COURSE NAME : 19CST201-Agile Software Engineering

III YEAR/ V SEMESTER

Topic: Agile Process Model

Ms.G.Swathi
Assistant Professor
Department of Computer Science and Engineering

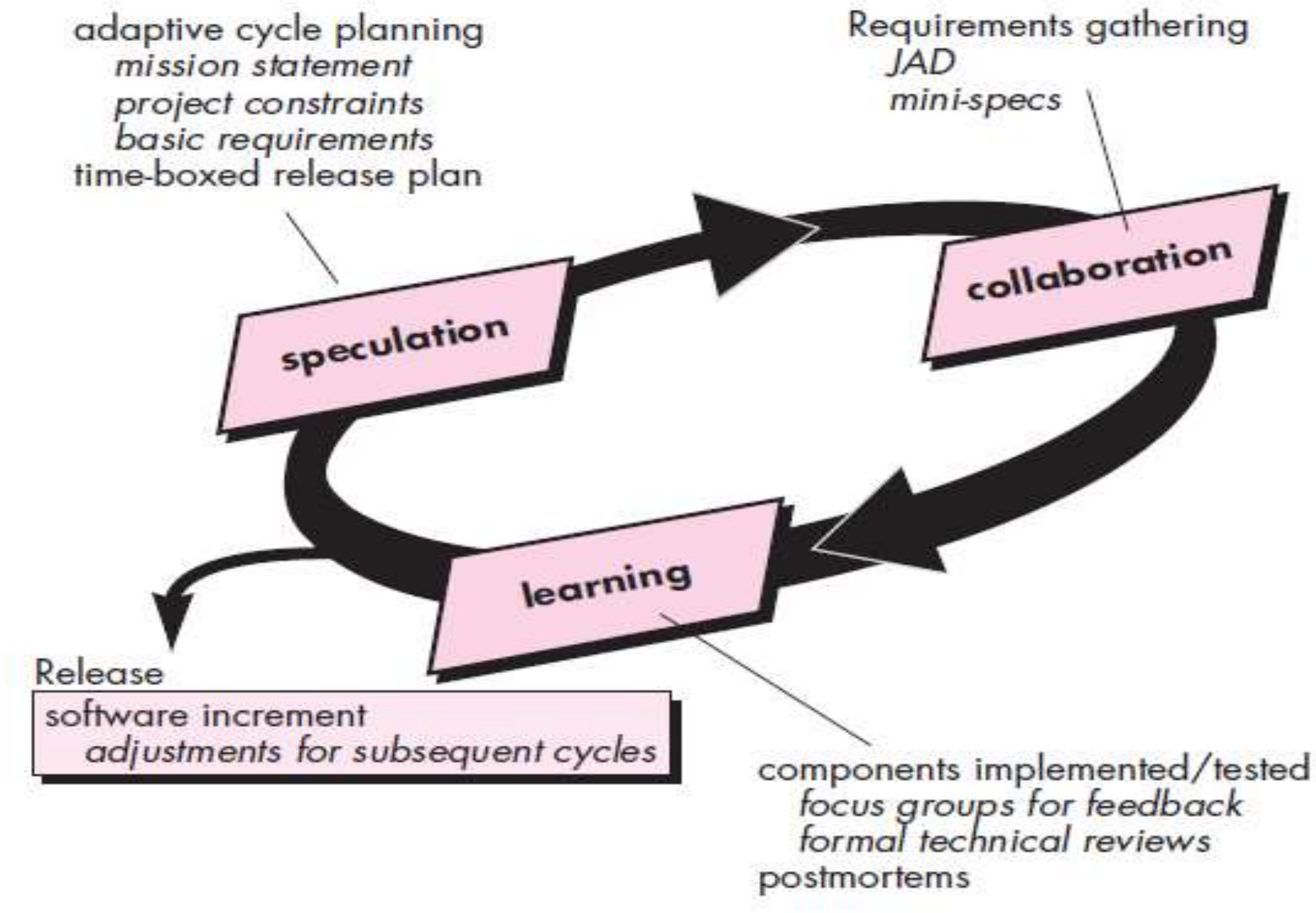
Agile Process Models

1. Adaptive Software Development (ASD)
2. Scrum
3. Dynamic Systems Development Method (DSDM)
4. Crystal
5. Feature Drive Development (FDD)
6. Lean Software Development (LSD)
7. Agile Modeling (AM)
8. Agile Unified Process (AUP)

Adaptive Software Development (ASD)

- proposed by Jim Highsmith
- technique for building complex software and systems.
- ASD “life cycle” incorporates three phases,
  - speculation,
  - Collaboration
  - learning.

