

XP and Design

IndicThreads.com Conference On Java Technology

PUNE, INDIA

25

26

27

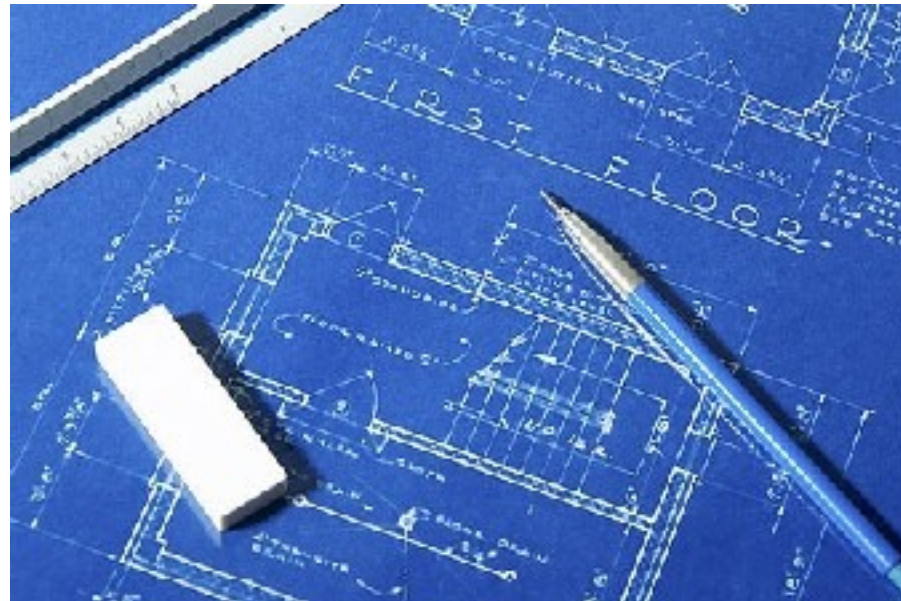
Nov. 2008

Paulo Caroli & Sudhindra Rao

ThoughtWorks

XP and Design

- Where did the Design phase go?



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

About us

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

About us

14 + 6

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

About us

Certified Architect

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

About us

Agile Coach / Developer

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Agenda

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Agenda

- XP
- Traditional Methods and XP
- XP and Design
 - Simple Design
 - Testing
 - Refactoring
 - Continuous Integration
- Conclusion
- Q&A

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

eXtreme Programming

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP

- Lightweight methodology
- Kent Beck, late 90s
- 4 Values and 12 Practices (1st edition)
- Focused on the developers

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP Values

- Courage
- Simplicity
- Communication
- Feedback

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP Practices

- The Planning Game
- Whole Team
- Testing
- Short Releases
- Continuous Integration
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- Refactoring
- Simple Design

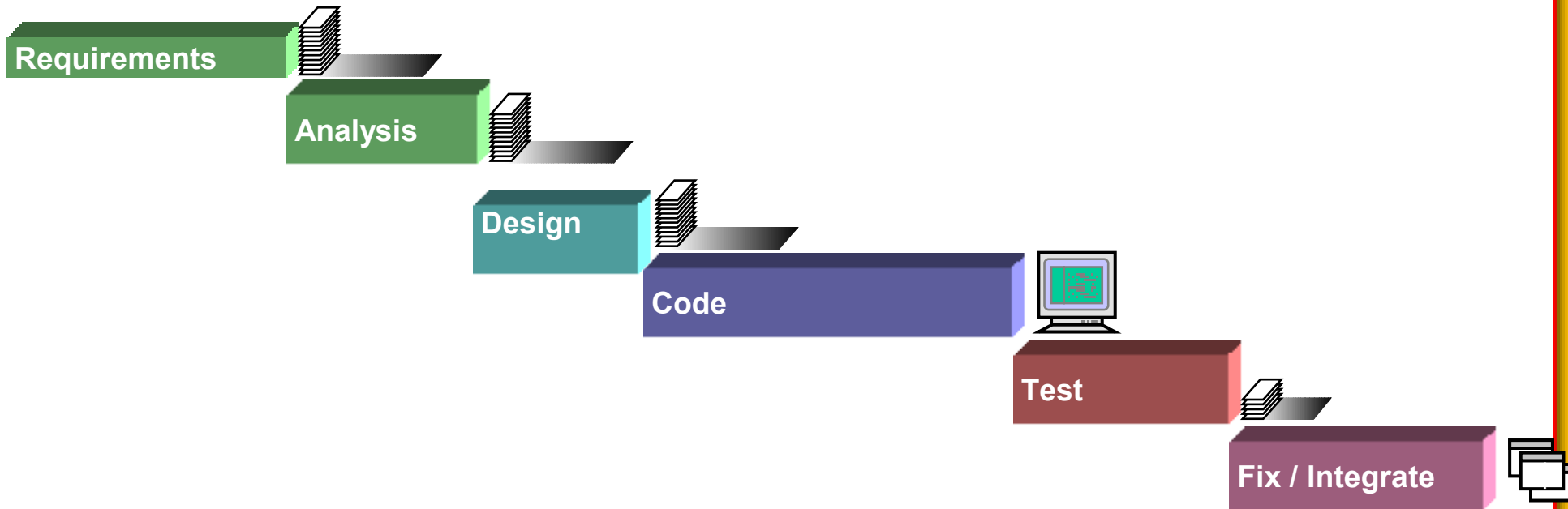
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Traditional Methods

