WRITING A PROGRAM

CS 130

Creative Software Architectures for Collaborative Projects

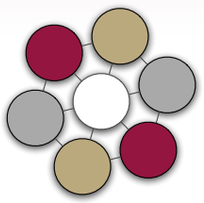
Prof. Donald J. Patterson

Content adapted from Essentials of Software Engineering 3rd edition by Tsui, Karam, Bernal
Jones and Bartlett Learning

Overview

- Introduction
- Interaction Design
- Decisions
- Requirements
- Design Constraints
- Design Decisions
- Testing
- Implementation

Overview



- Introduction
- Interaction Design
- Decisions
- Requirements
- Design Constraints
- Design Decisions
- Testing
- Implementation

Introduction

Writing a program

- Creating software is a team effort
- Teams require governance
  - Leadership
  - Processes
  - Standards
  - Institutional Artifacts that transcend an individual

Introduction

Team sizes

- Individuals
  - Personal projects, school projects, learning
- Pizza sized teams
- Federated teams
  - Boeing
  - Microsoft
  - Google

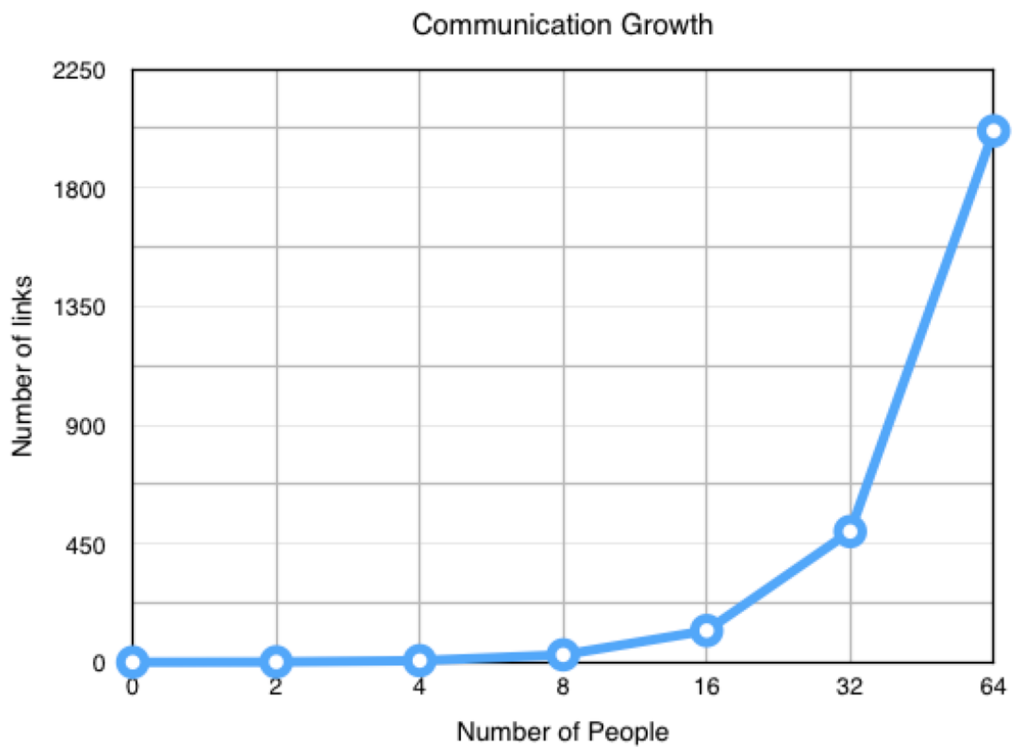
Introduction

Software Teams

- Software requires understanding abstract arrangements of information
- This requires communication - lots of it $\frac{(n)(n-1)}{2}$
- As teams grow the interconnections grow

Table 1

# of people	# of links
0	0
2	1
4	6
8	28
16	120
32	496
64	2016



Introduction

Software Teams

- Pizza-sized teams limit communication links
- Larger teams require hierarchy to manage complexity
  - This can slow down architecture
- Architecture becomes influenced by communication structure

7 nodes hierarchical

6 nodes

FOODBANK
SANTA BARBARA COUNTY

MEMBER OF FEEDING AMERICA

Search | Select Language

MOVING THE COMMUNITY FROM HUNGER INTO HEALTH

THE FOODBANK

SUPPORTS 300 OF YOUR FAVORITE SANTA BARBARA COUNTY CHARITIES!

1 in 4 people need your help.

The Foodbank touches so many people across Santa Barbara County, and each connection is valuable to us. Anyone who connects with the Foodbank is our client – whether in a nutritional, educational, or supporting role.

Introduction

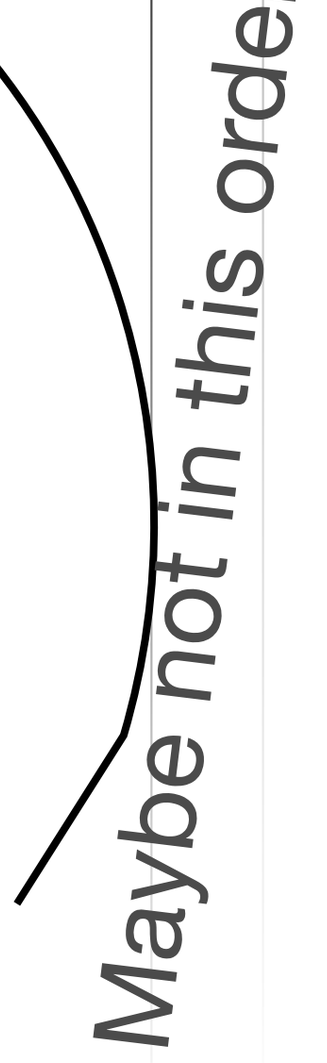
We all “start” by learning how to code in some programming language.

- With a small, hypothetical, and fairly well defined problem
- Usually the code is within one module

Introduction

We then learn that the program usually does not work on the first try, second try ------ may be even 5th or 6th try!

- We learn about “testing” the program
- We learn about re-reading and re-thinking the (problem) requirements more carefully --- then find that we may not have all the answers
- We learn about tracing and “debugging” the program
- Then ---- somehow magically ---- we decide that it’s “good enough !”



“Simple” Set of Steps

1. Understand the problem

- Contextual Inquiry
  - Study the environment in which the intervention will be deployed
- Requirements Engineering
  - Work with the client to figure out
    - Functionalities
    - Non-functionalities
      - performance, security, modifiability, marketability, etc

“Simple” Set of Steps

2. Perform Some Design

- Organizing the functionalities in some sequence;
  - possibly using some diagrams
- Focus on input/output (data, formats, organization)
- Think about some constraints (non-functionalities) such as speed, UI looks, programming language, dependencies, etc.
- any specific algorithm and improvements on sequence of functionalities.

“Simple” Set of Steps

3. Code/Implement

- Turning the design into actual code
  - Depending on how much design is completed, one may either directly engage in conversion to code (language dependent) or do some more designing.
    - Converting input/output to specific UI Interface or I/O format
    - Sequencing the processing in the desired order
    - Ensuring and converting the processing “algorithm” correctly to the target language construct.
    - figure out how to use language library (properly)

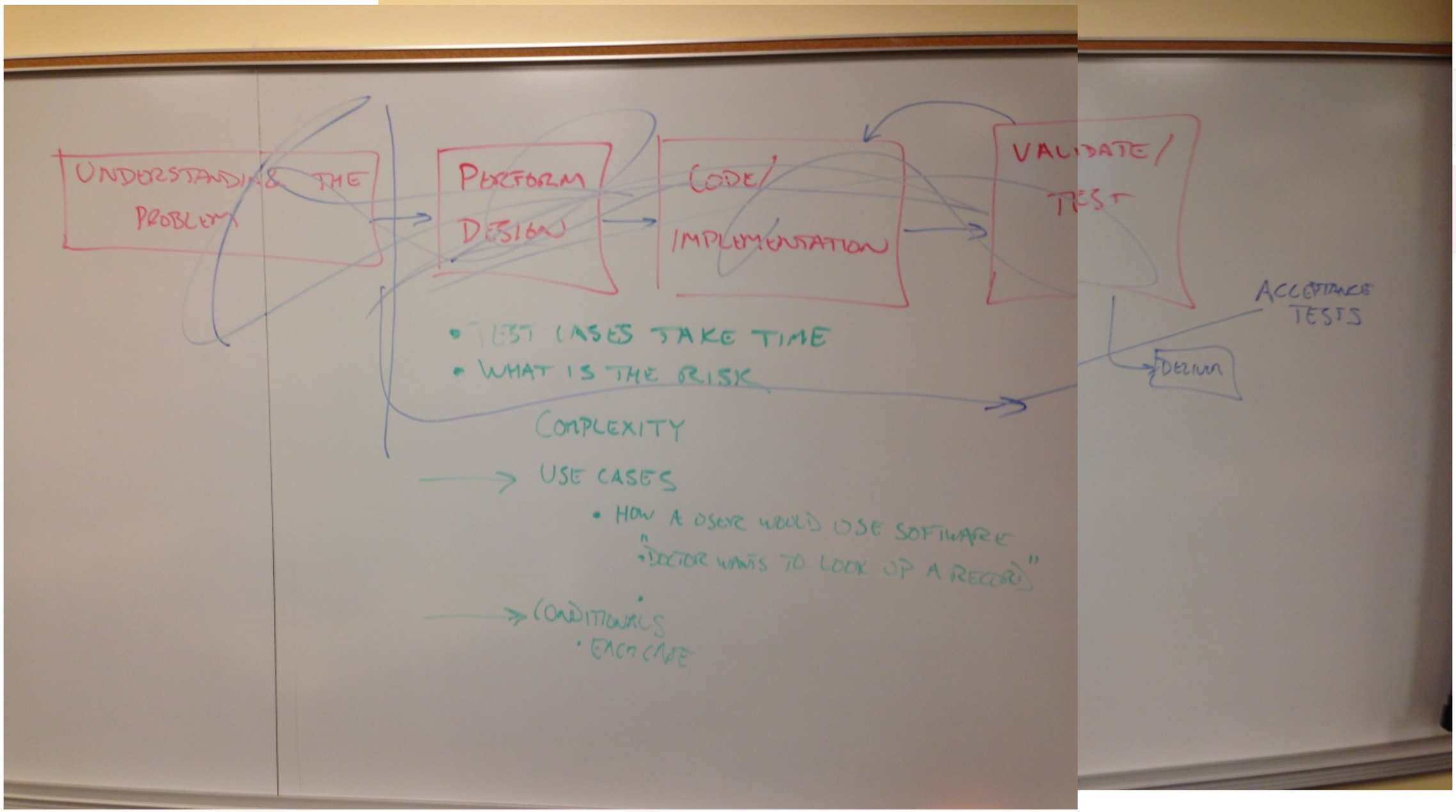
“Simple” Set of Steps

4. Validate/Test the program

- check the program results (via output) with some predetermined set of inputs.
  - The pre-determined inputs are “test cases” and requires some thinking.
- If the results do not match what is expected then:
  - “Debug”
  - Fix
  - Retest ---- revalidate
- Stop when all test cases produce the expected results.

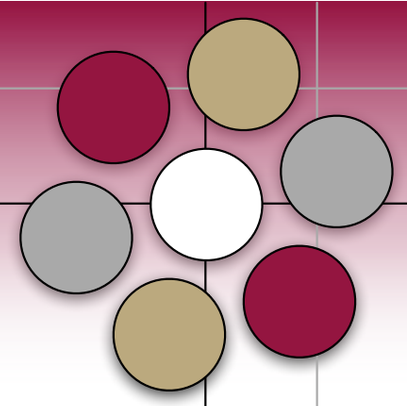
How many test cases should we develop and run?

Narrative vs. Reality



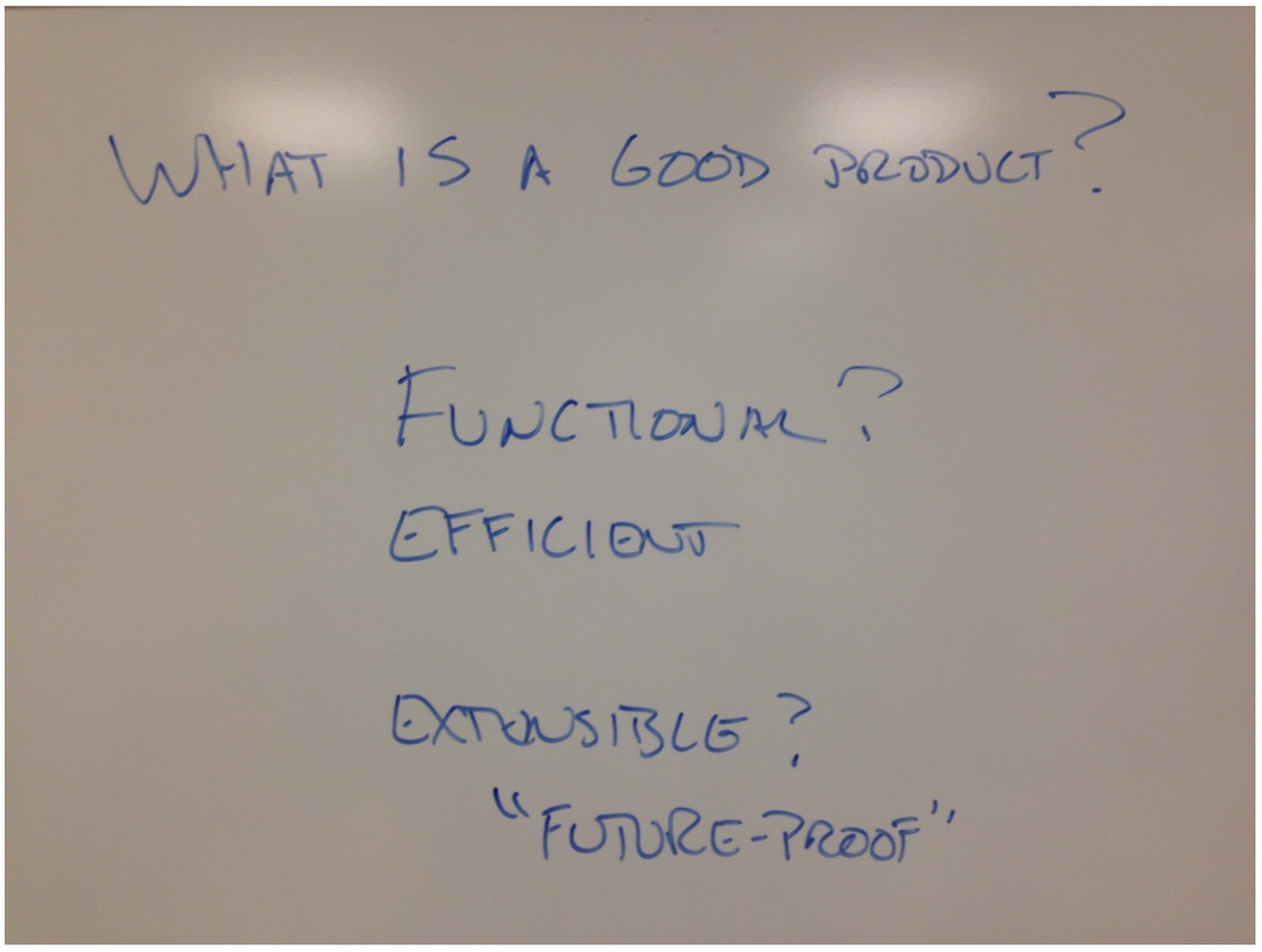
- The real process is a messy mix of the idealized process
- At the end, acceptance tests are contractual obligations

Code is “Done!” What Else Matters?



