Information Systems Modelling and Design

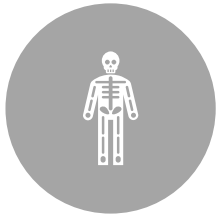
SYSTEM DEVELOPMENT METHODOLOGIES

RELEVANT CHAPTER IN THE CORE TEXT: CHAPTER 7

TOPIC 7

Topics to be covered

SOFTWARE DEVELOPMENT



SSADM AND DSDM

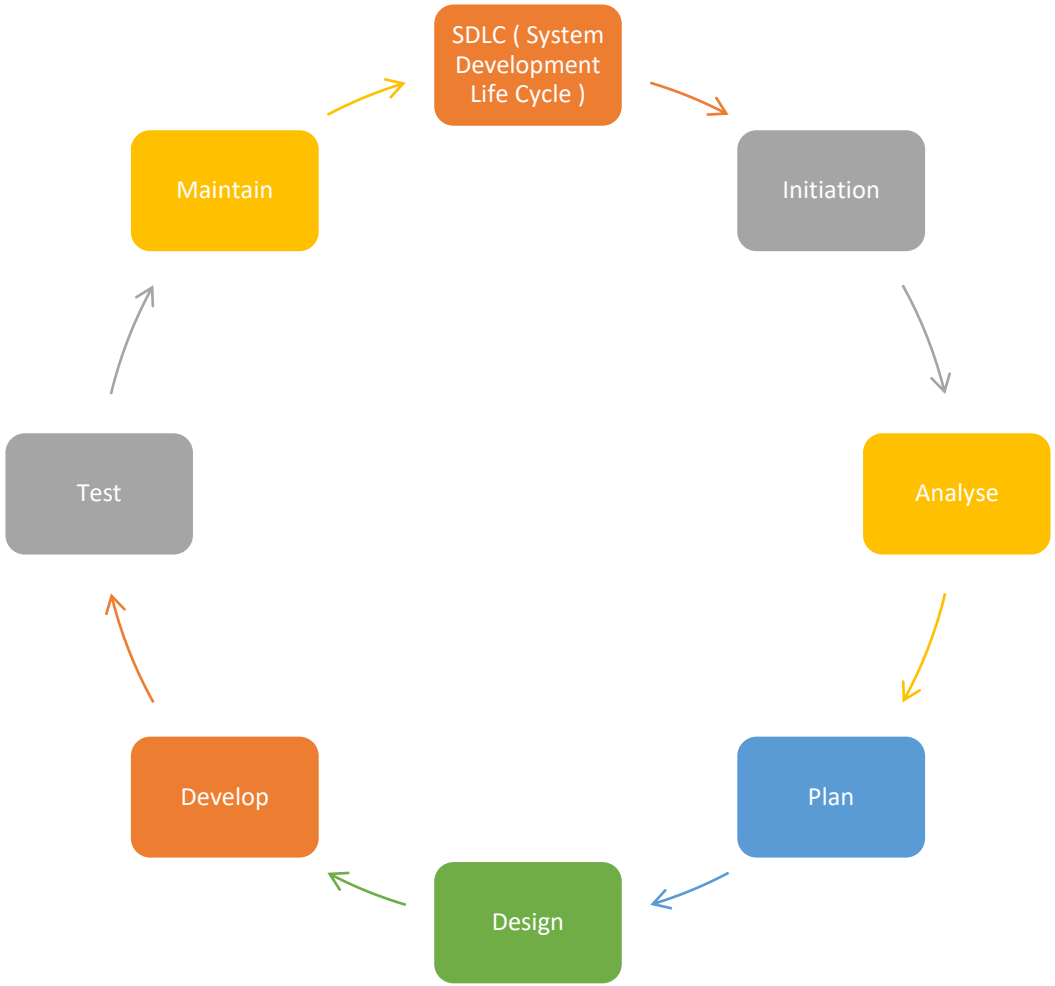
WATERFALL AND AGILE METHODOLOGIES

KANBAN, XP, AUP, SCRUM

SUMMARY

REFERENCES

Related Topics



SSADM

Structured systems analysis and design method (SSADM):

A methodology that defines the methods of analysis and design that should occur in a large-scale software development project. It is used extensively in the UK, particularly in government and public organisations.