Adaptive Software Development (ASD)



adaptive cycle planning
mission statement
project constraints
basic requirements
time-boxed release plan

Requirements gathering
JAD
mini-specs

Release
software increment
adjustments for subsequent cycles

components implemented/tested
focus groups for feedback
formal technical reviews
postmortems

Adaptive Software Development (ASD)

During *speculation*,

- the project is initiated and ***adaptive cycle planning*** is conducted.
- Adaptive cycle planning uses project initiation information—the customer's
- **mission statement, project constraints** (e.g., delivery dates or user descriptions), and
- **basic requirements**—to define the set of release cycles

Adaptive Software Development (ASD)

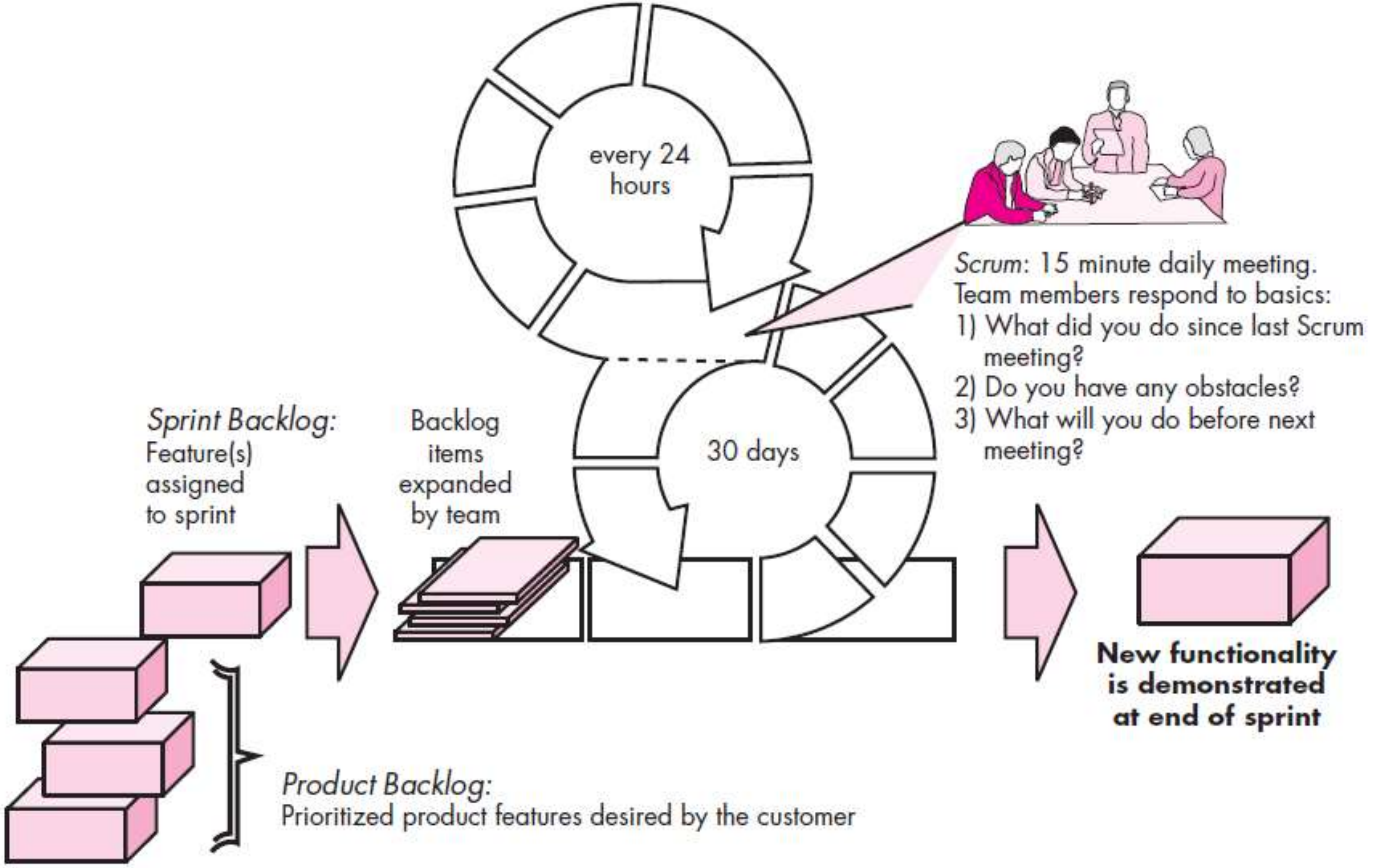
During *collaboration*

- communication and teamwork, but it also emphasizes individualism

Learning

- learning will help them to improve their level of real understanding.
- ASD teams learn in three ways: focus groups ,technical reviews and projects management.

SCRUM


every 24 hours
30 days
Sprint Backlog: Feature(s) assigned to sprint
Backlog items expanded by team
New functionality is demonstrated at end of sprint
Product Backlog: Prioritized product features desired by the customer

Scrum: 15 minute daily meeting.
Team members respond to basics:
1) What did you do since last Scrum meeting?
2) Do you have any obstacles?
3) What will you do before next meeting?

SCRUM

- Scrum principles are consistent with the agile manifesto and are used to guide development activities within a process that incorporates the following framework activities:
- Requirements
- Analysis
- Design
- Evolution
- Delivery.

SCRUM

- *Scrum meetings*—are short (typically 15 minutes) meetings held daily by the Scrum team.
- Three key questions are asked and answered by all team members
  - ➢ What did you do since the last team meeting?
  - ➢ What obstacles are you encountering?
  - ➢ What do you plan to accomplish by the next team meeting?

Dynamic Systems Development Method (DSDM)

- The *Dynamic Systems Development Method* (DSDM) is an agile software development approach that "provides a framework for building and maintaining systems which meet tight time constraints through the use of incremental prototyping in a controlled project environment"

Dynamic Systems Development Method (DSDM)

DSDM life cycle that defines three different iterative cycles, preceded by two additional life cycle activities:

- **Feasibility study** - basic business requirements and constraints
- **Business study** - the functional and information requirements that will allow the application to provide business value
- **Functional model iteration** - set of incremental prototypes
- **Design and build iteration** - revisits prototypes built during functional model iteration to ensure that each has been engineered in a manner that will enable it to provide operational business value for end users.
- **Implementation**