- How Long (elapsed time) did it take to complete the work?
- How much effort (total person hours) is expended to do the work?
- Does the solution solve the complete problem?
- How “good” is the work – (code, design, documentation, testing, etc.)?

How “good” is the work?



Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”

- Think about it for a second, then let’s discuss missing requirements

- 'THE FILE'
  - TYPE
  - LOCATION
- SORT BY ALPHA?
  - WORDS
  - LINES
  - NON-ALPHABETIC CHAR.
- WHAT IS A 'LINE'?

Decisions

Specifying the problem

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”

- It turns out that this task is under specified

- We need to clarify
  - the program requirements
  - the design constraints

Decisions

Complete specification

- The program requirements
  - State what the program does
  - Qualify what the program does
- Design constraints
  - Provide boundaries for ways in which the program can be created

Decisions

Requirements

- Not exactly the same as in common English
- Requirements cost money
  - Many are negotiable. Which ones?

Good

Fast

Cheap

Choose 2

Decisions

Requirements

- Functional Requirements
  - What the program does
- Non-functional Requirements
  - How the program behaves

Decisions

Requirements

- Functional Requirements
  - What the program does
    - “Sort a file”
- Non-functional Requirements
  - How the program behaves
    - Performance
    - Usability
    - Maintainability

Decisions

Design Constraints

- What languages can you use?
- What frameworks can you use?
- On what platforms must it run?

Decisions

Requirements vs Design Constraints

- The categories are not always clear cut
  - Functional Requirements
  - Non-functional Requirements
  - Design Constraints

- Some requirements can be de facto design constraints

Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”

- What are the functional requirements?
  - Brainstorm some

Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Functional requirements:
  - What is the format of the input data?
  - How is the data stored?
  - What is “a character”?
  - What is “a line”?

Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Functional requirements:
  - What order should the sort be?
  - How should sort react to non-alphabetic characters?
  - How should sort react to numbers?
  - Upper-case vs lower-case

Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Functional requirements:
  - Special cases:
    - Empty file?
    - Empty line?

Simple Problem

Thought Experiment

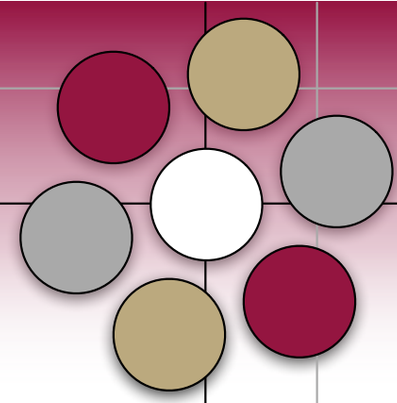
- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Non-Functional requirements:
  - Performance Requirements:
    - How long should it take?
  - Real-time Requirements
    - What about the variability in performance? Worst-case vs. Average-case
  - Modifiability in the future?

Simple Problem

Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Design Constraints
- What’s the user interface like? GUI vs CLI? web-based?
- Typical input size?
  - How much should we worry about algorithm?
  - Will it fit in memory? on disk? on one rack?

Simple Problem



Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Platforms
- What OS?
  - Often a business decision based on licenses, other systems
  - The computational world is fragmented, but each new platform incurs additional cost

Simple Problem

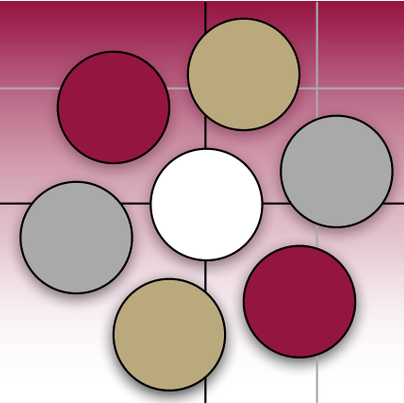
Thought Experiment

- “Given a collection of lines of text stored in a file, sort them in alphabetical order, and write them to another file”
- Schedule
- When does it have to be delivered?
  - Speed costs $$
  - $$ can’t make everything possible however

Thought Experiment

- 1 byte characters (UTF-8)
- Sort ascending, treating digits as characters, upper and lowercase differently in Unicode order
- Empty lines not special, empty lines make empty files
- 1 minute to sort 100 lines of 100 characters
- no real-time requirements
- no modifiability requirements
- We should have a GUI, run on Mac, in Java
- typical input size will be 100 lines
- For Prof. Patterson: Due ASAP

Prof. Patterson Experiment



I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Decisions

Testing

- Acceptance Tests
  - High-level evaluations by a client, contractually bound
  - Do a sorting walk-through of examples
- UI Tests
  - Automated tests that simulate UI input
- Functional tests
  - Automated tests that simulate scenarios
- JUnit tests
  - Automated tests that test method and class specs

Decisions

Testing

- “Extreme Programming” methodology
  - Write the tests before you write the code

Decisions

Estimating Effort (aka Project Management)

- Breaking down a problem into sub-tasks
- Estimating the time for each
- Assigning a cost to the project

Decisions

Exercise

Write a “program” in your favorite language that will accept numerical numbers as inputs, compute the average, and output the answer.

Provide an estimate within one minute:

- How long (in elapsed-time) would it take you to implement this solution?
- How much overall effort (in person-hours) will this take?

How well will your solution match the problem?
How good is your code/design/documentation/testing?

Decisions

Estimating Effort

- Did you include meal breaks, rest breaks, bathroom breaks?
- Did you break down the task?
- Did you include GUI? Testing? Bug fixing?

Decisions

Estimating Effort

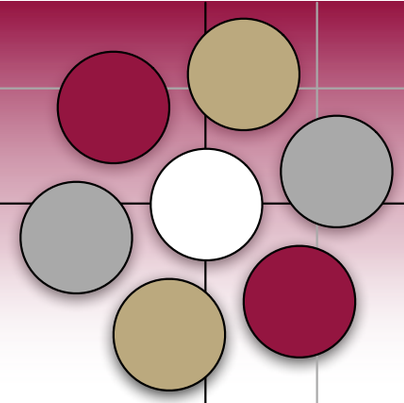
- This is one of the toughest problems in software project management.
- Accurate estimates are very hard to make
- Estimates should be made by the person assigned to the task
  - Hopefully after some reflection and data on performance.

- Pivotal Tracker (video), JIRA (video)

Previous Class Answers

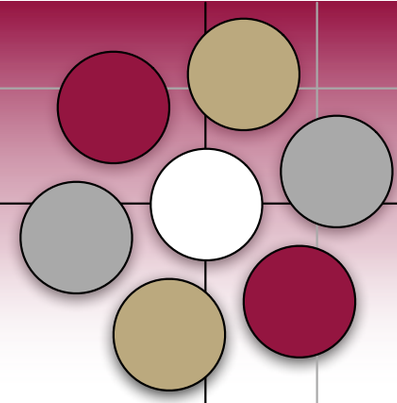
- How long (in elapsed-time) would it take you to implement this solution?
  - Class Answer: 10 min. (A); 15 min (B); 1 hr. (C);
- How much overall effort (in person-hours) will this take?
  - Class Answer: 10 person min. ; 15 person min. ; 1 person hour; 3 person-hrs
- Will your solution match the problem?
  - Class Answer: YES!
- How “good” will your solution be?
  - Class Answer: Awesome!

Some “Previous Class” Inputs



- How long do you think assignment #1 would take?
  - 1 hr --- 7 people
  - 2 hrs --- 6 people
  - 3 hrs --- 2 people
  - 10 hrs --- 3 people
- Real data from class:
  - Elapsed time: range was 5 days to 46 minutes – mostly between 1 to 3 hours
  - Effort: range was 8 person-hours to 40 person-minutes – mostly between 1 person-hour to 3 person-hours

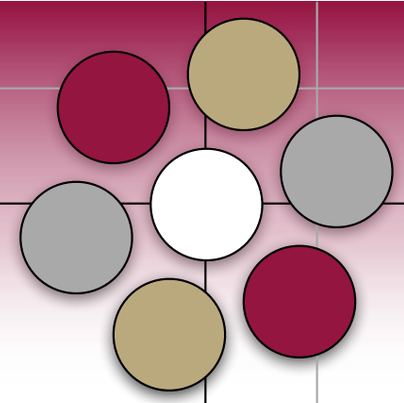
Implementation



Conventions help teams move faster

- Teams need to agree on syntax conventions and stick to them
- When do you use capital letters?
- How are things named?
- How are comments utilized?
- Testing before giving code to others
- Having someone else review your code before sharing

Prof. Patterson Experiment



I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

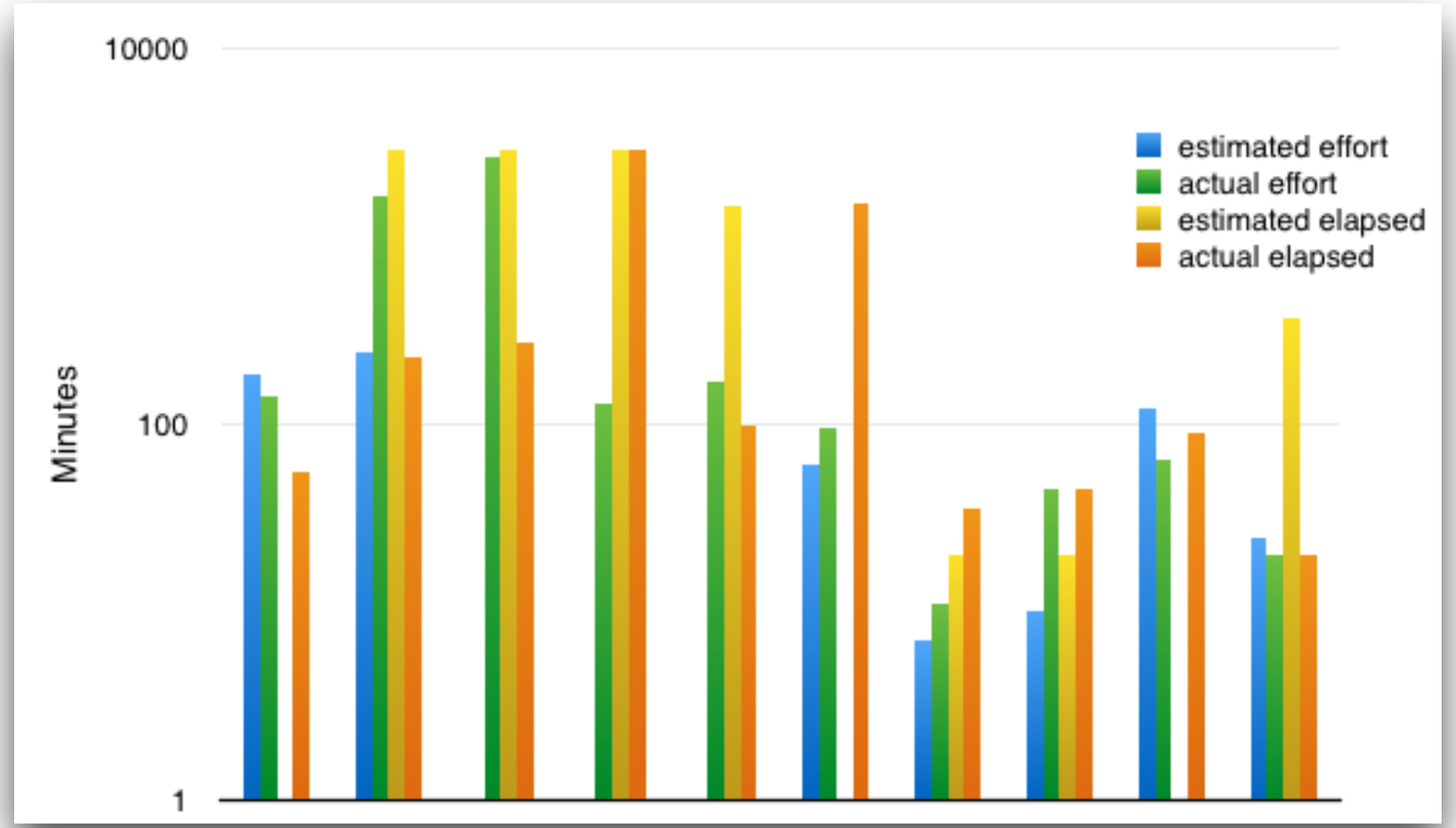
I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours

Prof. Patterson Experiment

I estimated 30 minutes to do the task

- 10:00 - started
- 10:15 - Eclipse crashed
- 10:30 - Decide initial design was bad
- 11:00 - laptop battery died - no charger - stop
- 13:00 - restart
- 13:21 - done debugging
- 13:36 - done writing tests
  - effort ~ 1.6 person hours
  - elapsed time ~ 3.5 hours



- effort estimate:
  - 220 minutes low (on average)
- elapsed estimate:
  - 986 minutes high (on average)

- ideal time or effort: straight through with no interruptions
  - units: e.g., person-hours, person-days, etc.
- elapsed time or duration: actual calendar time including everything
  - units: e.g., days, weeks, etc.