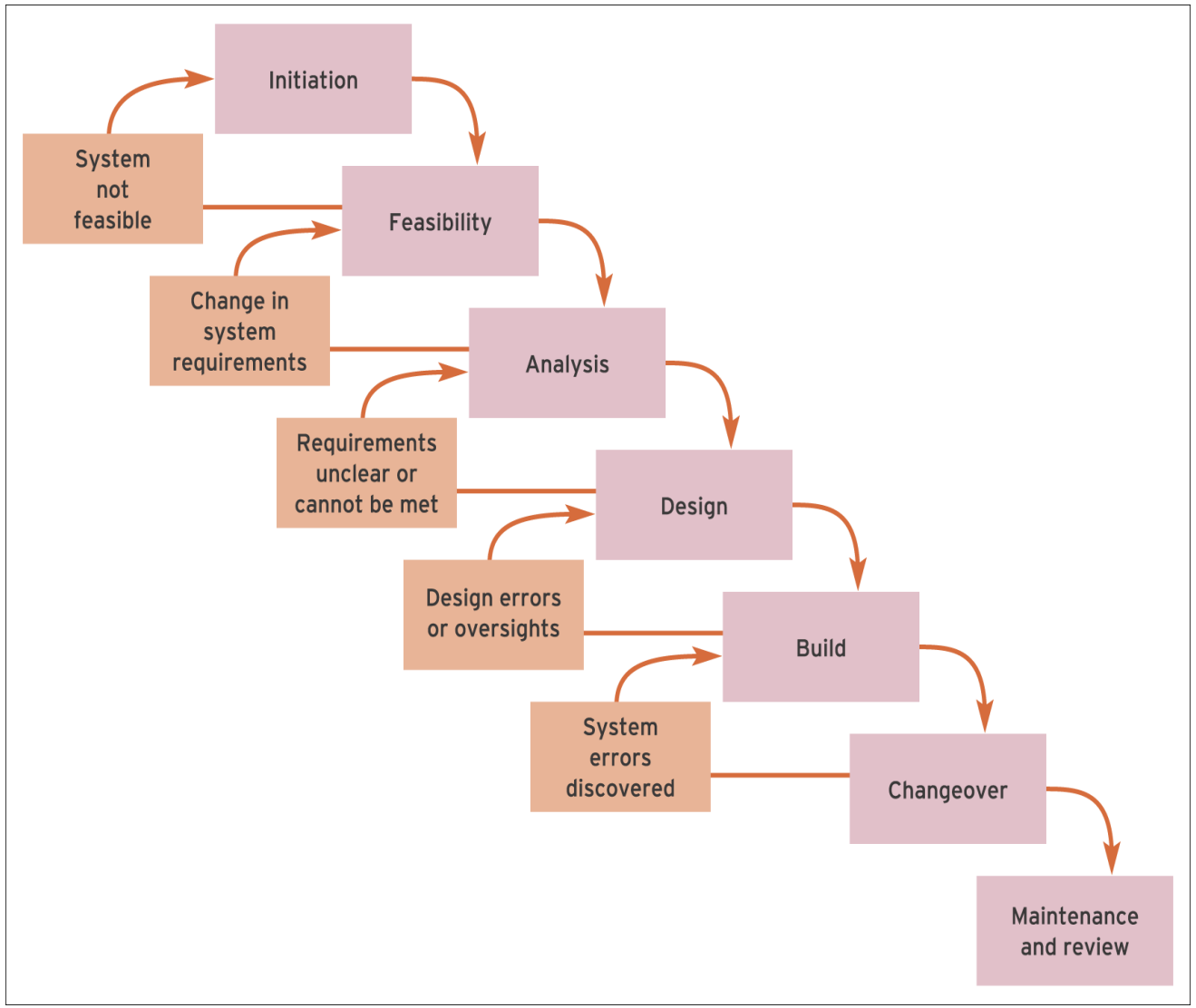
Feasibility study, requirements analysis, review of business options, technical options, logical design and physical design

Waterfall model of systems development

Waterfall model: Outlines the series of steps that should occur when building an information system.

The steps usually occur in a predefined order with a review at the end of each stage before the next can be started.