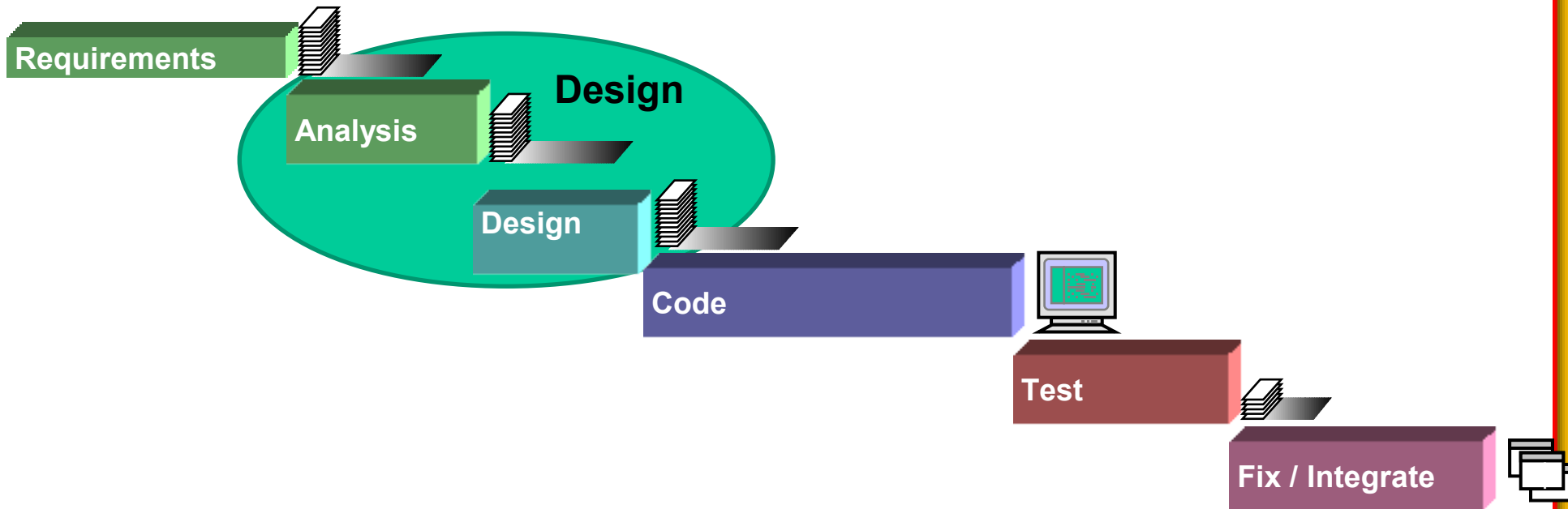
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



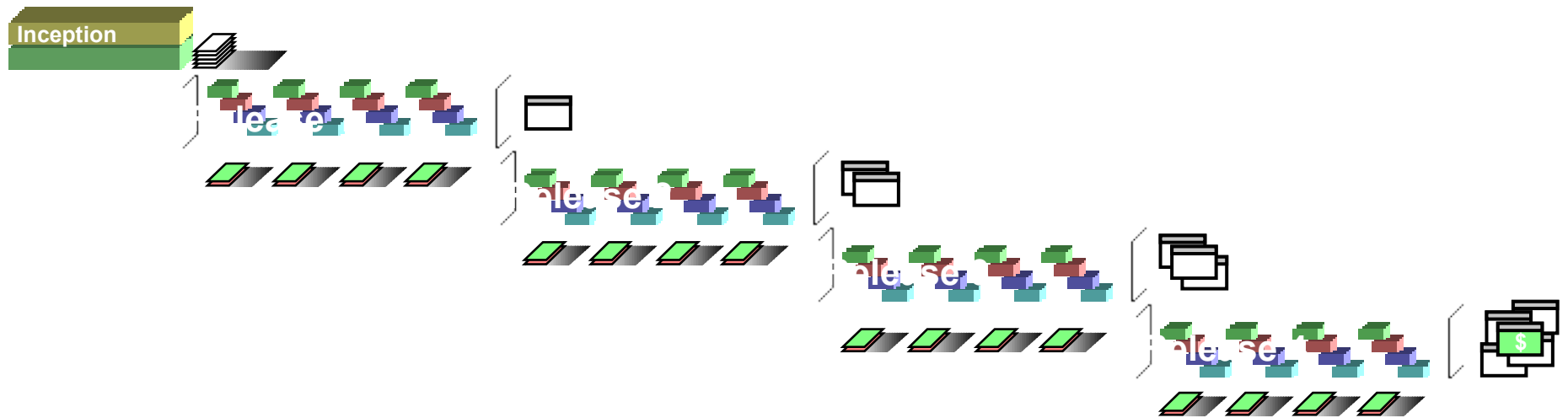
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Typical XP Project