- Friend stopped by interrupted me.
- Had to go to the ER
- Distracted by thing near me
- Lot of problems converting strings to text
- Test files changed, set me back
- sleep
- needed to clarify assignment
- chatting with roommate
- bug fix
- getting help
- sleep
- other homework
- I needed to work on other hw

Building a System

Moving from
writing a program to building a system

https://flic.kr/p/a8724x

Building a System

Moving from
writing a program to building a system

https://flic.kr/p/r3meit

Building a System

Moving from
writing a program to building a system

- What's the difference?

https://flic.kr/p/a8724x

https://flic.kr/p/r3meit

Building a System

Moving from
writing a program to building a system

- What's the difference?

https://flic.kr/p/a8724x

https://flic.kr/p/r3meit

- Size, which only matters because of increased complexity

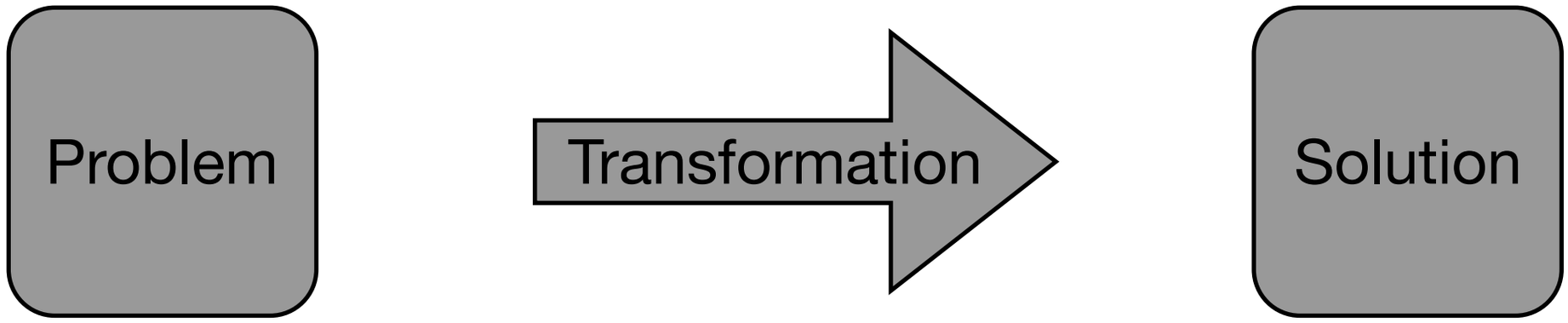
DIFFERENCES

- COMMUNICATION
  - TIME ZONE

- RESPONSIBILITIES ARE COUPLED AMONG MULTIPLE PEOPLE
- WHO HAS THE BIG PICTURE
- LOTS OF COMPUTERS
- SCALE REQUIRES SPECIALIZATION
- CONTINUITY
  - REDUNDANT PROGRAMMERS

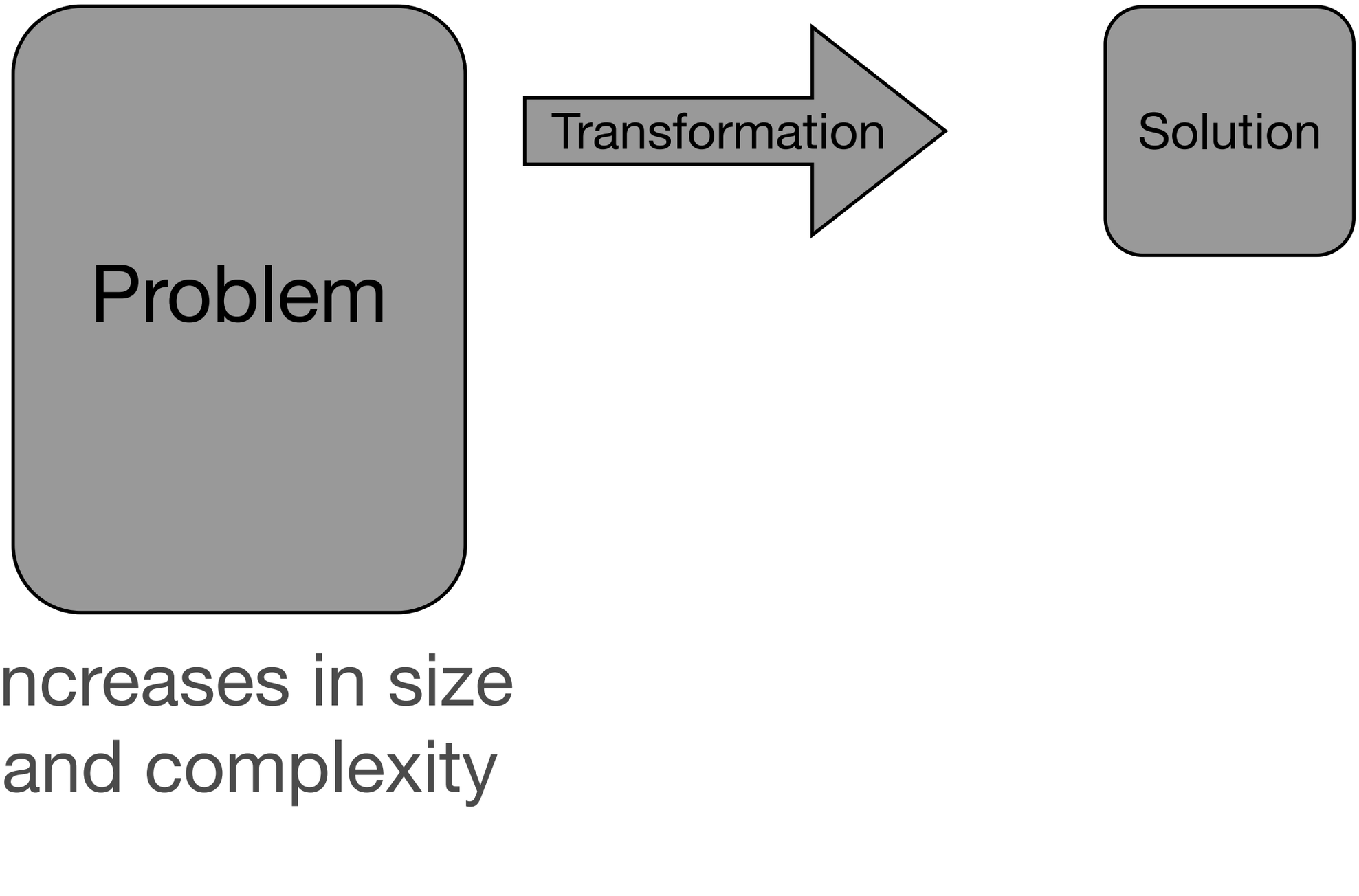
Building a System

Complexity Increases Everywhere



Building a System

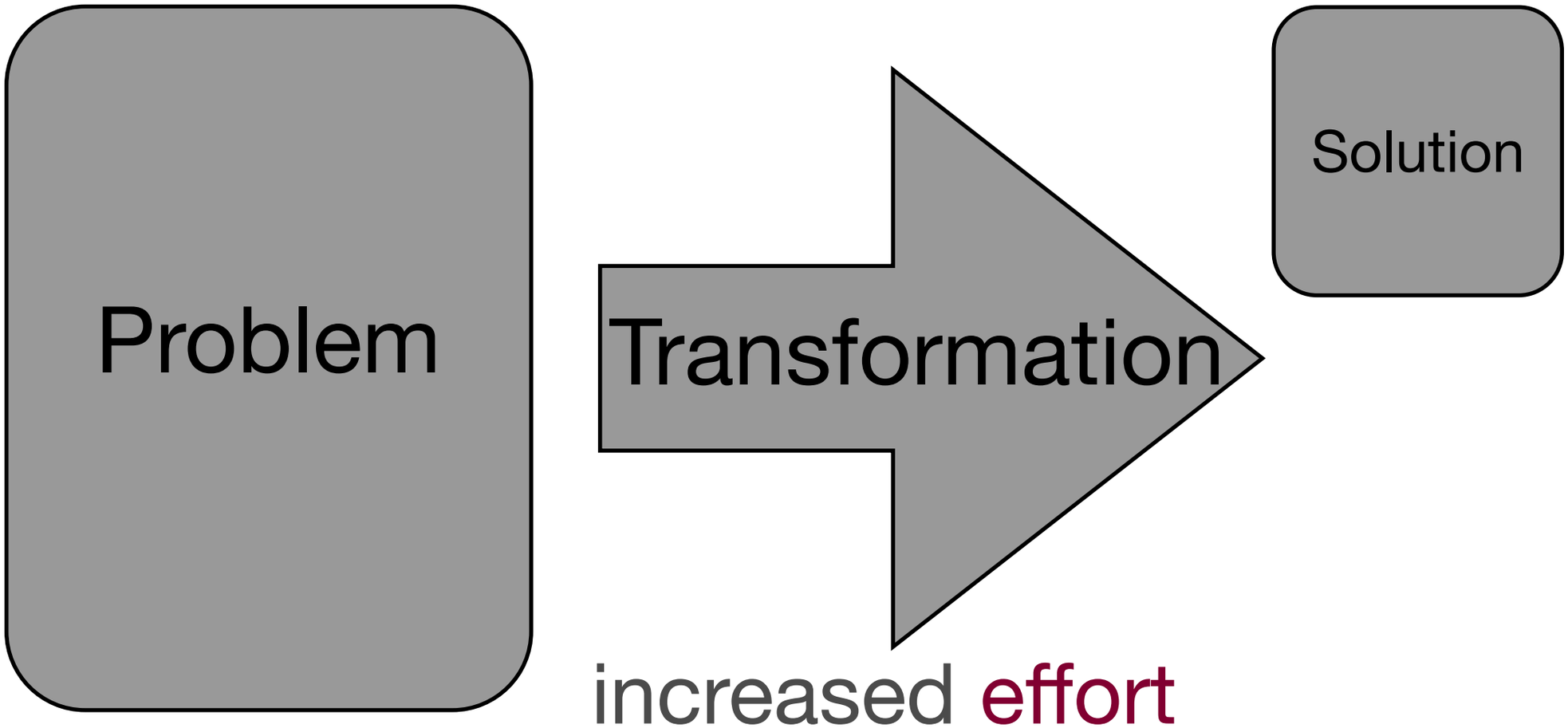
Complexity Increases Everywhere



increases in size
and complexity

Building a System

Complexity Increases Everywhere



increased effort
due to size and
complexity

increases in size
and complexity

Building a System

Complexity Increases Everywhere

increases in size and complexity

increased effort due to size and complexity

increases in size and complexity

Building a System

Complexity

Building a System

Complexity

- Breadth

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed
    - More major functions

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed
    - More major functions
  - More features in each functional area

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed
    - More major functions
  - More features in each functional area
  - More varieties of interfaces to users, internal and external systems

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed
    - More major functions
    - More features in each functional area
    - More varieties of interfaces to users, internal and external systems
    - More simultaneous users, more types of users

Building a System

Complexity

- Breadth
  - The sheer number of issues to be addressed
    - More major functions
    - More features in each functional area
    - More varieties of interfaces to users, internal and external systems
    - More simultaneous users, more types of users
    - More data, types of data and data structures

Topical Heading

For our Assignment 1

Topical Heading

For our Assignment 1

- What is it again?

Topical Heading

For our Assignment 1

- What is it again?

Topical Heading

For our Assignment 1

- What is it again?

- How would our solution change if the input size was increased to 1 trillion?

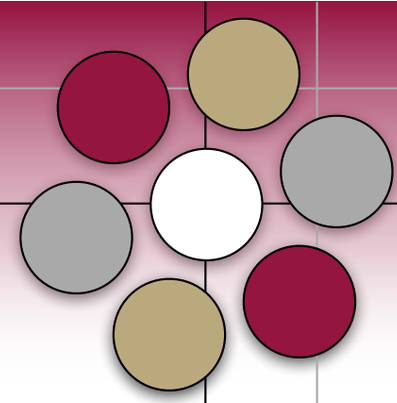
Topical Heading

For our Assignment 1

- What is it again?

- How would our solution change if the input size was increased to 1 trillion?

Topical Heading



For our Assignment 1

- What is it again?

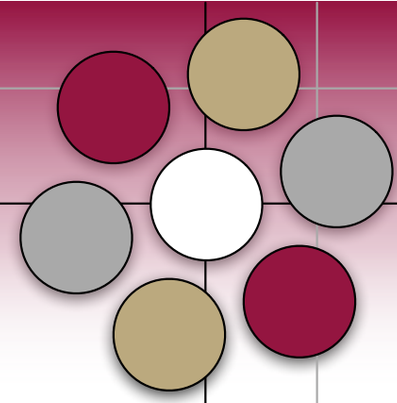
- How would our solution change if the input size was increased to 1 trillion?

- How would our solution change if the numbers were very large?

Building a System

Complexity

Building a System



Complexity

- Depth

Building a System

Complexity

- Depth
  - More linkages and connections

Building a System

Complexity

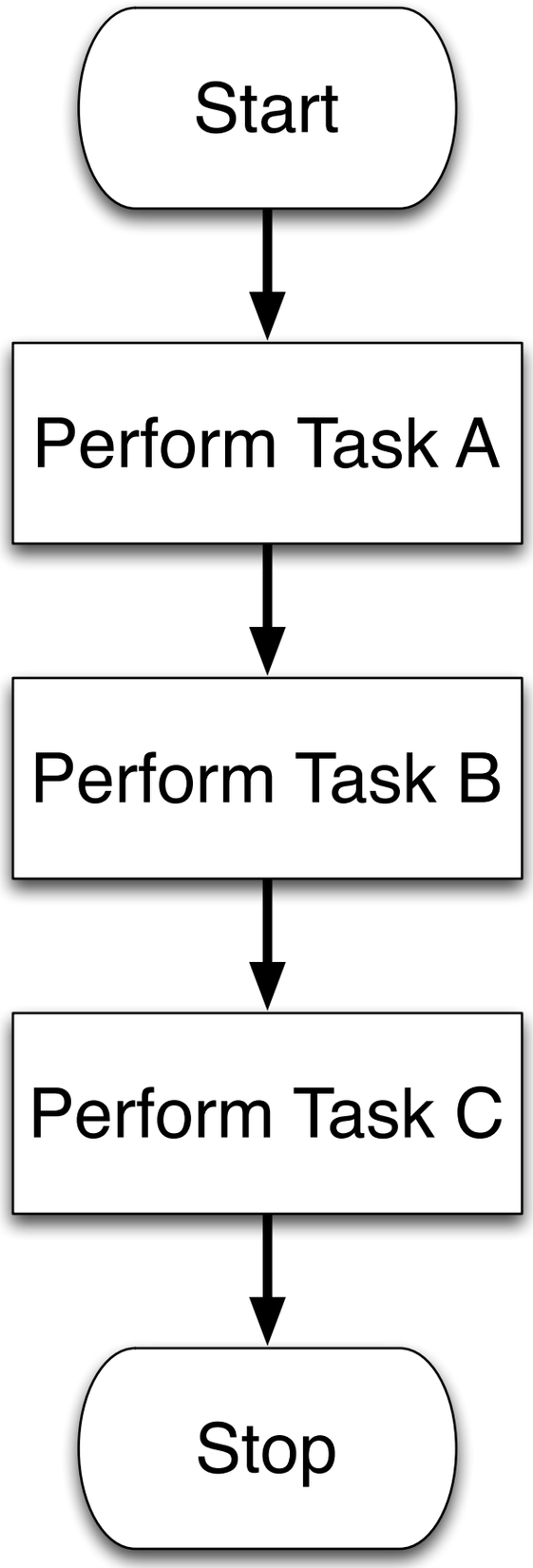
- Depth
  - More linkages and connections
    - Data sharing among the functionalities & logic