Figure 7.5 The traditional waterfall model of information systems development



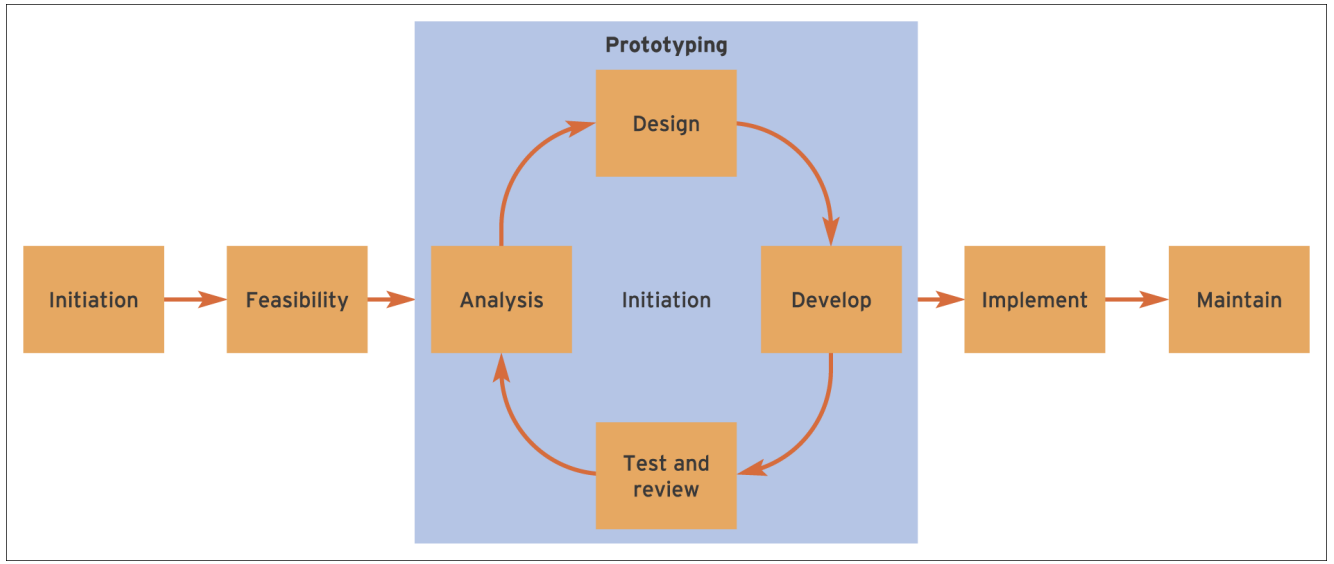


Figure 7.6 The role of prototyping within the systems development lifecycle

Spiral model

Spiral model: An iterative systems development model in which the stages of analysis, design, code and review repeat as new features for the system are identified.

The four main activities of this model are as follows:

Planning.	*Risk analysis.*	*Engineering.*	*Customer evaluation.*

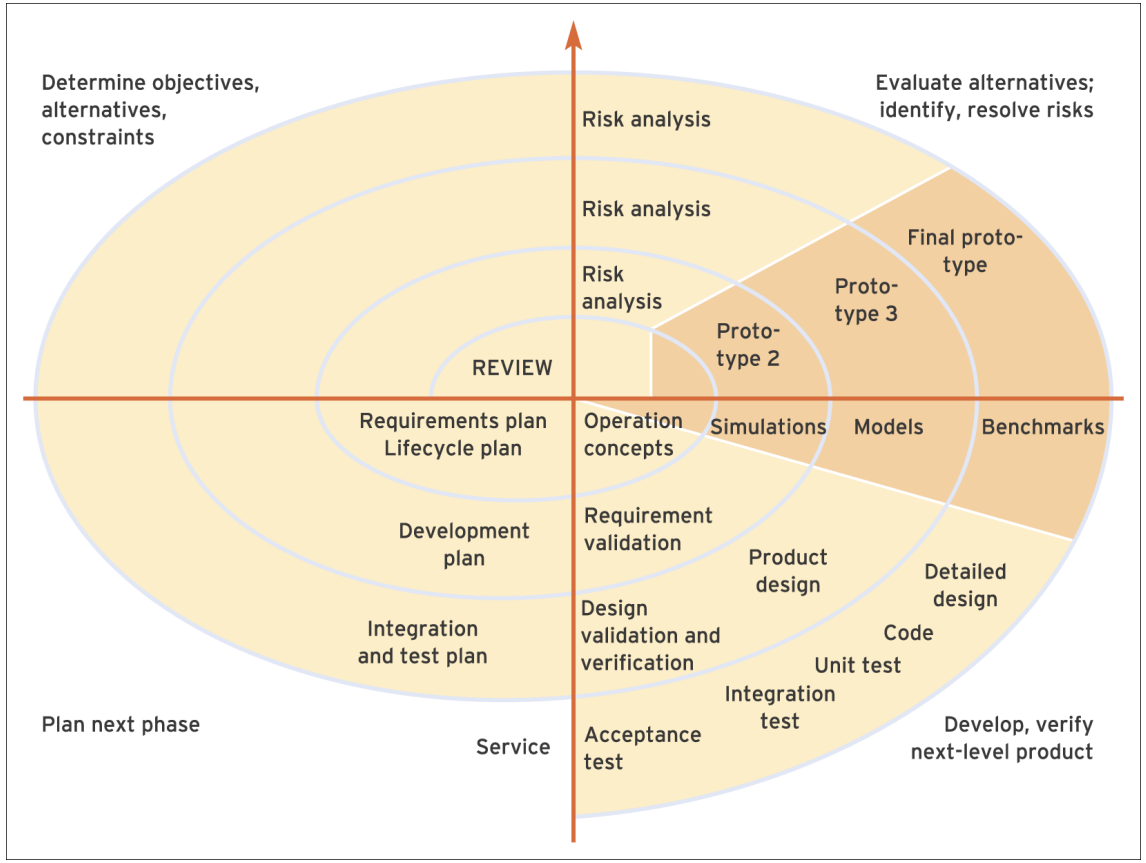


Figure 7.7 Boehm's spiral model of systems development

Dynamic Systems Development Methodology (DSDM)

- **Dynamic Systems Development Methodology (DSDM)**: A methodology that describes how RAD & AGILE can be approached.
  1. *Active user involvement is imperative (crucial)*.
  2. *DSDM teams must be empowered to make decisions*.
  3. *The focus is on frequent delivery of products*.
  4. *Fitness for business purpose is the essential criterion for acceptance of deliverables*.
  5. *Iterative(repeating) and incremental development.*
  6. *All changes during development are reversible.*
  7. *Requirements are baselined at a high level*.
  8. *Testing is integrated throughout the lifecycle*.
  9. *A collaborative and co-operative approach between all stakeholders is essential*.

Rapid applications development (RAD)

Rapid applications development (RAD): A method of developing information systems that uses prototyping to achieve user involvement and faster development compared to traditional methodologies such as SSADM.

Prototyping: A prototype is a preliminary version of part or a framework of all of an information system which can be reviewed by end-users.

Characteristics:

- Rapid
- Small teams
- User involvement
- Frequent review and testing.

What Is Agile ?

- Agile --readiness for motion, nimbleness, activity, dexterity in motion

- Agility

  The ability to both create and respond to change in order to profit in a turbulent business environment
  - Companies need to determine the amount of agility they need to be competitive

- Chaordic
  - Exhibiting properties of both ***cha***os and ***ord***er
    - The blend of chaos and order inherent in the external environment and in people themselves, argues against the prevailing wisdom about predictability and planning

    - Things get done because people adapt, not because they slavishly follow processes

Agile Software Development

Agile software development is a conceptual framework for software engineering that promotes development iterations throughout the life-cycle of the project.

Software developed during one unit of time is referred to as an iteration, which may last from one to four weeks.

Agile methods also emphasize working software as the primary measure of progress

Why Agile ?