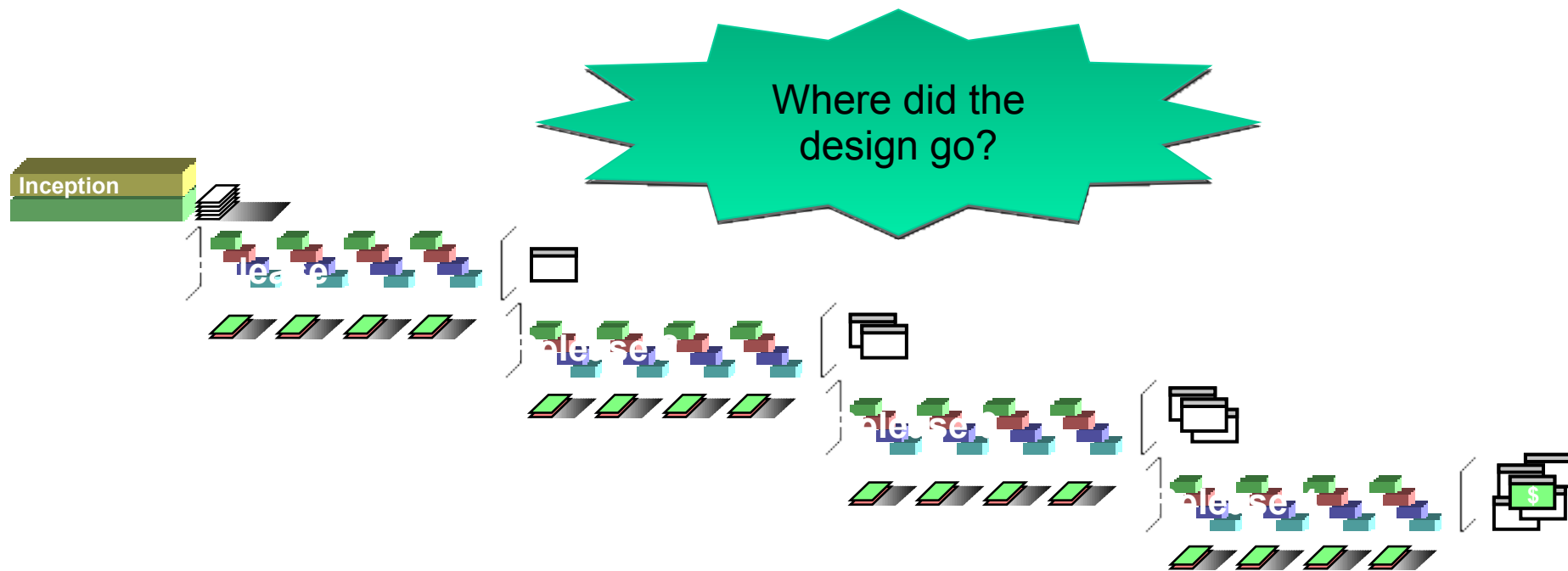
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



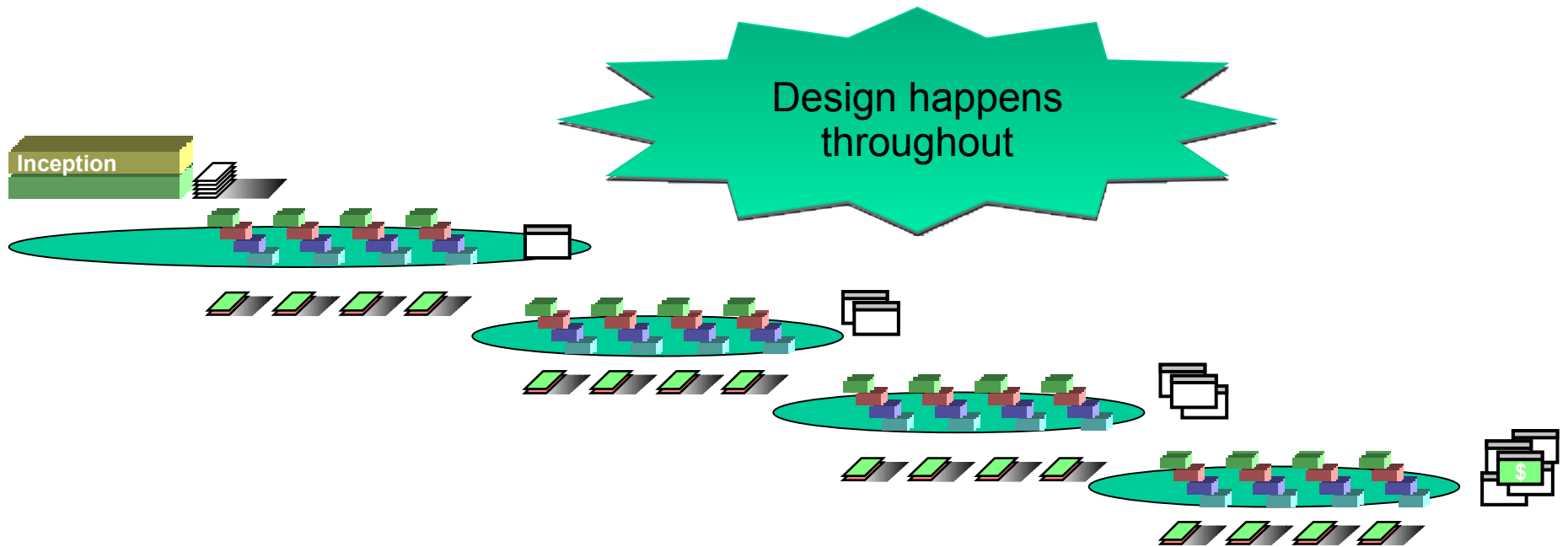
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