Building a System

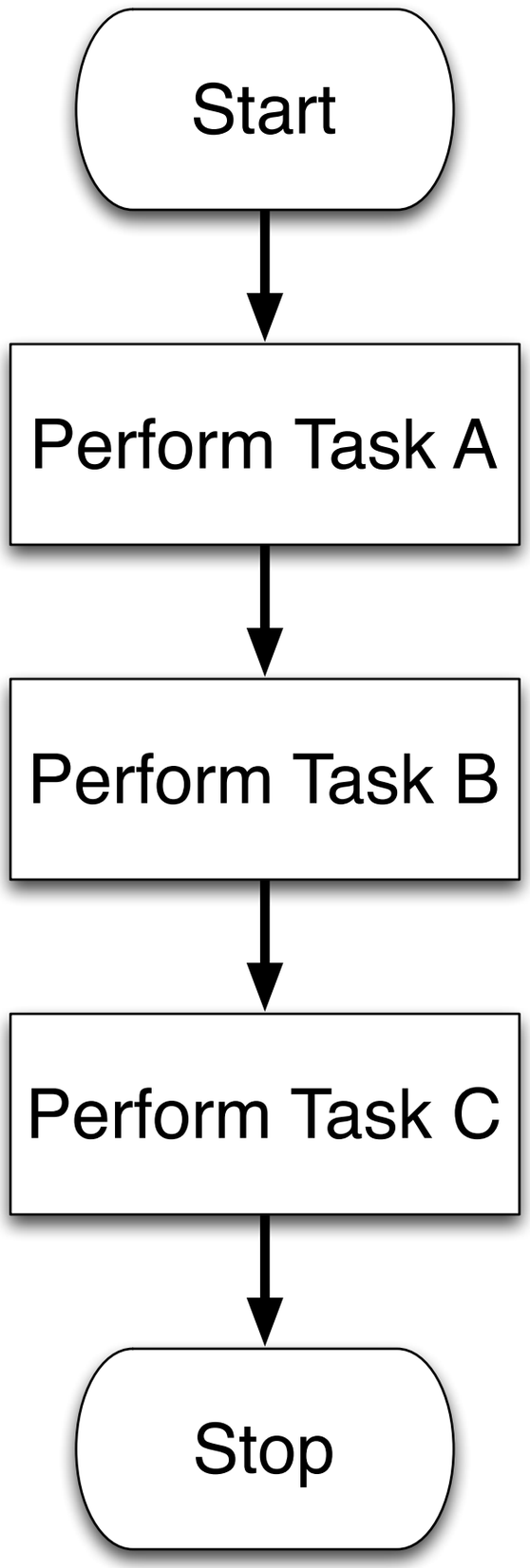
Complexity

- Depth
  - More linkages and connections
    - Data sharing among the functionalities & logic
  - Control Passing among functionalities

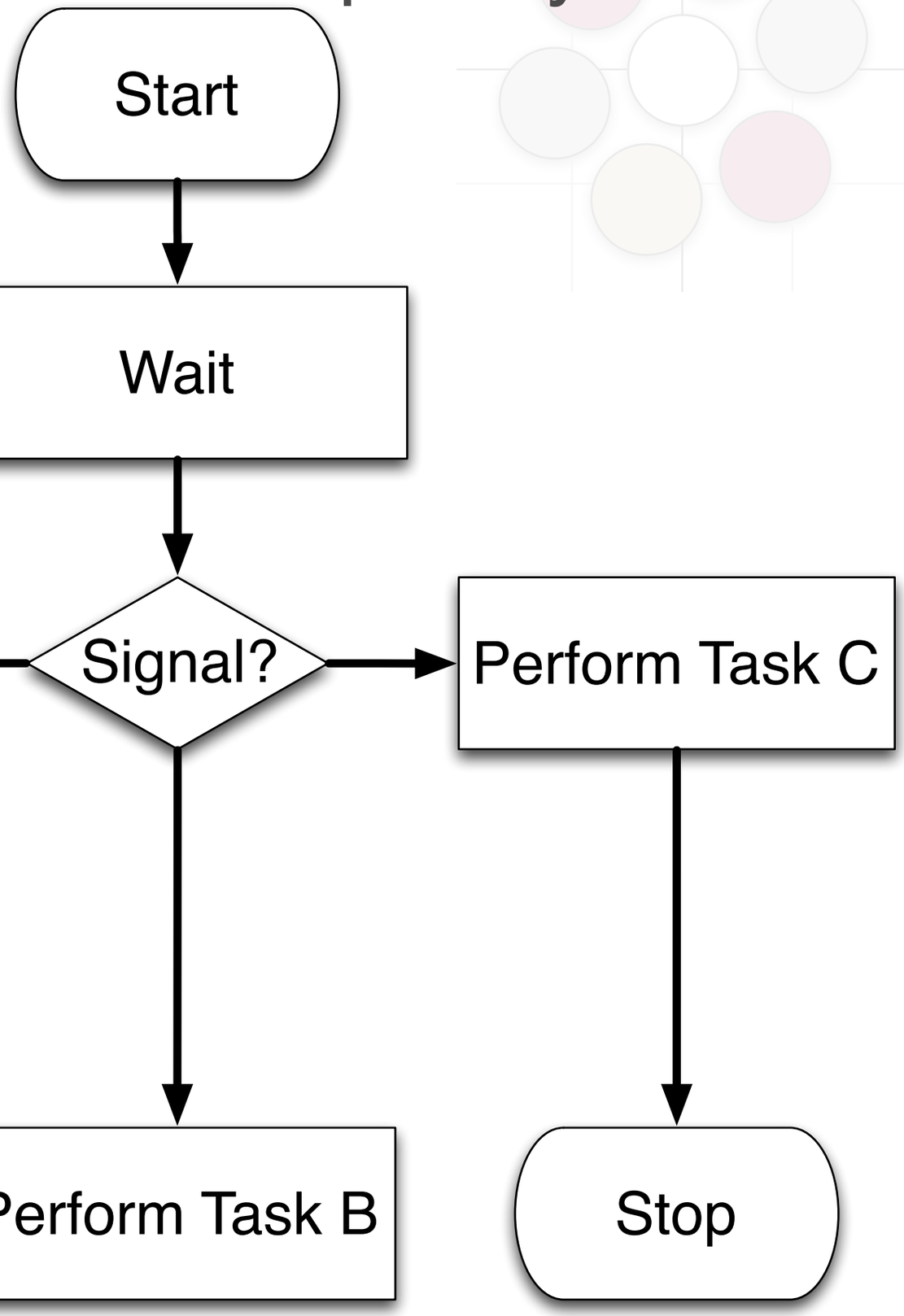
Simple Task



Simple Task



Increased Complexity



Building a System

Board work - Modified Assignment 1

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers
- Additionally show the largest and smallest of the read-in numbers

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers
- Additionally show the largest and smallest of the read-in numbers
- Where is the increased complexity?

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers
- Additionally show the largest and smallest of the read-in numbers
- Where is the increased complexity?

Building a System

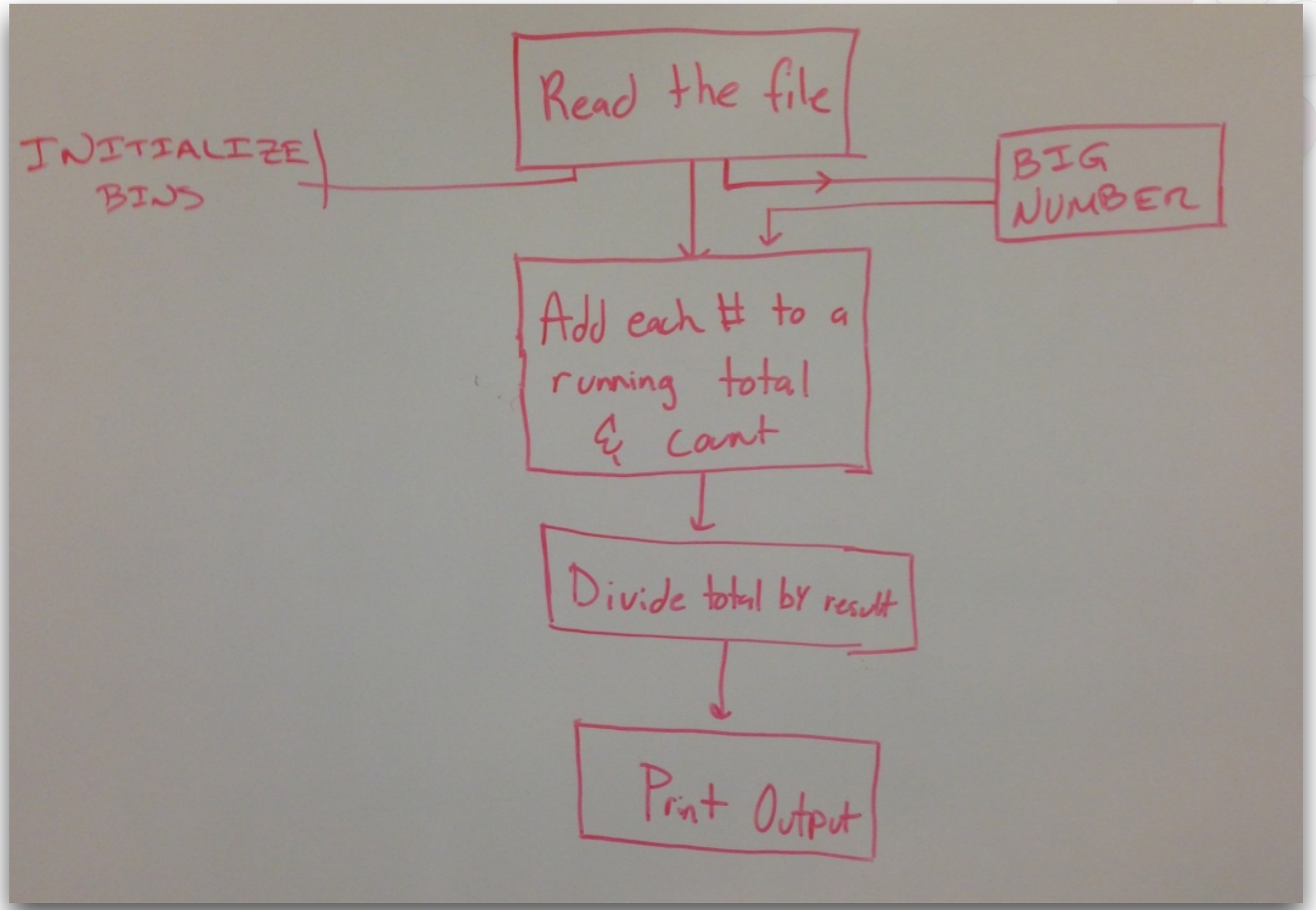
Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers
- Additionally show the largest and smallest of the read-in numbers
  - Where is the increased complexity?
- Additionally show the numbers in sorted ascending order

Building a System

Board work - Modified Assignment 1

- Compute and show the average of the read-in numbers
- Additionally show the largest and smallest of the read-in numbers
  - Where is the increased complexity?
- Additionally show the numbers in sorted ascending order
  - Where is the increased complexity?


Read the file
INITIALIZE BINS
BIG NUMBER
Add each # to a running total & count
Divide total by result
Print Output

Building a System

Handling complexity

Building a System

Handling complexity

- Strategy 1: Simplification

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution
  - Modularization of the solution

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution
  - Modularization of the solution
  - Separation of concerns of the problem and the solution

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution
  - Modularization of the solution
  - Separation of concerns of the problem and the solution
  - Possibly reduce the problem

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution
  - Modularization of the solution
  - Separation of concerns of the problem and the solution
  - Possibly reduce the problem

Building a System

Handling complexity

- Strategy 1: Simplification
  - Decomposition of the problem and the solution
  - Modularization of the solution
  - Separation of concerns of the problem and the solution
  - Possibly reduce the problem
  - Incrementally address the problem components

Building a System

Handling complexity

Building a System

Handling complexity

- Strategy 2: Better technology and tools

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network
  - Multi-developer configuration management

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network
  - Multi-developer configuration management
  - Modeling techniques

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network
  - Multi-developer configuration management
  - Modeling techniques
  - Automated testing

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network
  - Multi-developer configuration management
  - Modeling techniques
  - Automated testing

At first this doesn't seem to be reducing complexity

Building a System

Handling complexity

- Strategy 2: Better technology and tools
  - Database to handle information and structures of information
  - Programming and development platforms
  - Computing network
  - Multi-developer configuration management
  - Modeling techniques
  - Automated testing

At first this doesn't seem to be reducing complexity

Building a System

Handling complexity

Building a System

Handling complexity

- Strategy 3: Improve process and methodology

Building a System

Handling complexity

- Strategy 3: Improve process and methodology
  - Coordinate multiple and different people performing different tasks

Building a System

Handling complexity

- Strategy 3: Improve process and methodology
  - Coordinate multiple and different people performing different tasks
  - Guidance for overlapping incremental tasks

Building a System

Handling complexity

- Strategy 3: Improve process and methodology
  - Coordinate multiple and different people performing different tasks
  - Guidance for overlapping incremental tasks
  - Guidance for measuring separate artifacts and outcomes

Building a System

Handling complexity

- Strategy 3: Improve process and methodology
  - Coordinate multiple and different people performing different tasks
- Guidance for overlapping incremental tasks
- Guidance for measuring separate artifacts and outcomes

Again at first this doesn't feel like it is reducing complexity

Building a System

Handling complexity

- Strategy 3: Improve process and methodology
  - Coordinate multiple and different people performing different tasks
- Guidance for overlapping incremental tasks
- Guidance for measuring separate artifacts and outcomes