Modularity

Iterative

Time-bound

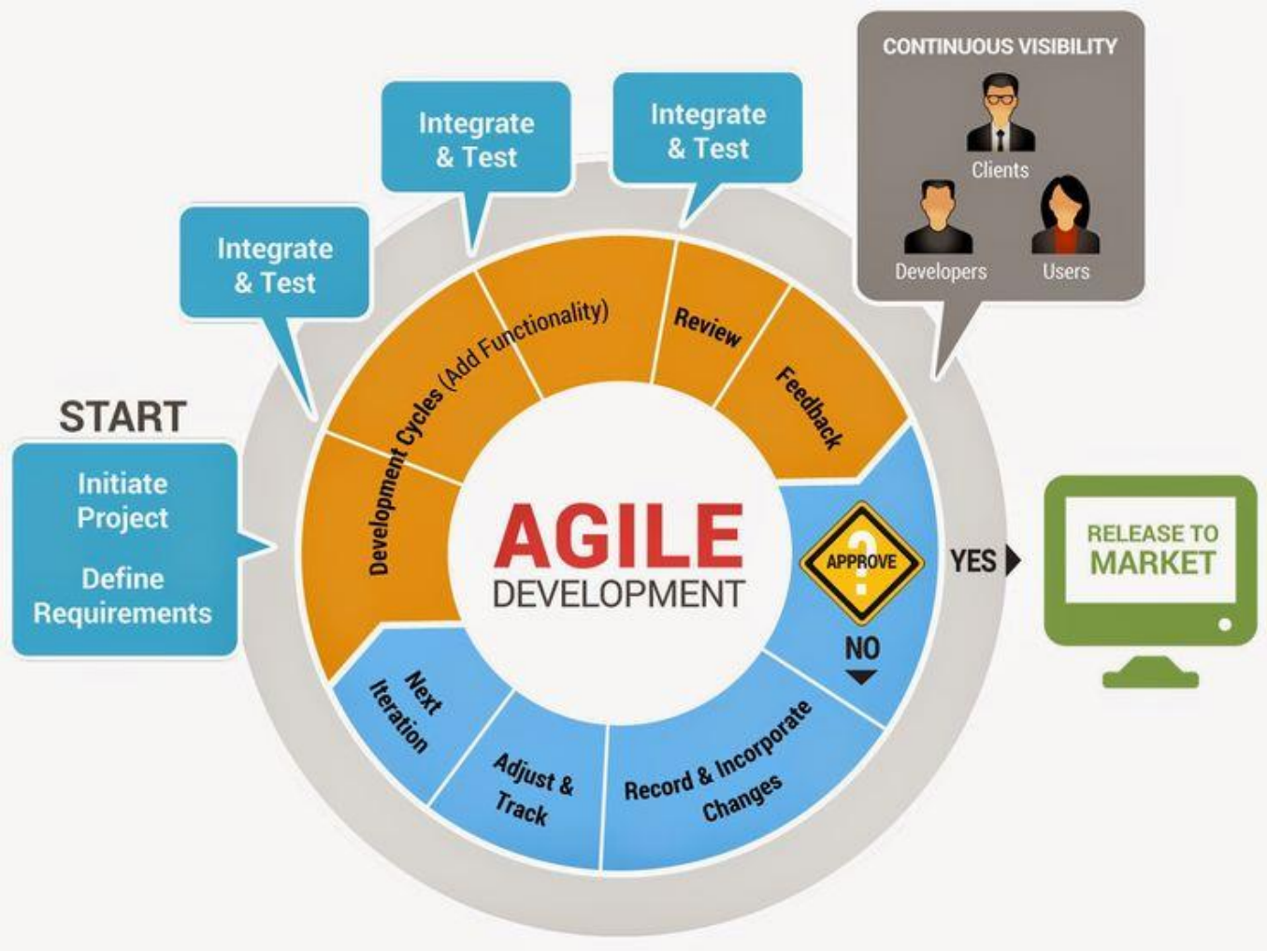
Incremental

Convergent

People-oriented

Collaborative

Agile Software Development

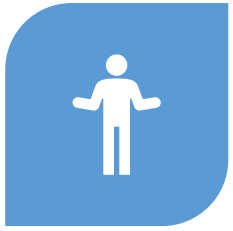


