Agile Methodologies

Lecture#15

Advantages of Scrum

- The product is broken down into a set of manageable and understandable chunks.
- Unstable requirements do not hold up progress.
- The whole team has visibility of everything and consequently team communication is improved.
- Customers see on-time delivery of increments and gain feedback on how the product works.
- Trust between customers and developers is established and a positive culture is created in which everyone expects the project to succeed.

Disadvantages of Scrum

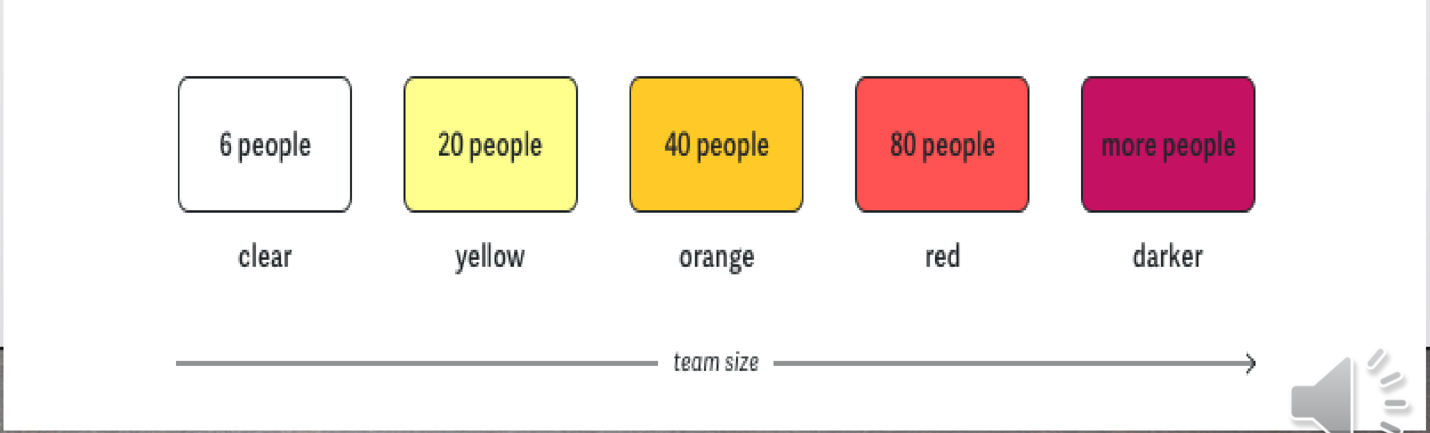
- Scrum often leads to scope creep, due to the lack of a definite end-date
- The chances of project failure are high if individuals aren't very committed or cooperative
- Adopting the Scrum framework in large teams is challenging
- The framework can be successful only with experienced team members
- Daily meetings sometimes frustrate team members

Crystal

- Crystal method is an agile software development approach that focuses primarily on people and their interactions when working on a project rather than on processes and tools.
- Crystal Method is based on **two fundamental assumptions**:
  - Teams can find ways on their own to improve and optimize their workflows
  - Every project is unique and always changing, which is why that project's team is best suited to determine how it will tackle the work

Crystal Family Members

- The project properties changed depending on the number of the people involved in the project and the level of criticality of the project at hand.
- Which approach will be most suitable for your projects depends on three dimensions:
    - Team Size
    - Criticality
    - What the priority of the project is
- They are characterized by colors, according to the number of people involved in the project:
- **Clear** - for teams of 8 or fewer people
- **Yellow** - for teams of 10-20 people
- **Orange** - for teams of 20-50 people
- **Red** - for teams of 50-100 people