Again at first this doesn't feel like it is reducing complexity

Handling Complexity: Macro Task Breakdown

Requirements Engineering

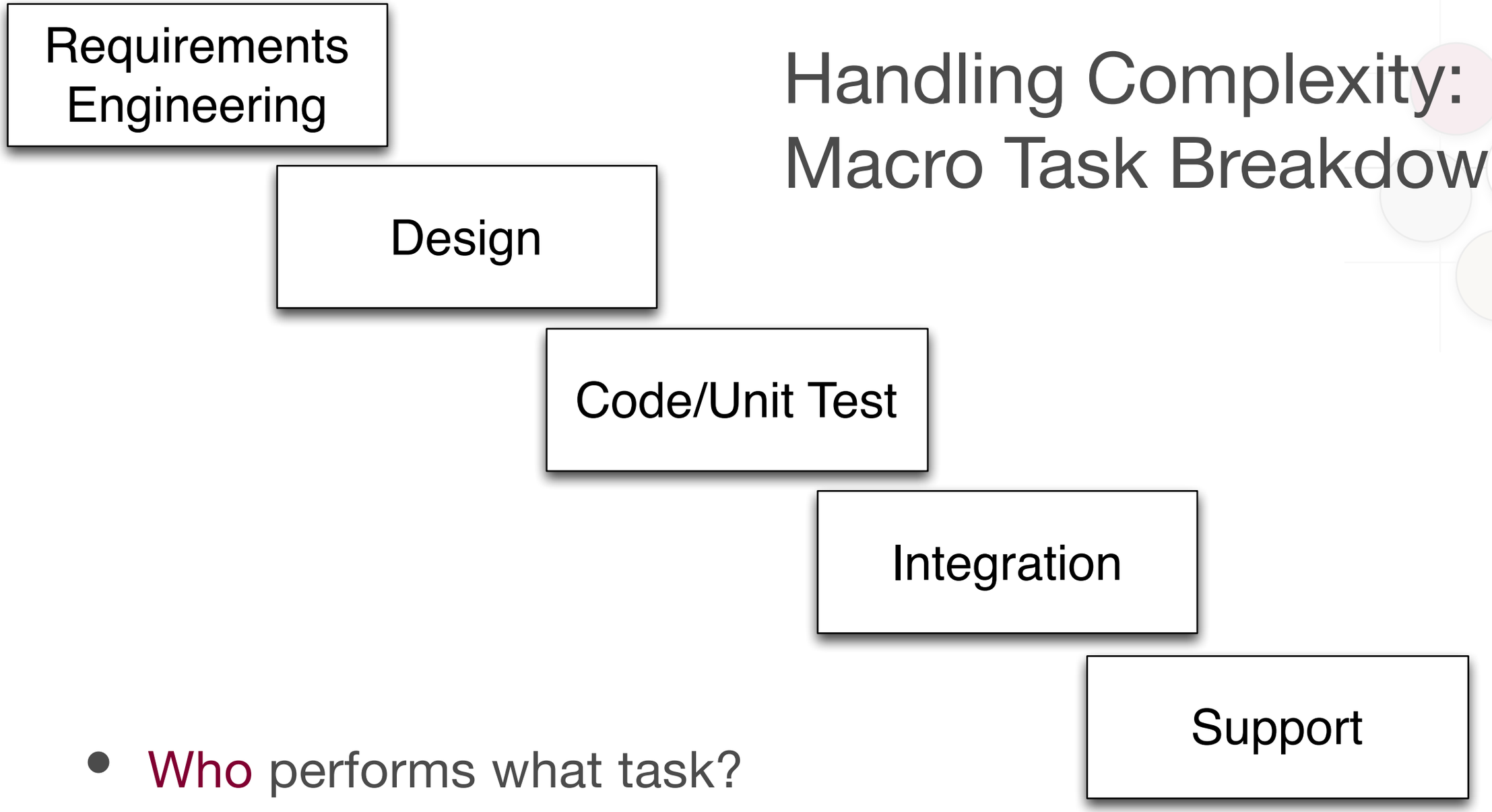
Design

Code/Unit Test

Integration

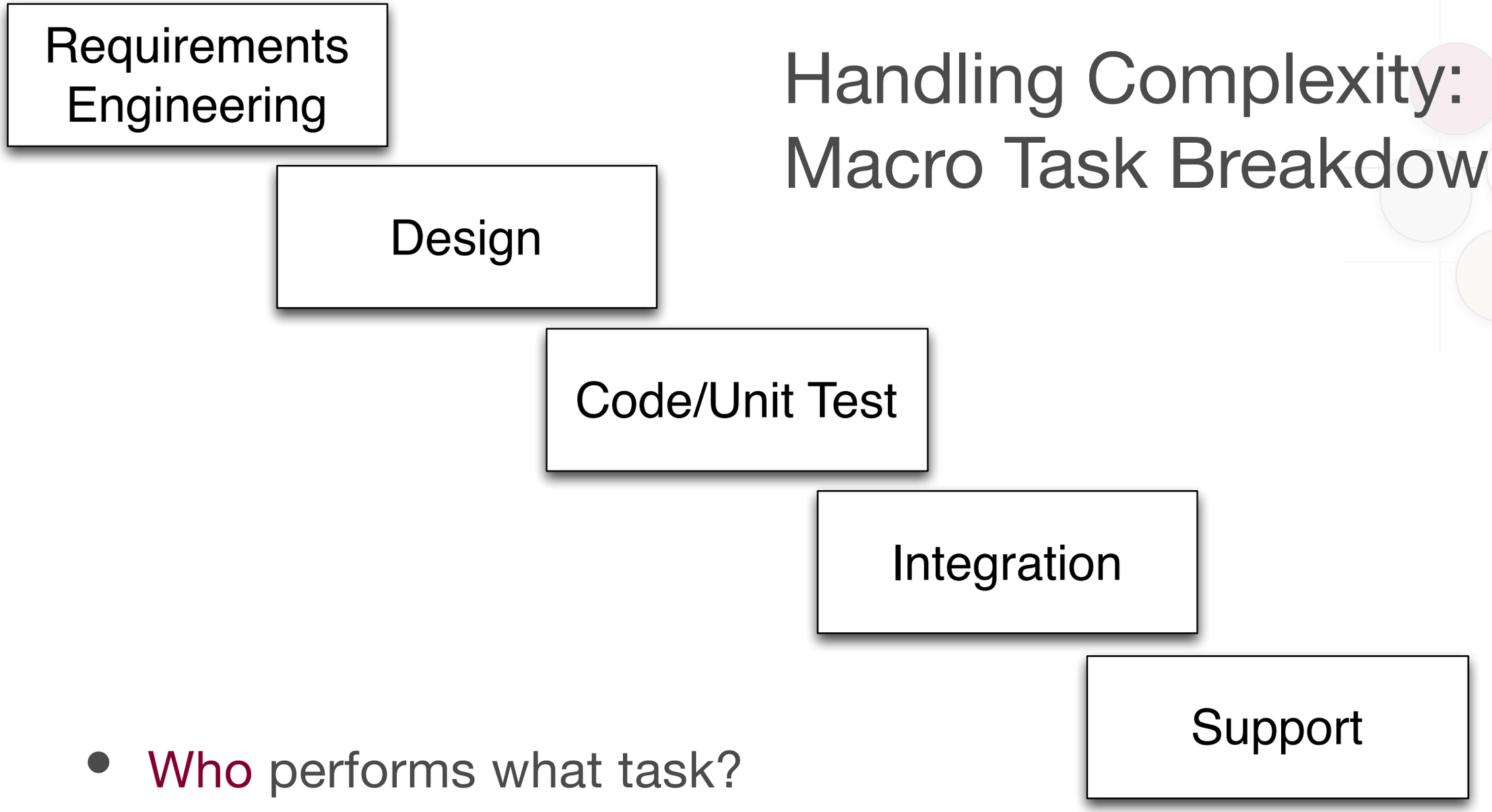
Support

Handling Complexity: Macro Task Breakdown

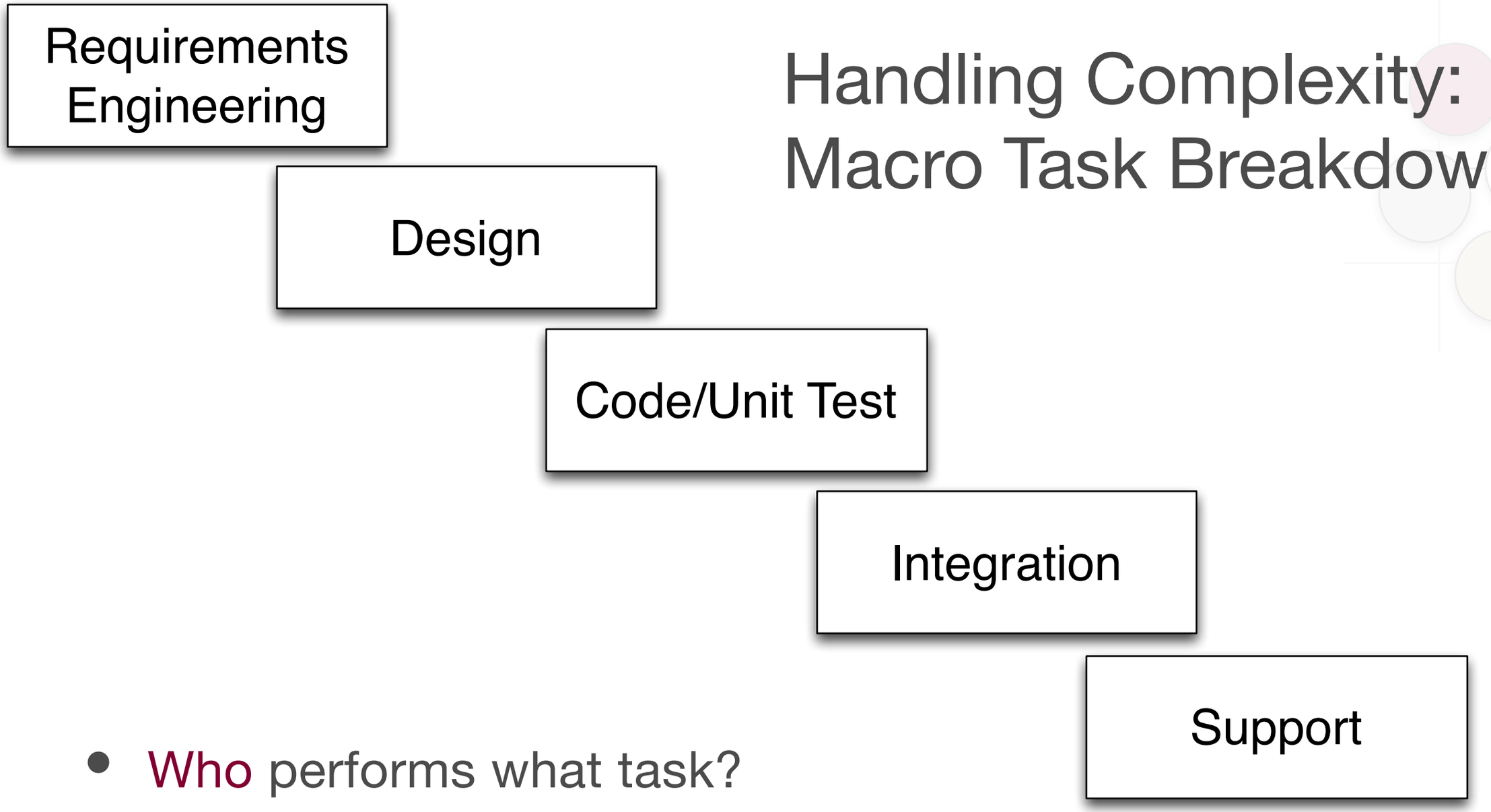


- Who performs what task?

Handling Complexity: Macro Task Breakdown

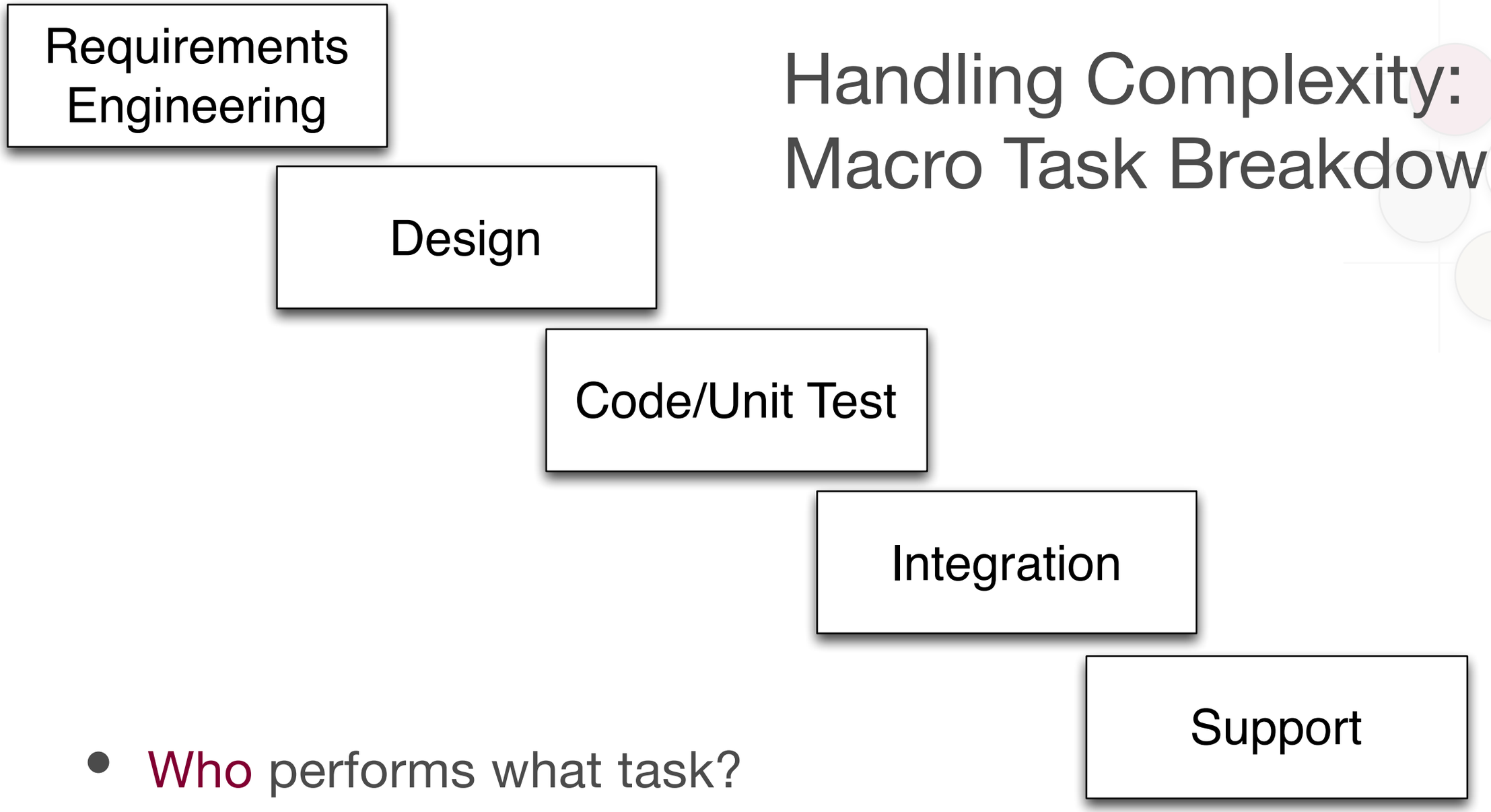


- Who performs what task?
- How is the task completed with what technique or tool?



Handling Complexity: Macro Task Breakdown

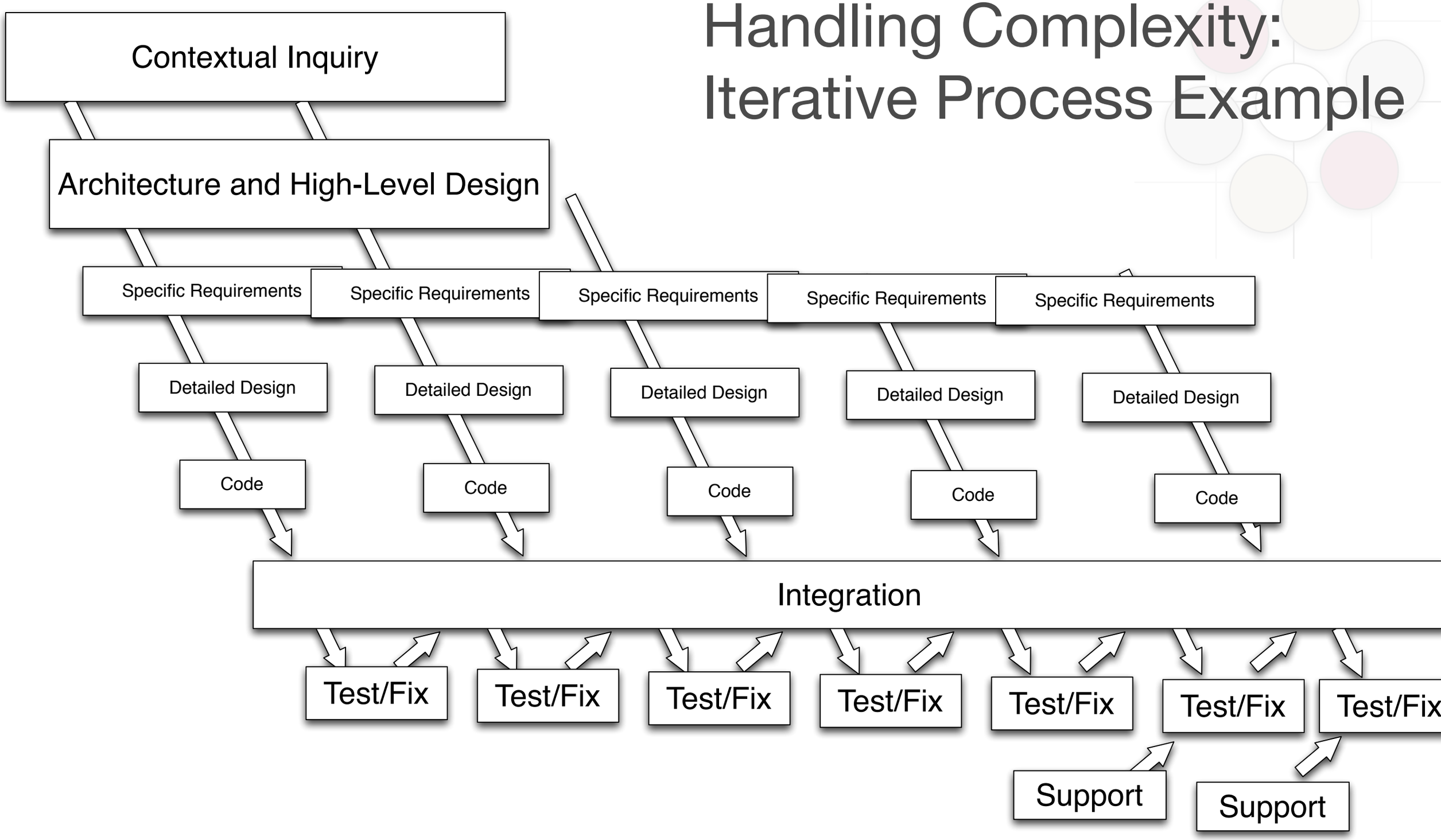
- Who performs what task?
- How is the task completed with what technique or tool?
- When should which task start and end?