Characteristics of Agile Software Development

LIGHT WEIGHTED METHODOLOGY

SMALL TO MEDIUM SIZED TEAMS

VAGUE AND / OR CHANGING REQUIREMENTS

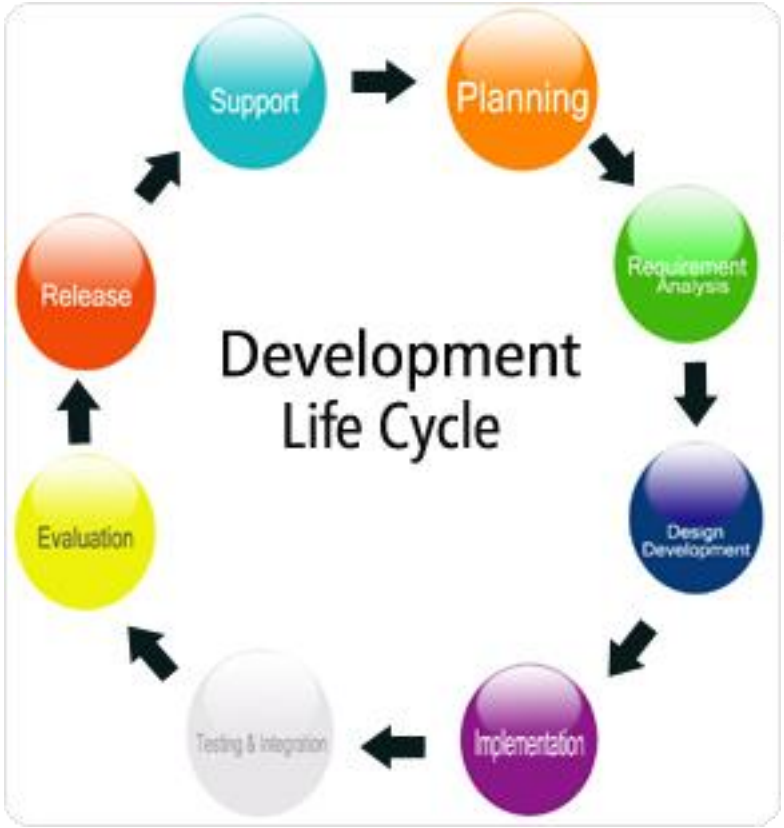