Crystal

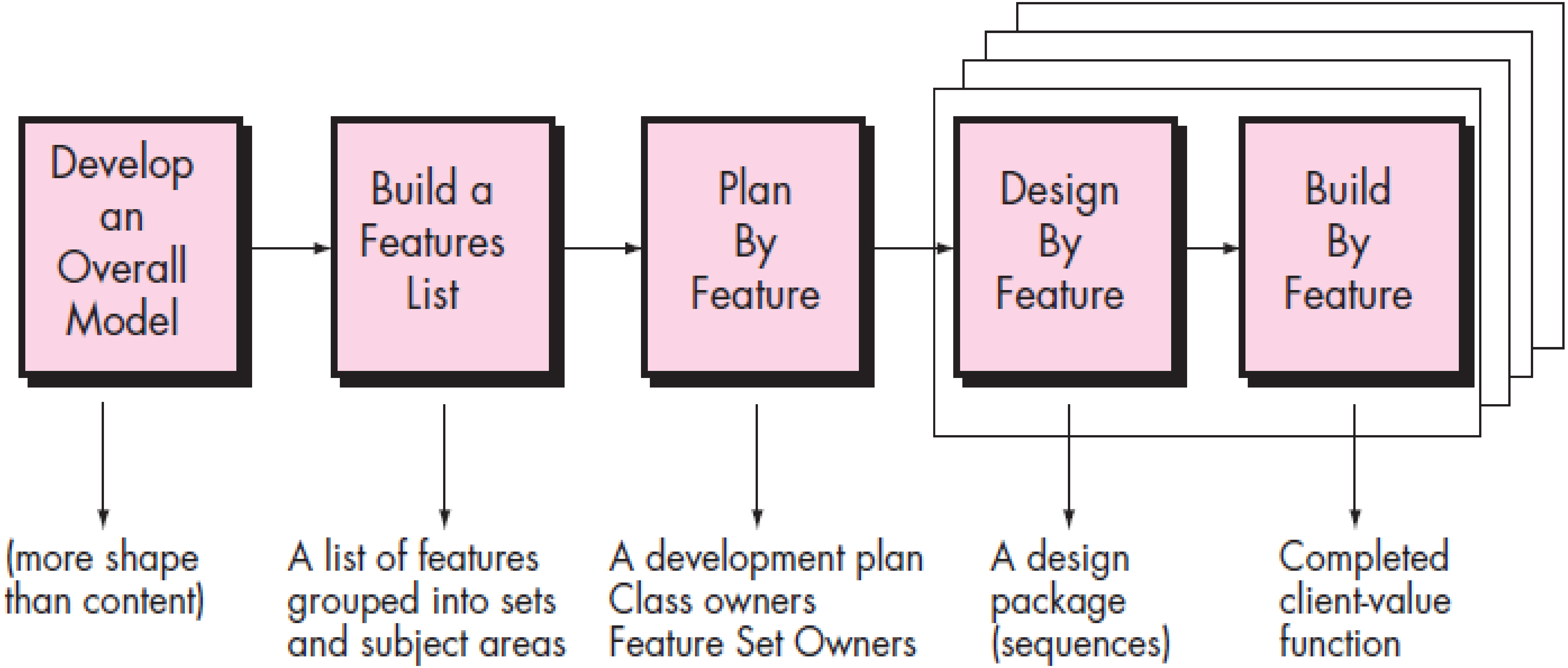
- primary goal of delivering useful, working software
- set of methodologies, each with core elements that are common to all, and roles, process patterns, work products, and practice that are unique to each.
- The intent is to allow agile teams to select the member of the crystal family that is most appropriate for their project and environment.

Feature Driven Development (FDD)

FDD adopts a philosophy that :

1) Emphasizes collaboration among people on an FDD team
2) Manages problem and project complexity using feature-based decomposition followed by the integration of software increments
3) Communication of technical detail using verbal, graphical, and text-based means.

Feature Driven Development (FDD)


Develop an Overall Model
Build a Features List
Plan By Feature
Design By Feature
Build By Feature
(more shape than content)
A list of features grouped into sets and subject areas
A development plan Class owners Feature Set Owners
A design package (sequences)
Completed client-value function

Feature Driven Development (FDD)

The emphasis on the definition of features provides the following benefits:

- **features are small blocks** of deliverable functionality - users can describe them more easily , understand how they relate to one another more readily
- Features can be organized into a **hierarchical business-related grouping**.
- Since a feature is the FDD deliverable software increment, the team develops **operational features every two weeks**.
- Because **features are small** - design and code representations are easier to inspect.
- **Project planning, scheduling, and tracking** are driven by the feature hierarchy, rather than an arbitrarily adopted software engineering task set.

Lean Software Development (LSD)

- adapted the principles of lean manufacturing to the world of software engineering
- lean principles can be summarized as eliminate waste, build quality in, create knowledge, defer commitment, deliver fast, respect people, and optimize the whole.

Agile Modeling (AM)

- Agile Modeling (AM) is a practice-based methodology for effective modeling and documentation of software-based systems.
- Agile Modeling (AM) is a collection of values, principles, and practices for modeling software that can be applied on a software development project in an effective and light-weight manner.

Agile Modeling (AM)

Agile Modelling principles :

- **Model with a purpose -**A developer should have a specific goal
- **Use multiple models –**
- **Travel light -** keep only those models that will provide long-term value. Every work product must be maintained as changes occur.
- **Content is more important than representation –**
- **Know the models and the tools you use to create them –** Understand the strengths and weaknesses of each model
- **Adapt locally -** adapted to the needs of the agile team.

Agile Unified Process (AUP)

- The team iterates to achieve agility and to deliver meaningful software increments to end users as rapidly as possible.

Agile Unified Process (AUP) Activities :

- *Modeling*
- *Implementation.*
- *Deployment*
- *Configuration and project management.*
- *Environment management.*