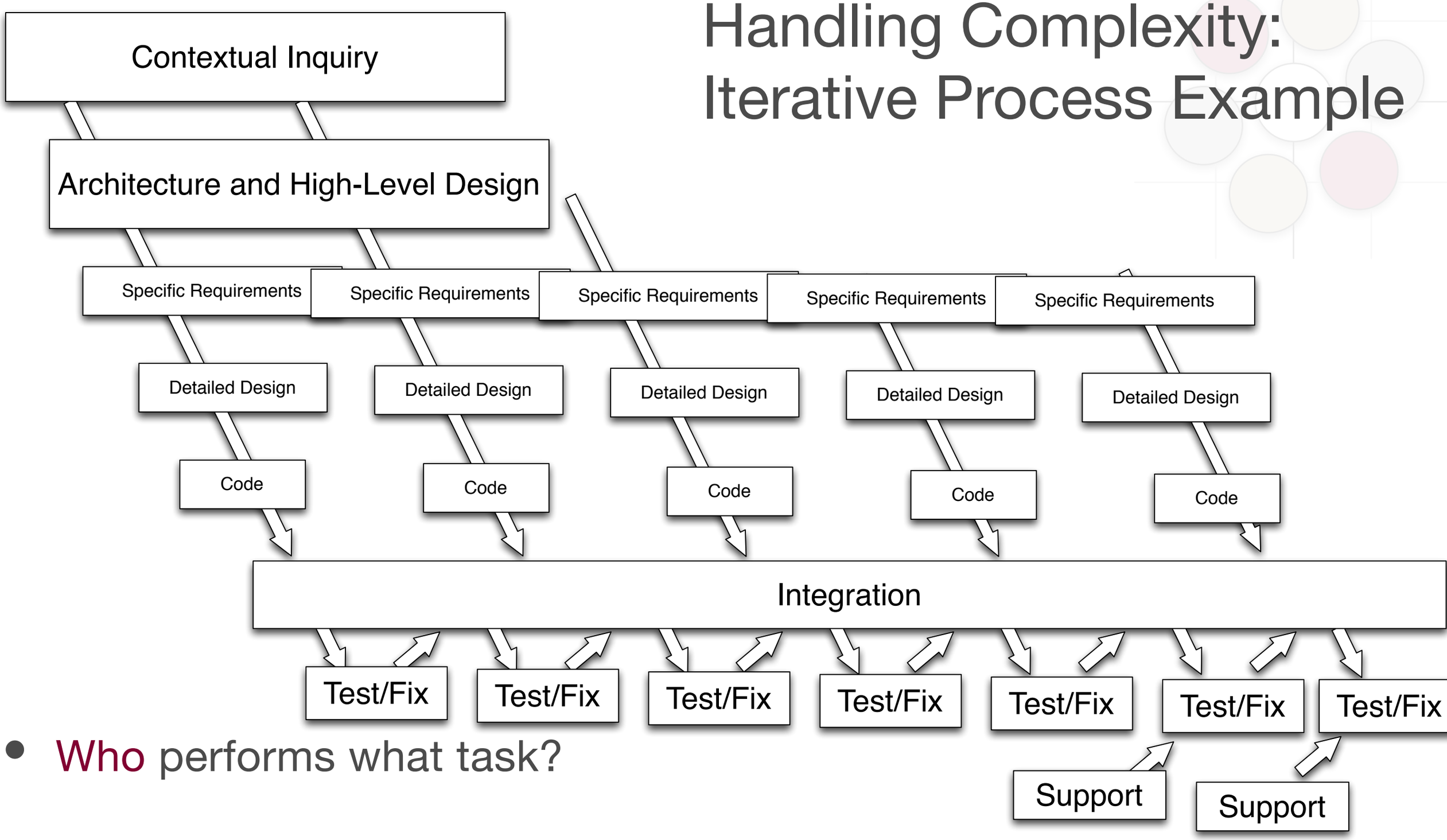
Handling Complexity: Macro Task Breakdown

- Who performs what task?
- How is the task completed with what technique or tool?
- When should which task start and end?
- Who should coordinate the people and the tasks?

Handling Complexity: Iterative Process Example

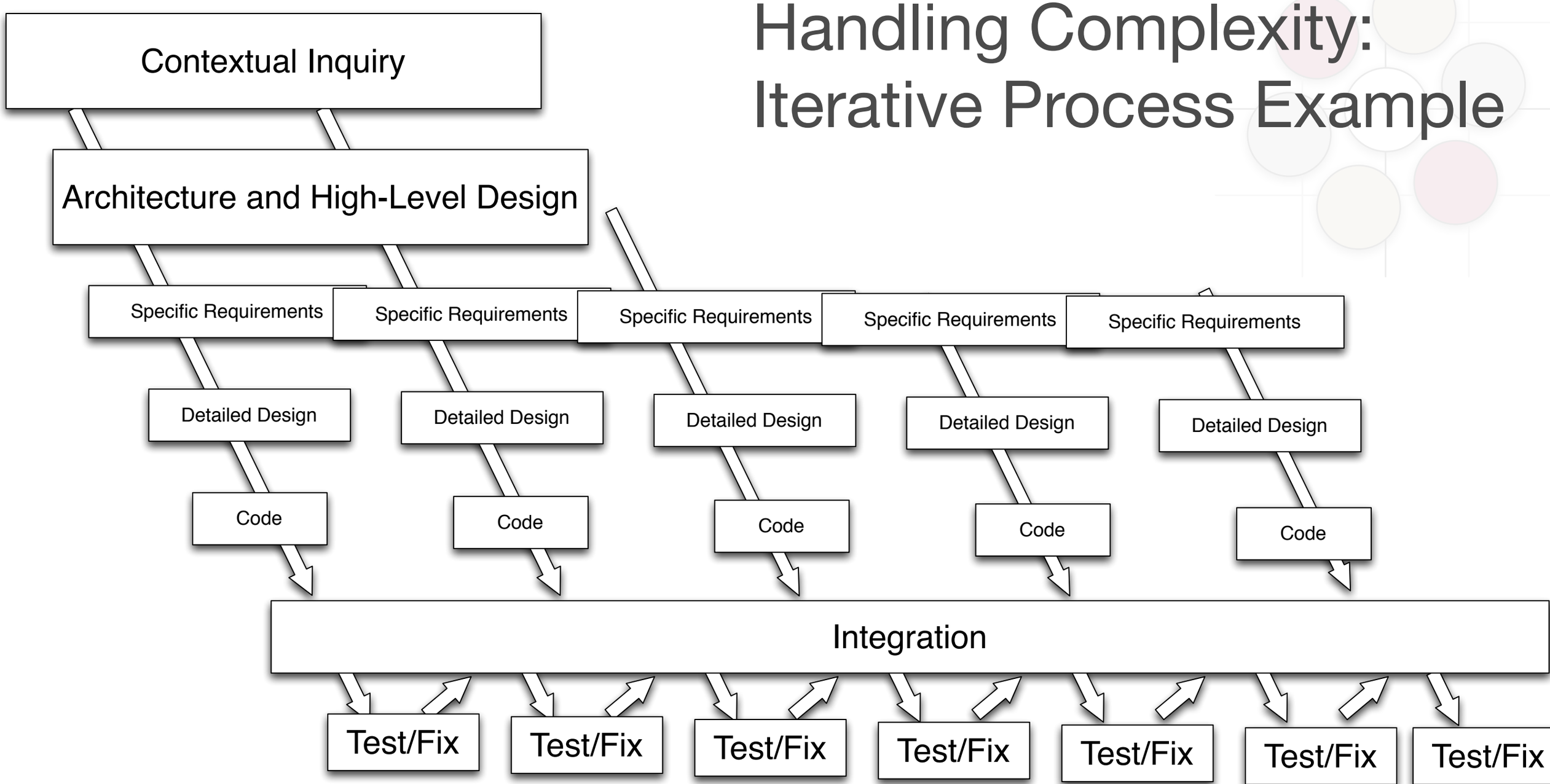


Handling Complexity: Iterative Process Example



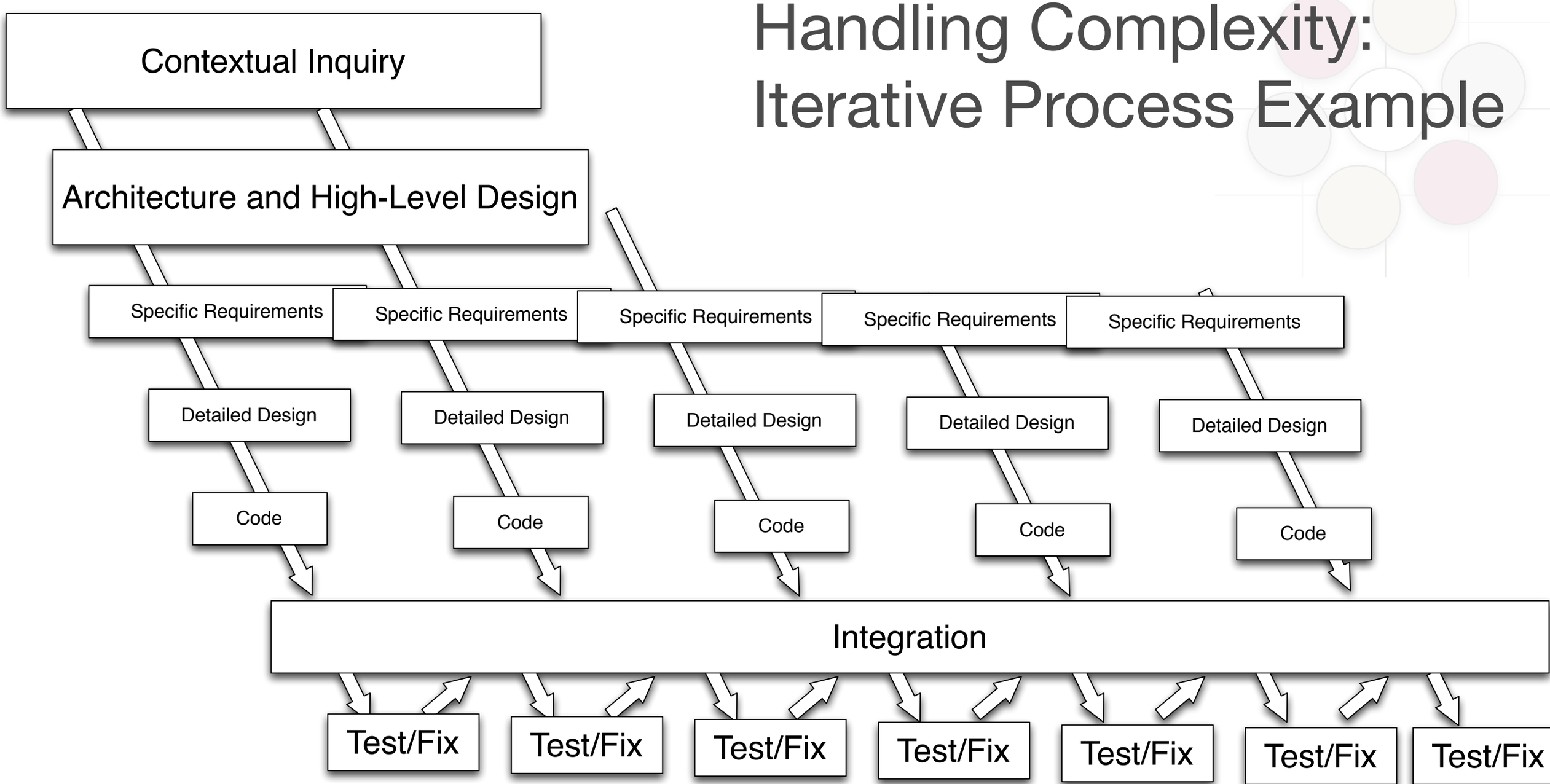
- Who performs what task?