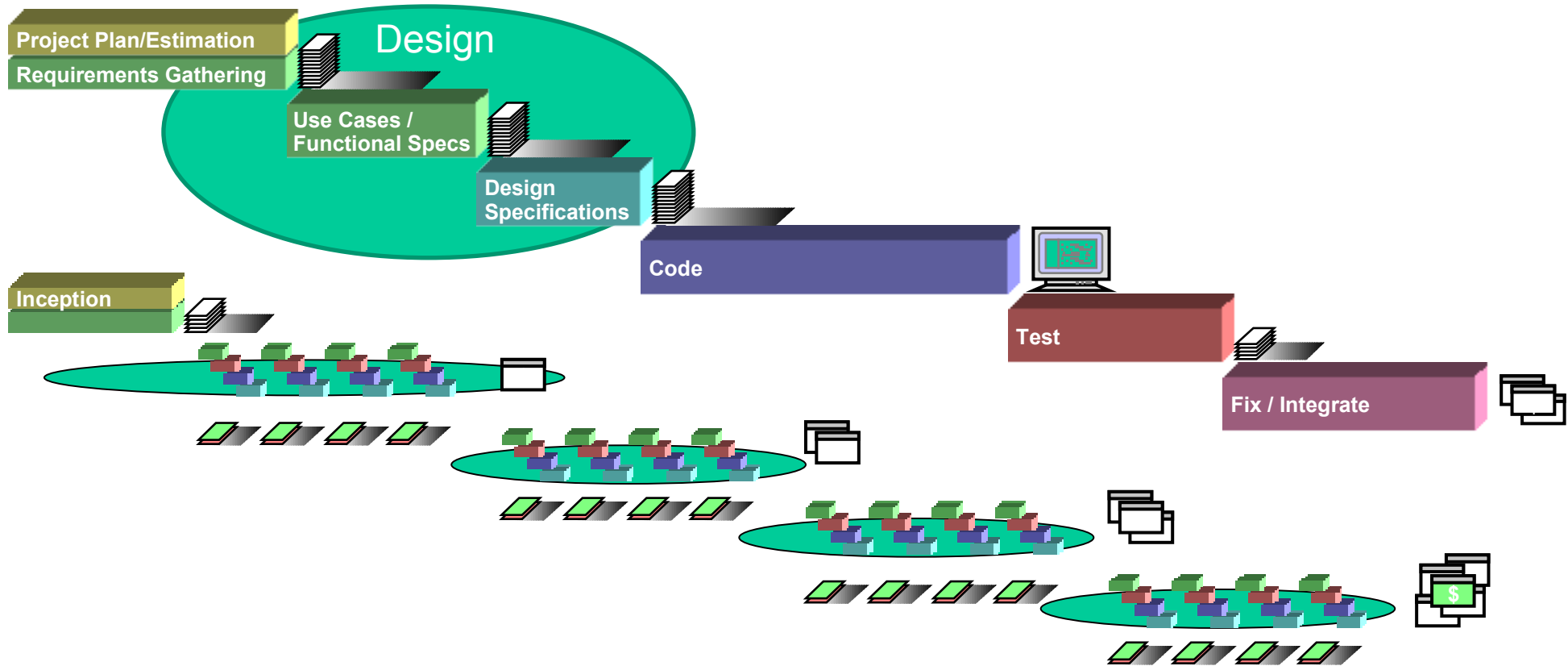
IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Traditional Methods and XP

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP Practices

- The Planning Game
- Whole Team
- Testing
- Short Releases
- Continuous Integration
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- Refactoring
- Simple Design

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- **Simple Design**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