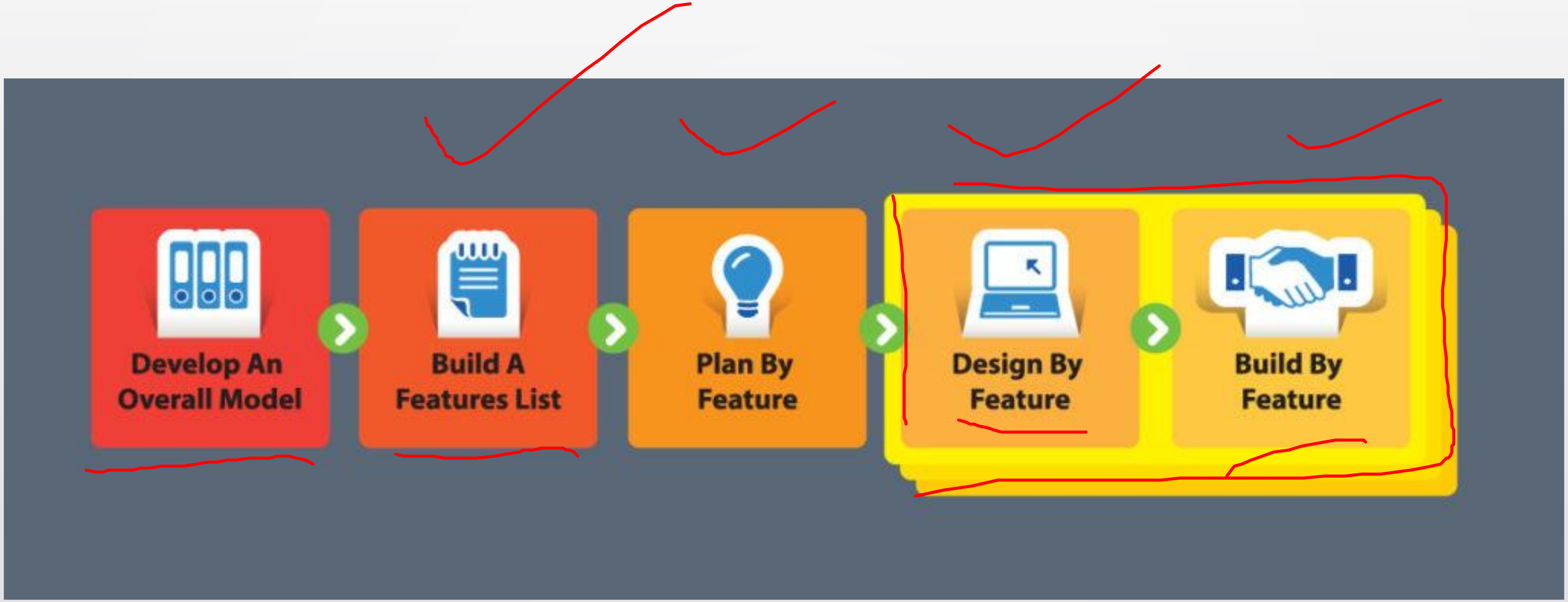
8 Properties of Crystal

- Frequent Delivery
- Reflective Improvement
- Osmotic Communication
- Personal Safety
- Focus
- Easy access to expert users
- Technical Environment
- Frequent Integration

Feature Driven Development (FDD)

- Feature-Driven Development (FDD) is a client-centric, architecture-centric, and pragmatic software process.
- A feature is a small, client-valued function expressed in the form . For example, "Calculate the total of a sale", "Validate the password of a user“.
- If one working on a large-scale software project, FDD might be right for project. But this methodology relies heavily on chief developers and has a top-down decision-making approach, as opposed to some of the other agile frameworks (such as XP) that are based more on collective project ownership. If that type of methodology fits your company’s culture, then Feature Driven Development is worth investigating.