Handling Complexity: Iterative Process Example



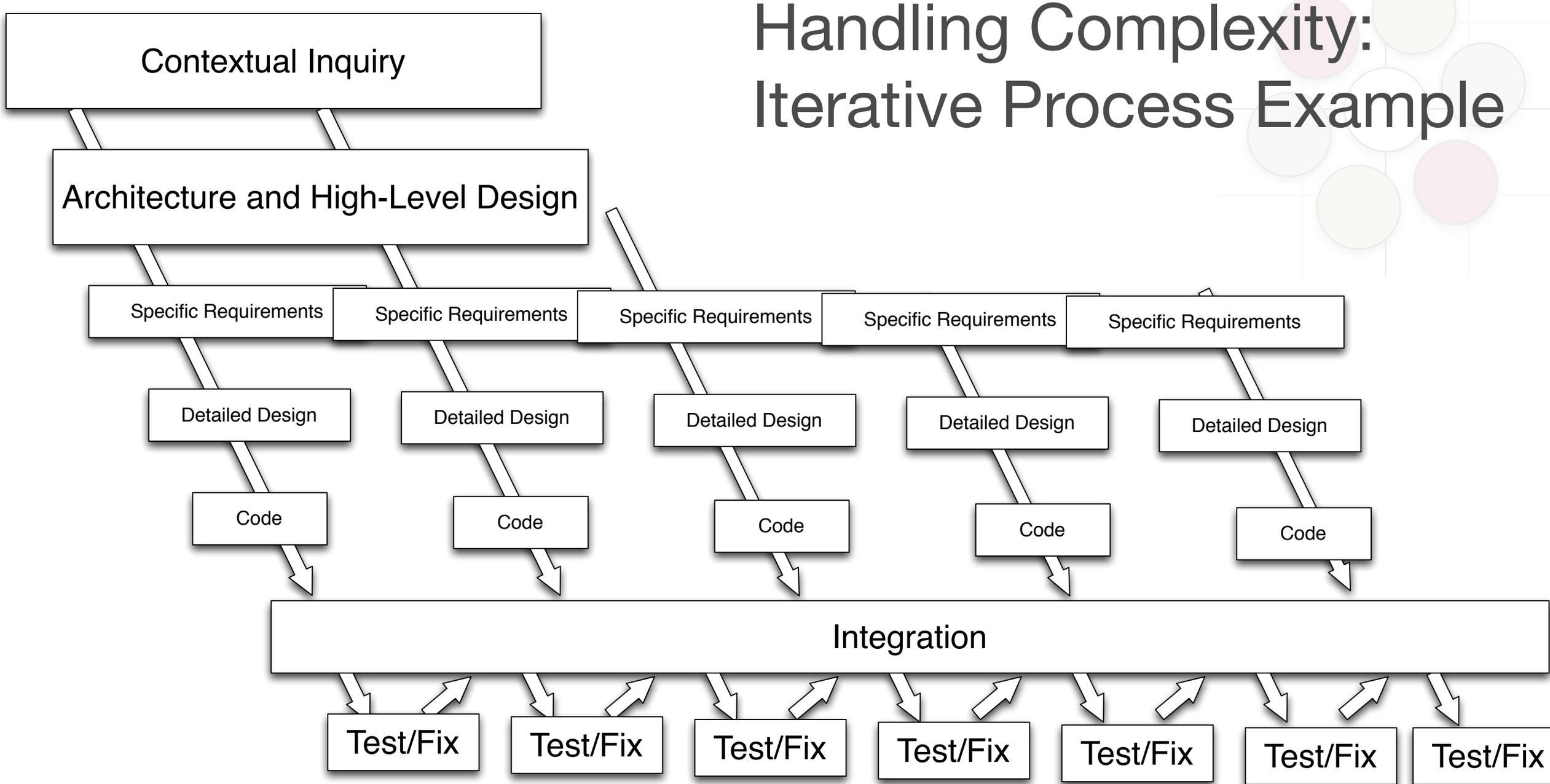
- Who performs what task?
- How is the task completed with what technique or tool?

Handling Complexity: Iterative Process Example



- Who performs what task?
- How is the task completed with what technique or tool?
- When should which task start and end?

Handling Complexity: Iterative Process Example

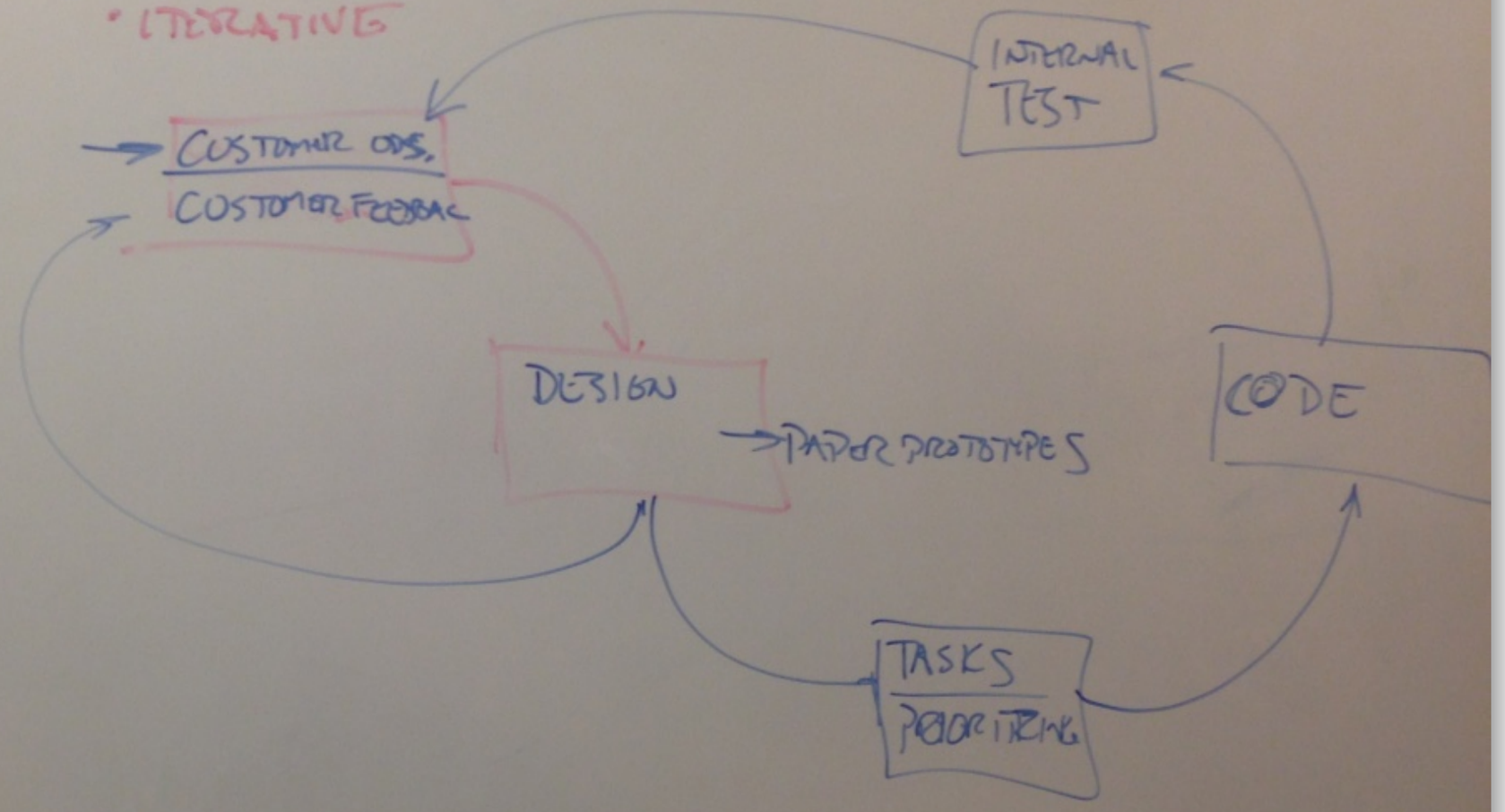


- Who performs what task?
- How is the task completed with what technique or tool?
- When should which task start and end?
- Who should coordinate the people and the tasks?

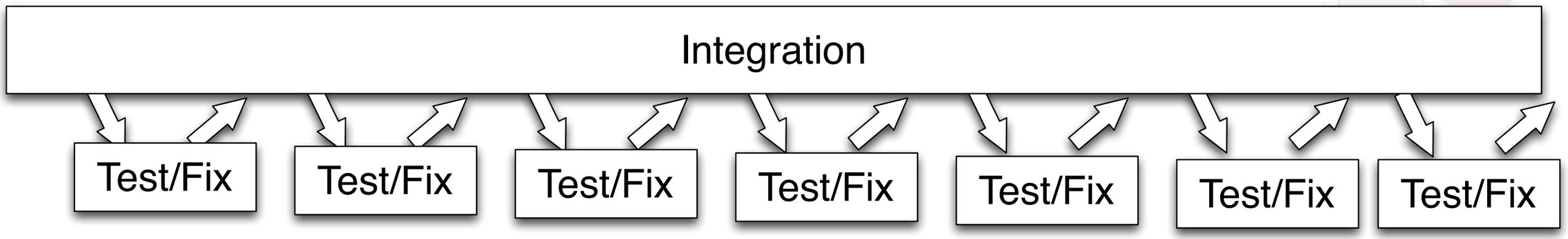

NORDSTROM
INNOVATION LAB

- SMALL DESIGN PROCESS
- ITERATING

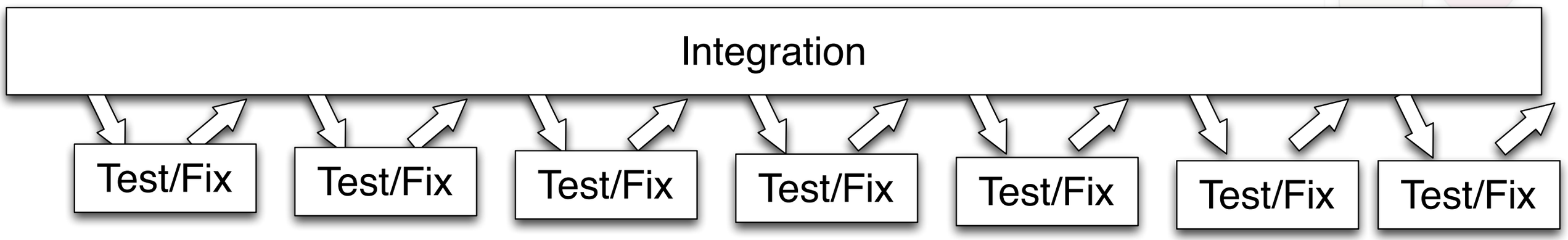
- POST-IT NOTES
- ROLE BREAKDOWN
  - CLEAR
- COLOR CODED
- GLASS WALLS
- NO COMMUNICATION TROUBLE



Handling Complexity: Separating out the details is not trivial

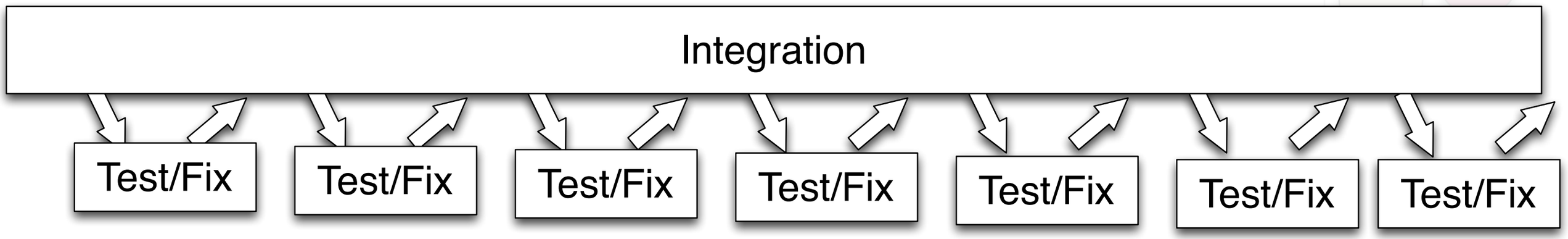


Handling Complexity: Separating out the details is not trivial



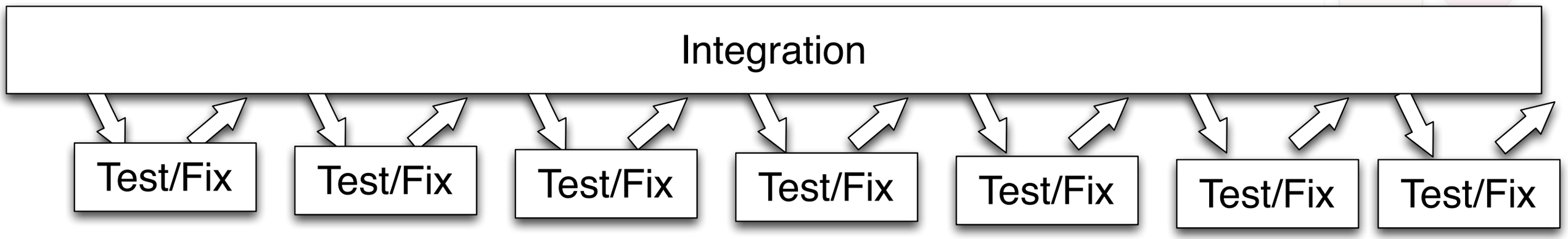
- Seemingly “simple” Test/Fix and Integrate steps:

Handling Complexity: Separating out the details is not trivial



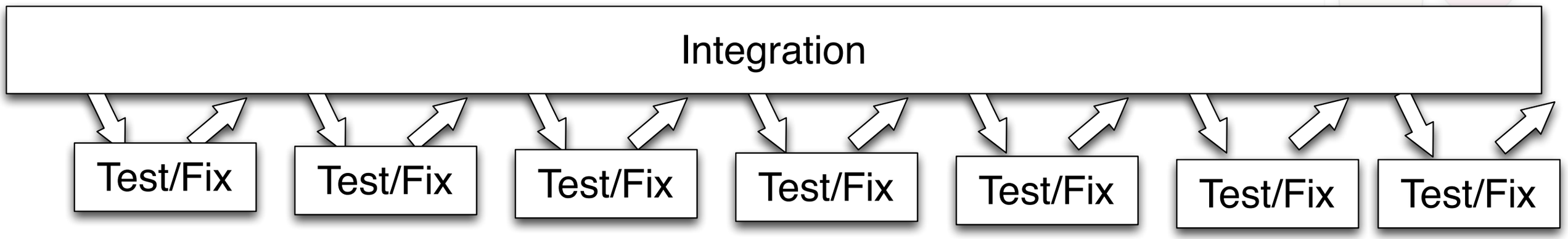
- Seemingly “simple” Test/Fix and Integrate steps:
- Should there be separate & independent test group?

Handling Complexity: Separating out the details is not trivial



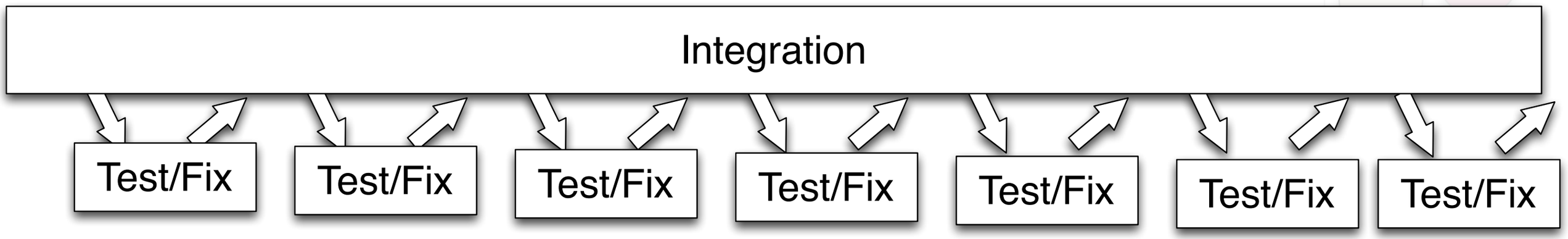
- Seemingly “simple” Test/Fix and Integrate steps:
- Should there be separate & independent test group?
- How should problem be reported and to whom?