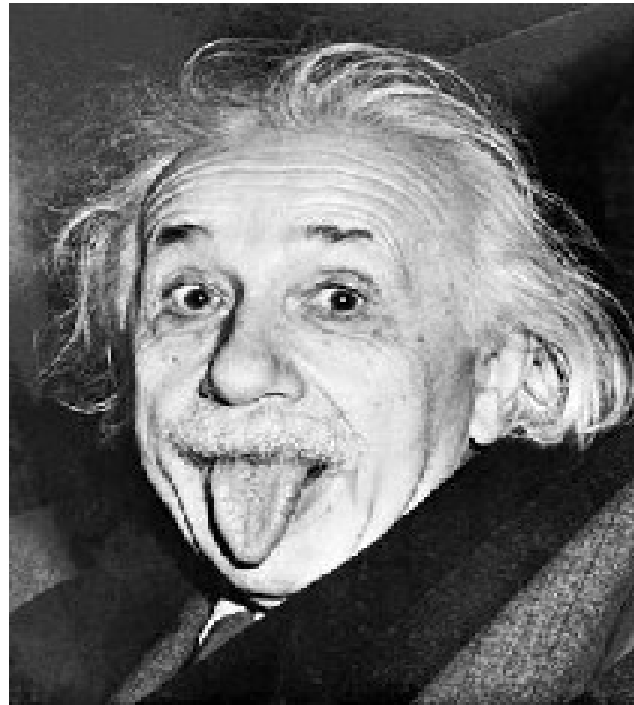
XP and Design

- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- **Simple Design**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

*"Things should be made as simple as possible,
but not any simpler."*



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- Don't try to solve **big problems** in one go
- Break them down into **little problems** and solve them one at a time
 - Produce **simple artifacts** that could be **used flexibly** instead of complex artifacts that aim to embody all possibilities

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- Never add functionality **before** it is needed
 - Think about tomorrow, but design, test and code for **today's need**
 - Don't design for **future complexity** that may not happen
 - Do the **simplest thing** that meets the **current requirement**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- Do the **simplest** thing that could **possibly work**
 - A simple design **takes less time** to finish
 - A simple design means that **changing it later** doesn't have a huge impact

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- **Simple Design**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP Testing

- Automated Tests
- Acceptance Tests
- Test Driven Development

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Automated Tests

- The automated test suite provides **feedback** from the system.
- Testing not only serves finding defects.
 - Its primary goal is **avoiding defects**.

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Acceptance Tests

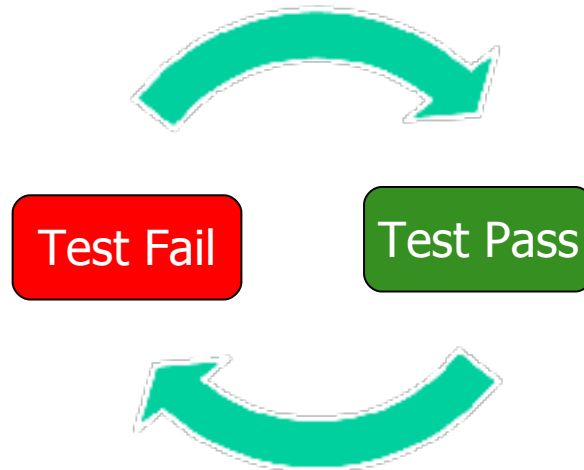
- Acceptance tests are based on the **requirements** given by the customer
 - Some recommend having acceptance tests instead of requirements.
- You can be uncertain whether what your system does is what your customer expects out of it.
 - Acceptance tests **eliminate** this **uncertainty**.
- Developer Test and Customer Test

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Test-Driven Development

Test-Driven Development (TDD)
is a design approach!



