should be considered (not yet illustrated by concrete examples)?
 - **Currently mileage charges are based on car type. But...**
 - Are there other ways the business would like to calculate them?
 - What happens when costs increase or vary by location? Should there be ways to calculate mileage charges based on location?
 - **Member categories now only determine daily and hourly rental and insurance charges.**
 - Are there other charges or benefits based on membership?
 - Do membership categories, charges, or benefits change often?

21

Agile Designers Favor Simplicity

- Choose the simplest, flexible solution.
- From simple to complex:
 0. Hardcoded checks—If-then-else statements to control branching. Not flexible. But it does support variation.
 1. Parameters/data that drive behavior
 2. Delegation to pluggable objects (composition)
 3. Classification and inheritance
 4. Define a “little language” or DSL that you interpret

22

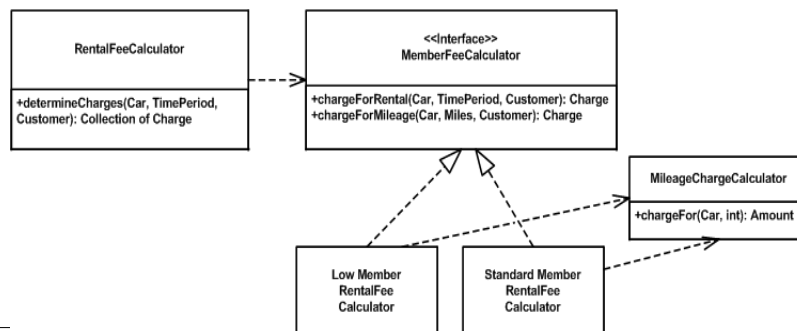
Find The Simplest Flexible Solution

- **A simple solution:** hard code the calculation of mileage fees into a single method.
- **A simple flexible solution:** invoke a MileageFeeCalulator, passing in car type and mileage as arguments. Retrieve fee rates from an external database.

23

A Flexible Solution: Encapsulate What Varies

- Calculations vary according to membership:
 - Standard member price = (number of days * daily rental price) + (number of hours * hourly rental price) + (miles driven * standard member mileage rate based on car type)
 - Low member price = (number of hours * hourly rental price) + (miles driven * low member mileage rate)



24

Designs Always Have Limits

- What if we want to waive fees for certain members or calculate mileage based on rental location? **Our current solution doesn't handle this!!**

- State design limits:

Daily mileage calculations based on car type.

The price per mile is read from a database.

Mileage rate table is cached.
So the cache will need to be flushed
& reloaded before new rates apply.



25

Characterize a New Variation

How would these requirements change your ideas?



- We have a new kind of member: Platinum
 - Mileage fees are waived for platinum members;
 - Platinum members get preferred rental rates after renting more than 3 times in a calendar year; and
 - Platinum members get a free rental day after every 4th car rental.

26

Analyze Complexity



27

Making a Design Flex



Flexible designs incorporate mechanisms that enable them to be easily changed to handle planned for adaptations.

28

Build Places to Support Variation



- Isolate changeable responsibilities;
- Design places where behavior can be “tuned” or replaced;
- Define stable interfaces;
- Introduce factories or service locators to provide varying help.

29

Design Placeholders



- Explicit spots that contain and encapsulate.
- Placeholders encapsulate behavior and information that you expect to change.
- Grow their responsibilities incrementally.

30

Hooks Allow Behaviors to Adjust

- A “hook” is a place in a design that is meant to be adapted. Add hooks to:
 - Enable or disable application features based on configuration data;
 - Configure how a feature works.



31

Concentrate Variable Information into Information Holders

- Often, many values control behavior:
 - Time out duration;
 - Language for user messages;
 - Availability of a feature;
 - Configurable display options...



Best Practice: Group related configuration information together.

32

The best designers never give up, they
just know when to call it a day!