Handling Complexity: Separating out the details is not trivial



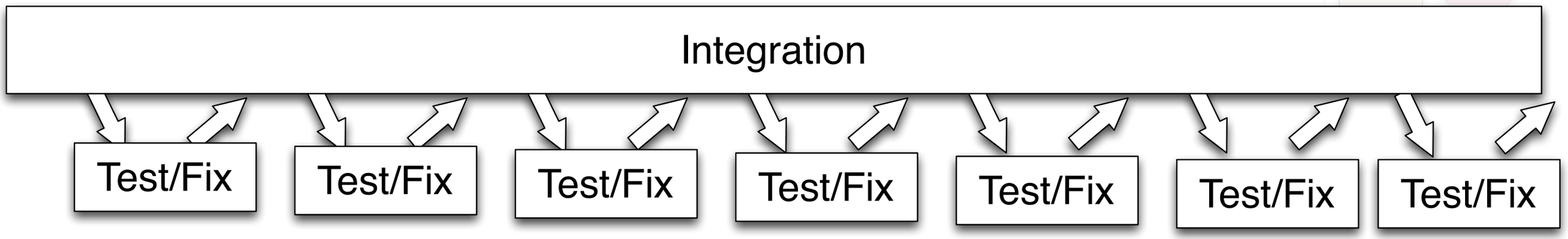
- Seemingly “simple” Test/Fix and Integrate steps:
- Should there be separate & independent test group?
- How should problem be reported and to whom?
- How much information must accompany a problem report?

Handling Complexity: Separating out the details is not trivial



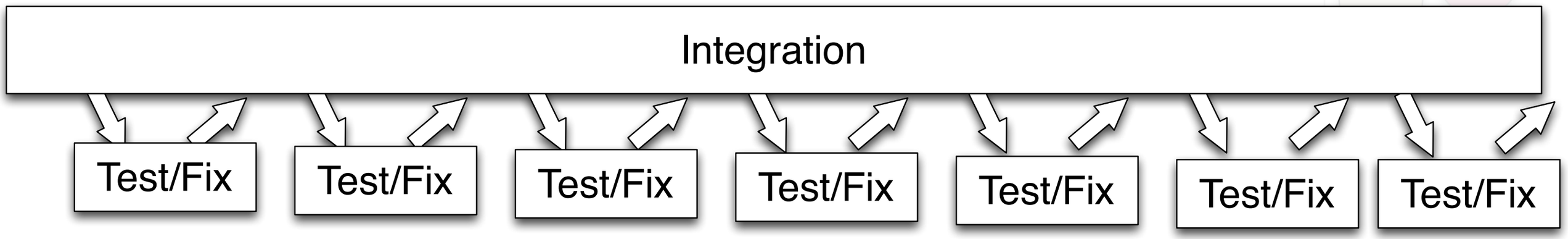
- Seemingly “simple” Test/Fix and Integrate steps:
- Should there be separate & independent test group?
- How should problem be reported and to whom?
- How much information must accompany a problem report?
- Who decides on the priority of the problem?

Handling Complexity: Separating out the details is not trivial



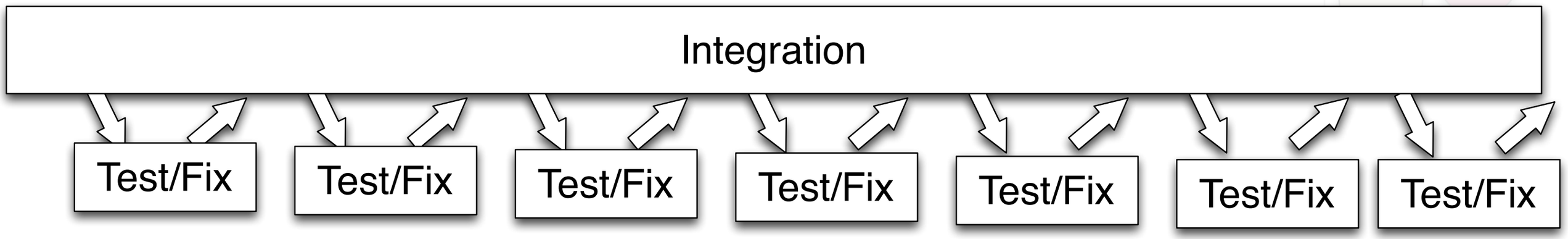
- Seemingly “simple” Test/Fix and Integrate steps:
- Should there be separate & independent test group?
- How should problem be reported and to whom?
- How much information must accompany a problem report?
- Who decides on the priority of the problem?
- How is the problem fix returned?

Handling Complexity: Separating out the details is not trivial



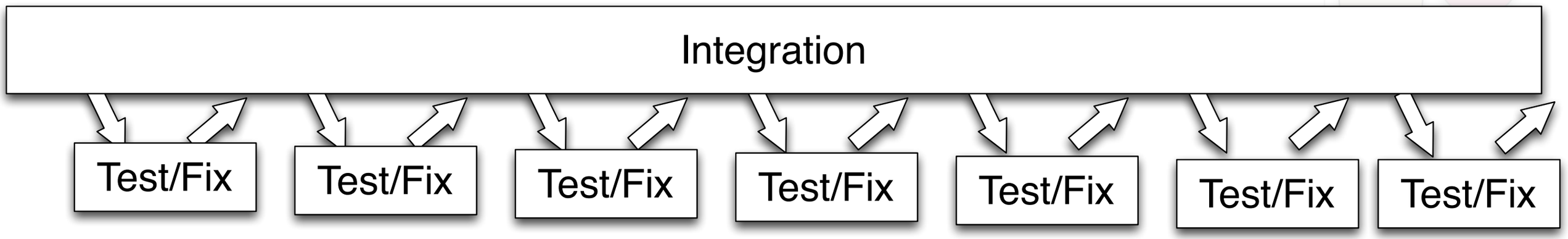
- Seemingly “simple” Test/Fix and Integrate steps:
  - Should there be separate & independent test group?
  - How should problem be reported and to whom?
  - How much information must accompany a problem report?
  - Who decides on the priority of the problem?
  - How is the problem fix returned?
  - Should all problems be fixed?

Handling Complexity: Separating out the details is not trivial



- Seemingly “simple” Test/Fix and Integrate steps:
  - Should there be separate & independent test group?
  - How should problem be reported and to whom?
  - How much information must accompany a problem report?
  - Who decides on the priority of the problem?
  - How is the problem fix returned?
  - Should all problems be fixed?
  - What should we do with non-fixed problem?

Handling Complexity: Separating out the details is not trivial



- Seemingly “simple” Test/Fix and Integrate steps:
  - Should there be separate & independent test group?
  - How should problem be reported and to whom?
  - How much information must accompany a problem report?
  - Who decides on the priority of the problem?
  - How is the problem fix returned?
  - Should all problems be fixed?
  - What should we do with non-fixed problem?
  - How are fixes integrated back to the system

Building a System

Non-Technical Considerations for Developing and Supporting a System

Building a System

Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion

Building a System

Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion
- How does one estimate and handle this?

Building a System

Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion
- How does one estimate and handle this?

Building a System

Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion
  - How does one estimate and handle this?
- Assignment and Communications Expansion?

Building a System

Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion
  - How does one estimate and handle this?
- Assignment and Communications Expansion?
  - Do we need some process?

Building a System

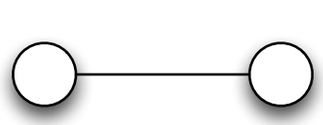
Non-Technical Considerations for Developing and Supporting a System

- Effort & Schedule Expansion
  - How does one estimate and handle this?
- Assignment and Communications Expansion?
  - Do we need some process?
  - Do we need some tools?

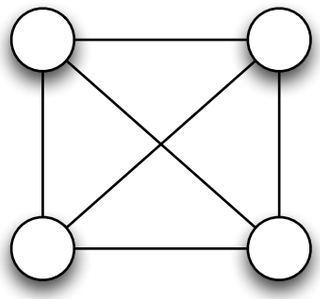
Building a system

Increased complexity means increased human resources

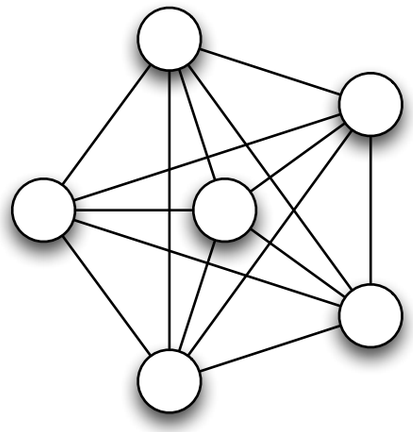
$$\frac{(n)(n-1)}{2}$$



2 people
1 path



4 people
6 paths



6 people
15 paths

Consider communication errors as well

Building a system

A Large, Complex System

- Building “Mission critical” or “Business critical” system (e.g. payroll) requires (1) several separate activities performed by (2)more than 1 person (e.g. 50 ~ 100):
  - Requirements: gathering, analysis, specification, and agreement
  - Design: abstraction, decomposition, cohesion, interaction and coupling analysis
  - Implementation: coding and unit testing
  - Integration and tracking of pieces and parts
  - Separate testing: functional testing, component testing, system testing, and performance testing
  - Packaging and releasing the system

Building a system

Need to support the system (for real production)

- Pre-release: preparation for education & support:
  - Number of expected users
  - Number of “known problems” and expected quality
  - Amount of user and support personnel training
  - number of fix and maintenance cycle

- Post-release: preparation for user and customer support:
  - Call center and problem resolutions
  - Major problem fixes and code changes
  - Functional modifications and enhancements

https://flic.kr/p/eeQ512

https://flic.kr/p/dx3QAu

Building a system

Need to support the system (for real production)

- Pre-release: preparation for education & support:
  - Number of expected users
  - Number of “known problems” and expected quality
  - Amount of user and support personnel training
  - number of fix and maintenance cycle

- Post-release: preparation for user and customer support:
  - Call center and problem resolutions
  - Major problem fixes and code changes
  - Functional modifications and enhancements

https://flic.kr/p/eeQ512

https://flic.kr/p/dx3QAu

Building a system

Need to support the system (for real production)

- Pre-release: preparation for education & support:
  - Number of expected users
  - Number of “known problems” and expected quality
  - Amount of user and support personnel training
  - number of fix and maintenance cycle

- Post-release: preparation for user and customer support:
  - Call center and problem resolutions
  - Major problem fixes and code changes
  - Functional modifications and enhancements

https://flic.kr/p/eeQ512

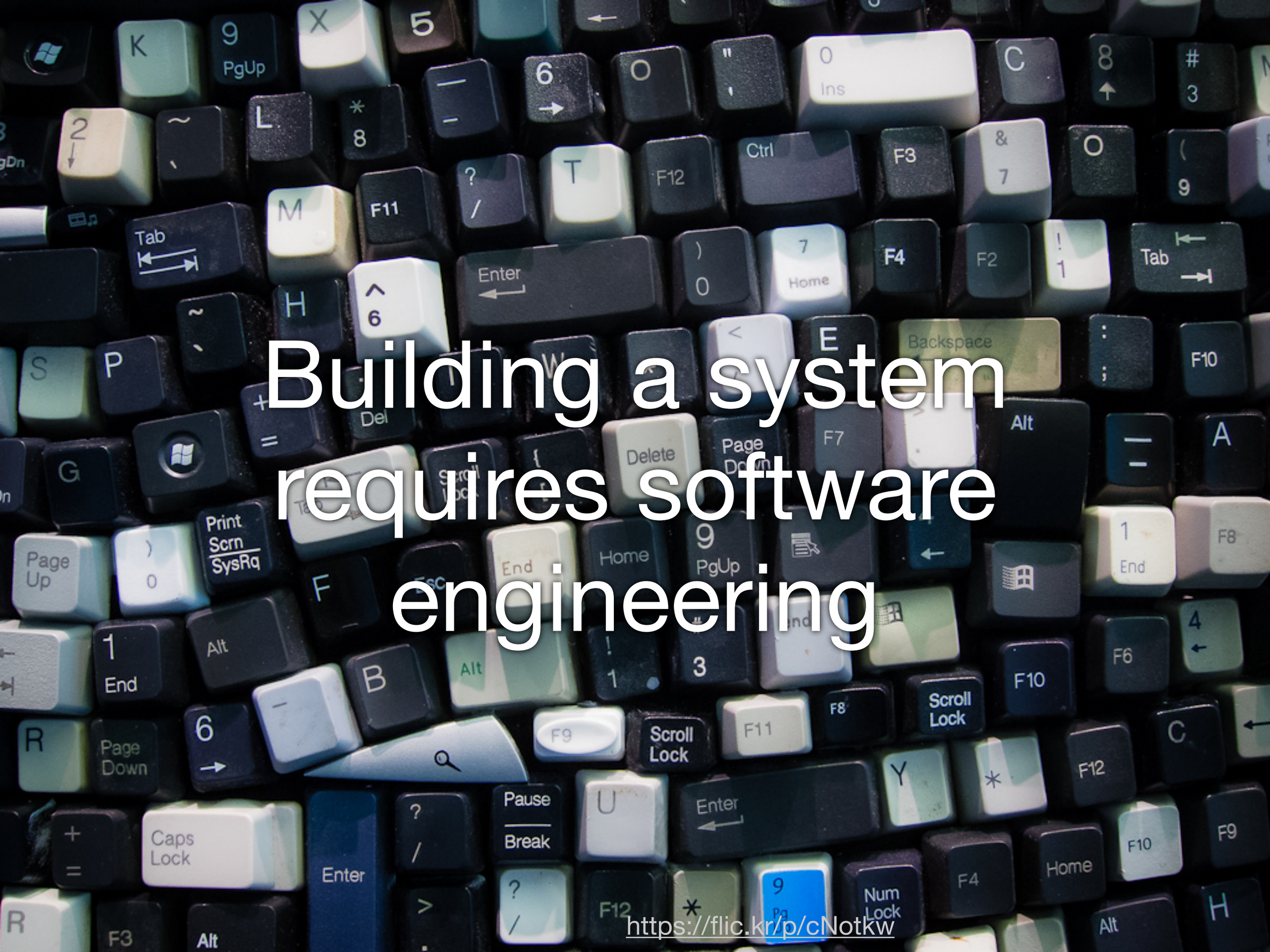
https://flic.kr/p/dx3QAu

Building a system

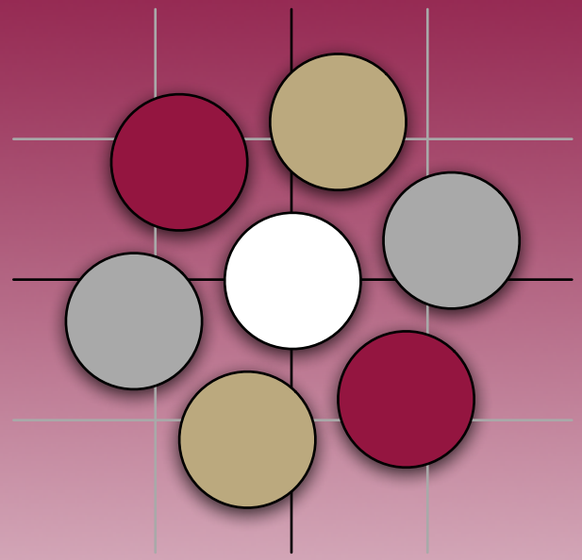
Coordination Efforts Required in Systems Development and Support

- Because there are
  - more parts,
  - more developers
  - more users to consider in “Large Systems” than a single program developed by a single person for a limited number of users, there is the need for Coordination of (3P’s):
    - ‘Processes’ and methodologies to be used
    - Final ‘product’ and intermediate artifacts
    - ‘People’ (developers, support personnel, and users)

Building a system requires software engineering



https://flic.kr/p/cNotkw



WESTMONT COMPUTER SCIENCE