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Test-Driven Development

elegantly meet the
requirements

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Test-Driven Development

current needs

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

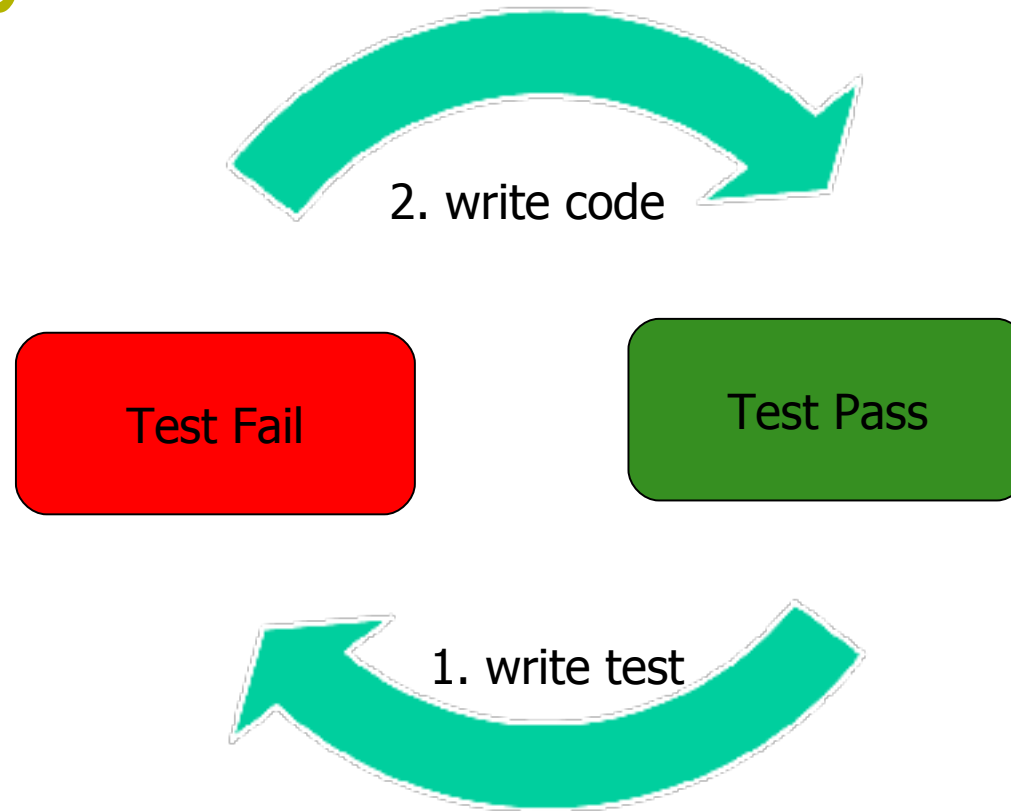
Test-Driven Development

design evolves

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

TDD Cycle



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

TDD Step 1 - Test Code

- Testable code only!
- Precise specification
- Helps to think about **design**
- Guarantees tangible functional code

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

TDD Step 2 - Functional Code

- Meet the requirement (test code)
- Simplest solution that works
- Leave improvements for a later step
- Avoid writing any further (untested) code

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

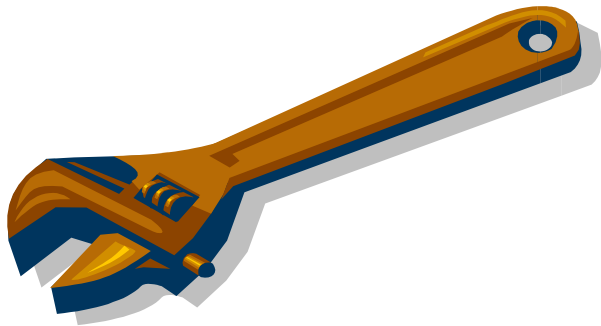
- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- Simple Design

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Refactoring

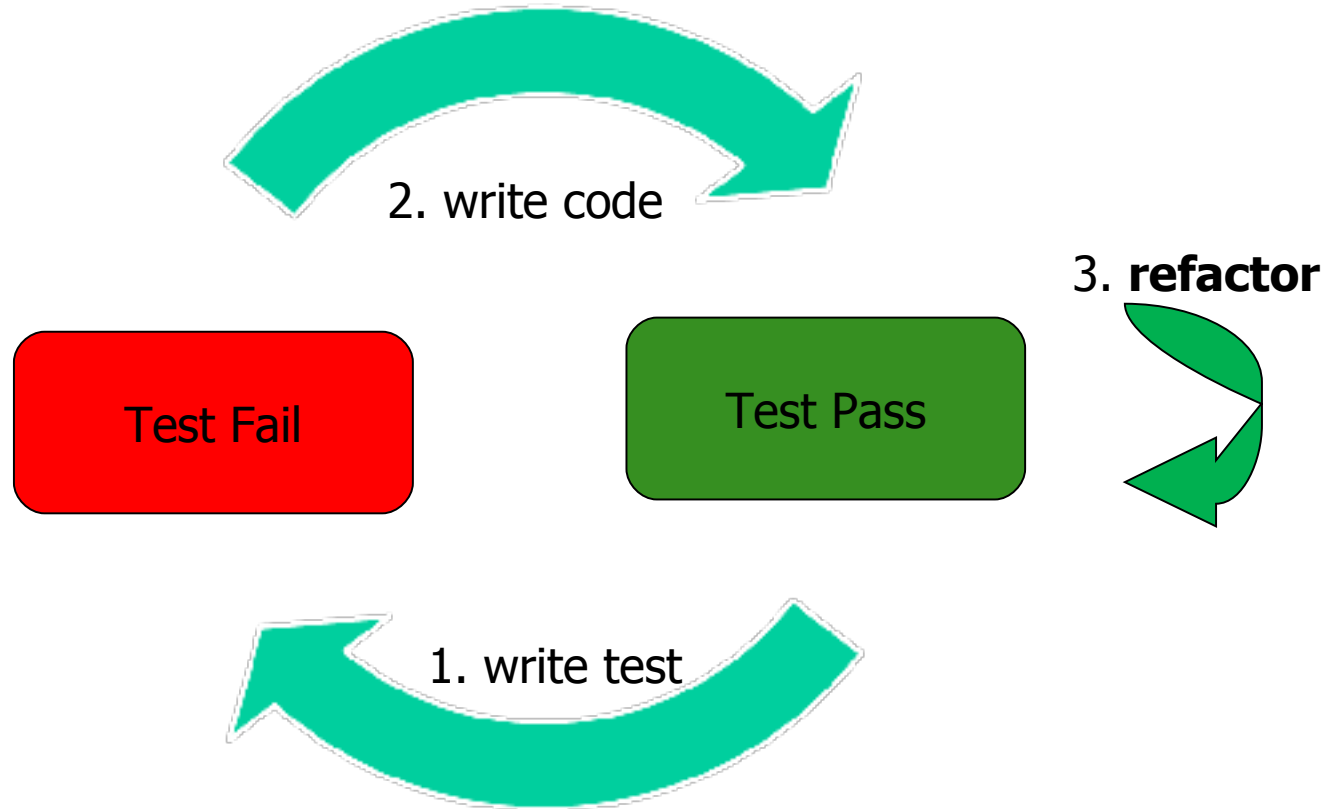
Refactoring is a development practice for **restructuring** an existing code, altering its internal structure to make it more simple, without changing its external behavior