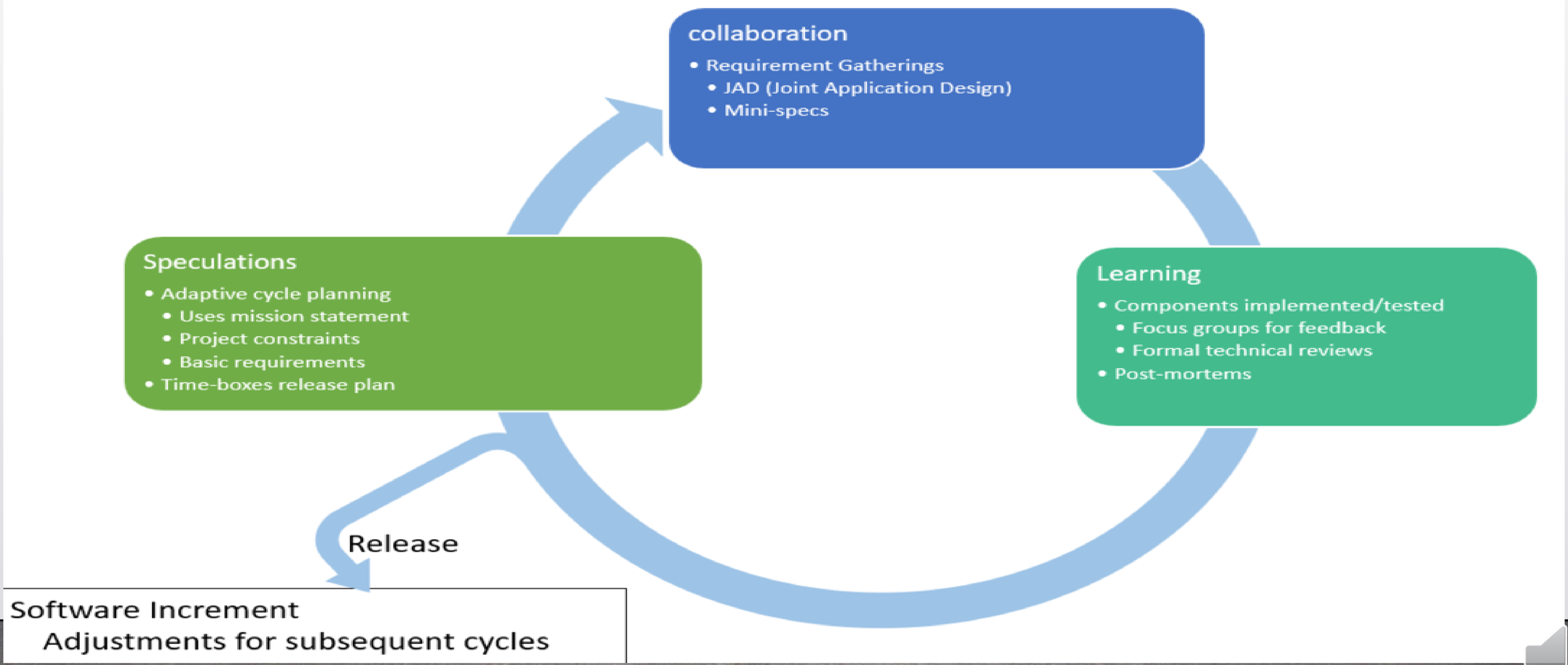
Feature Driven Development (FDD)



Adaptive Software Development (ASD)

- It is a method to build complex software and system.
- ASD focuses on human collaboration and self-organization.
- Adaptive Software Development has evolved from RAD practices.
- It aims to enable teams to quickly and effectively adapt to changing requirements or market needs by evolving their products with lightweight planning and continuous learning.

Adaptive Software Development (ASD)



collaboration

- Requirement Gatherings
  - JAD (Joint Application Design)
  - Mini-specs

Speculations

- Adaptive cycle planning
  - Uses mission statement
  - Project constraints
  - Basic requirements
- Time-boxes release plan

Learning

- Components implemented/tested
  - Focus groups for feedback
  - Formal technical reviews
- Post-mortems

Release

Software Increment
Adjustments for subsequent cycles