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

A Critical Step



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

TDD Step 3 - Refactoring

- Clean up the code (test and functional)
- Make sure that the code expresses intent
- Re-think the **design**
- Look for Code Smell (improve the design)
- Avoid Broken Windows!

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- When things get complicated
 - Invest the time to **refactor** it so that it is **simple** again
 - If the design is necessarily complicated, now is a good time to document it thoroughly.

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

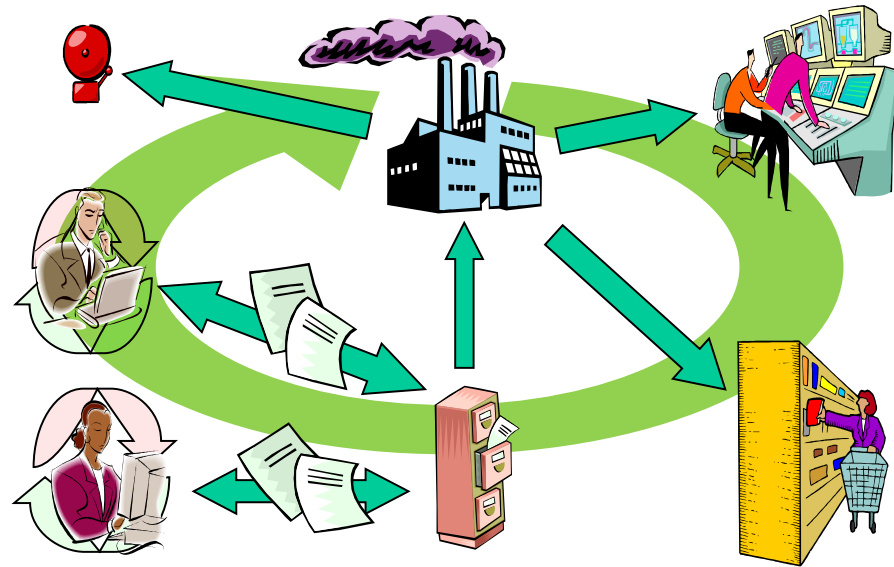
- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- **Simple Design**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Continuous Integration

- The software development practice where members of a team integrate their work frequently



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- No feature is complete until the suite of **acceptance tests** that define it are passing.
- **Automated tests** ensure that your implementation matches your design
 - Everybody can then know what will be affected when the **design** changes
 - The tests form much of your detailed design documentation

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

Conclusion

IndicThreads.com Conference On Java Technology 2008

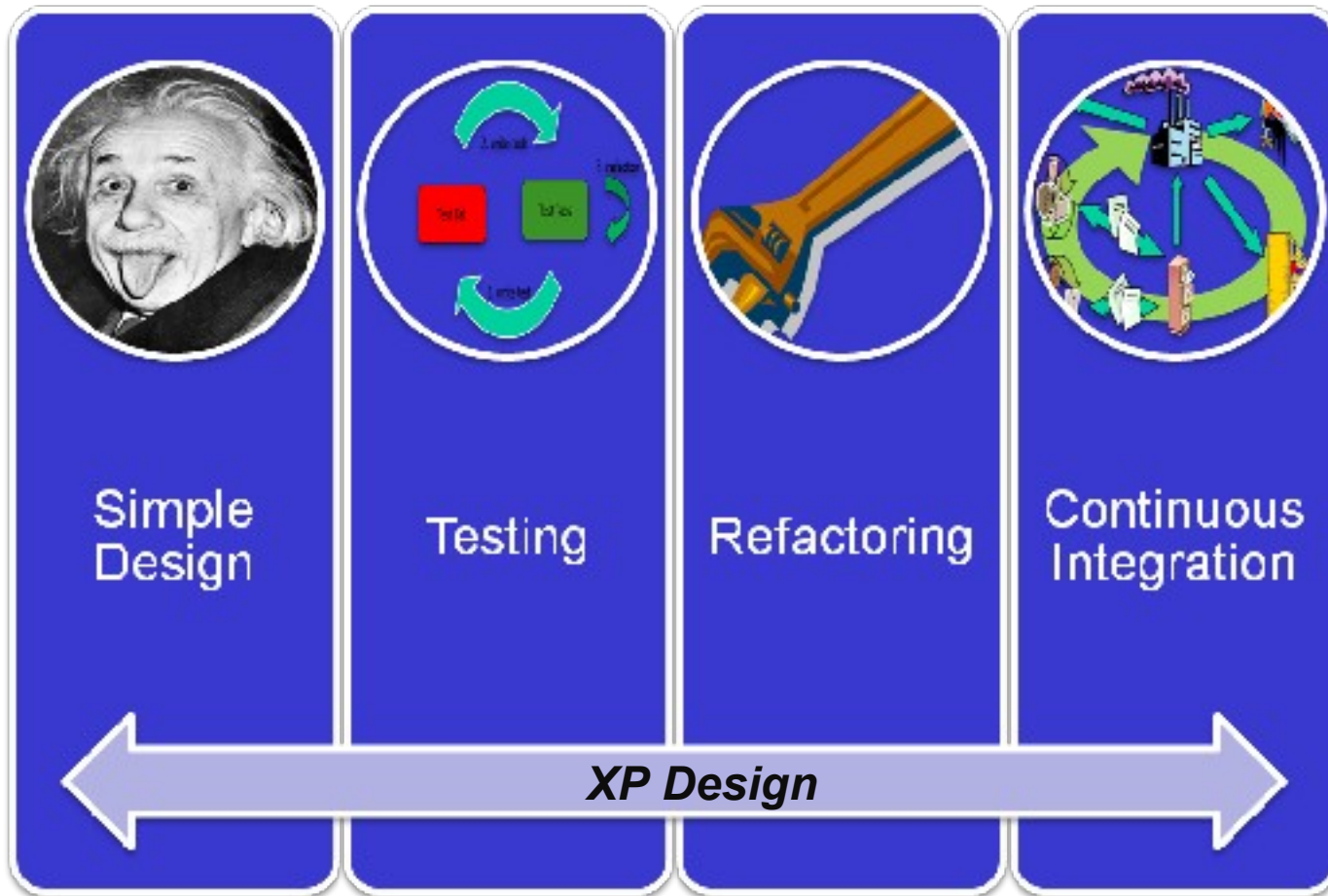
PUNE, INDIA

XP and Design

- The Planning Game
- Whole Team
- **Testing**
- Short Releases
- **Continuous Integration**
- Collective Code Ownership
- Coding Standard
- Metaphor
- Sustainable Pace
- Pair Programming
- **Refactoring**
- **Simple Design**

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

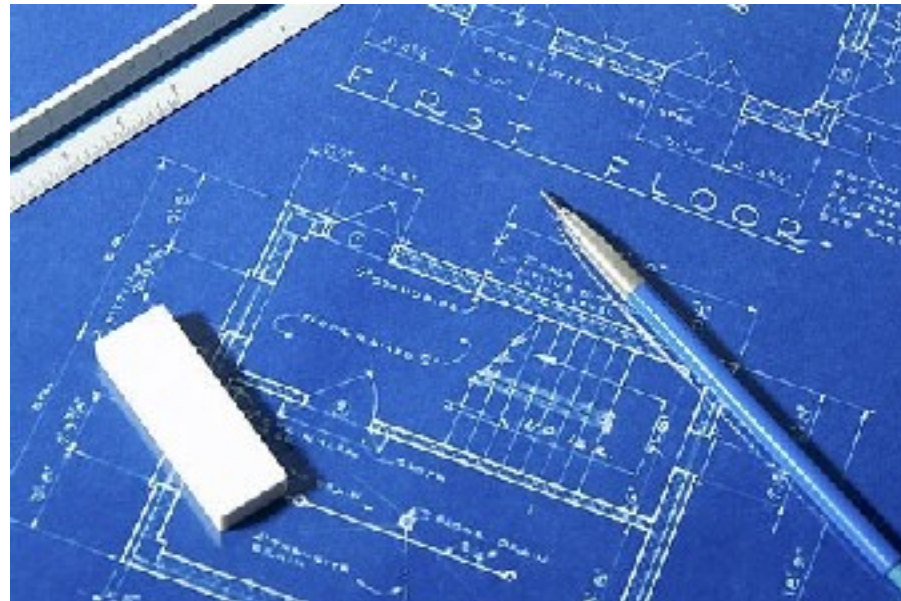


IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- Where did the Design phase go?



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

- Design is so important that we do it everyday!



IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA

XP and Design

Thanks for attending!
Questions?

Paulo Caroli & Sudhindra Rao

pcaroli@thoughtworks.com

www.caroli.org

srao@thoughtworks.com

IndicThreads.com Conference On Java Technology 2008

PUNE, INDIA