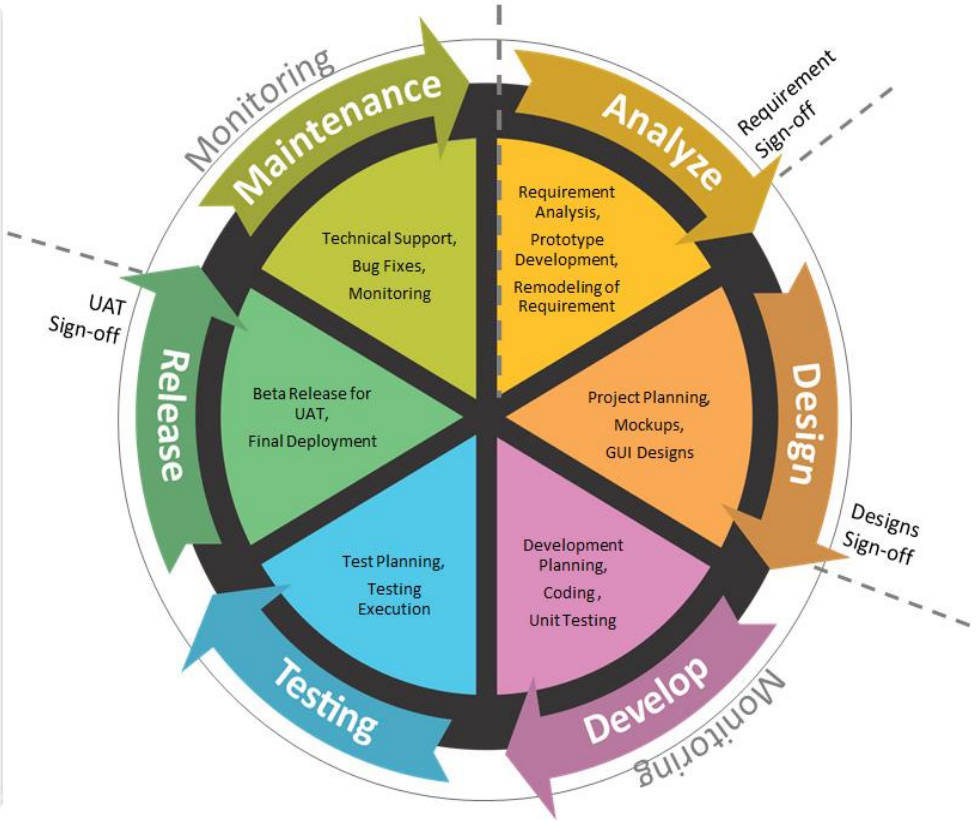
VAGUE AND/OR CHANGING TECHNIQUES

SIMPLE DESIGN

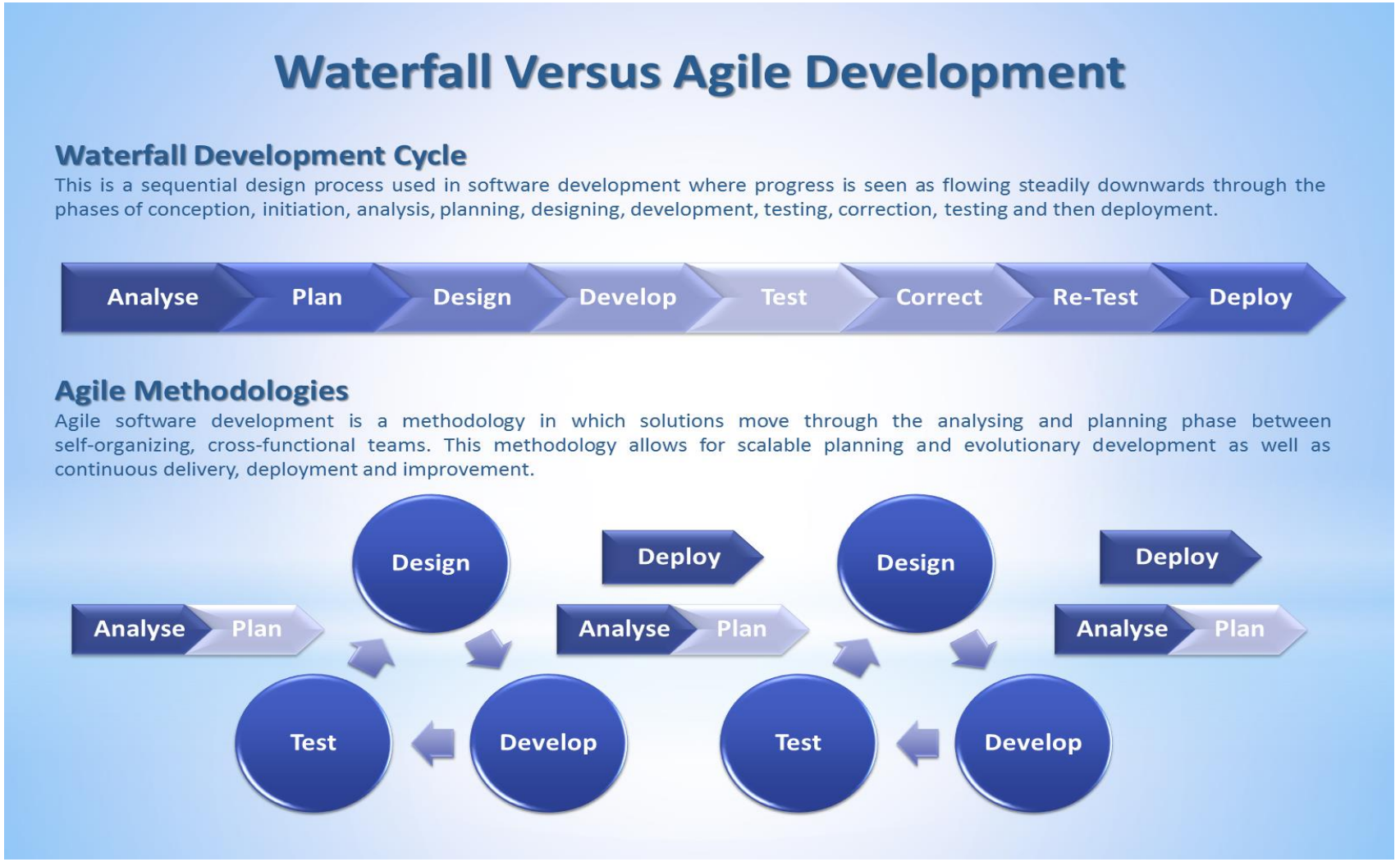
MINIMAL SYSTEM INTO PRODUCTION

Traditional Software Development

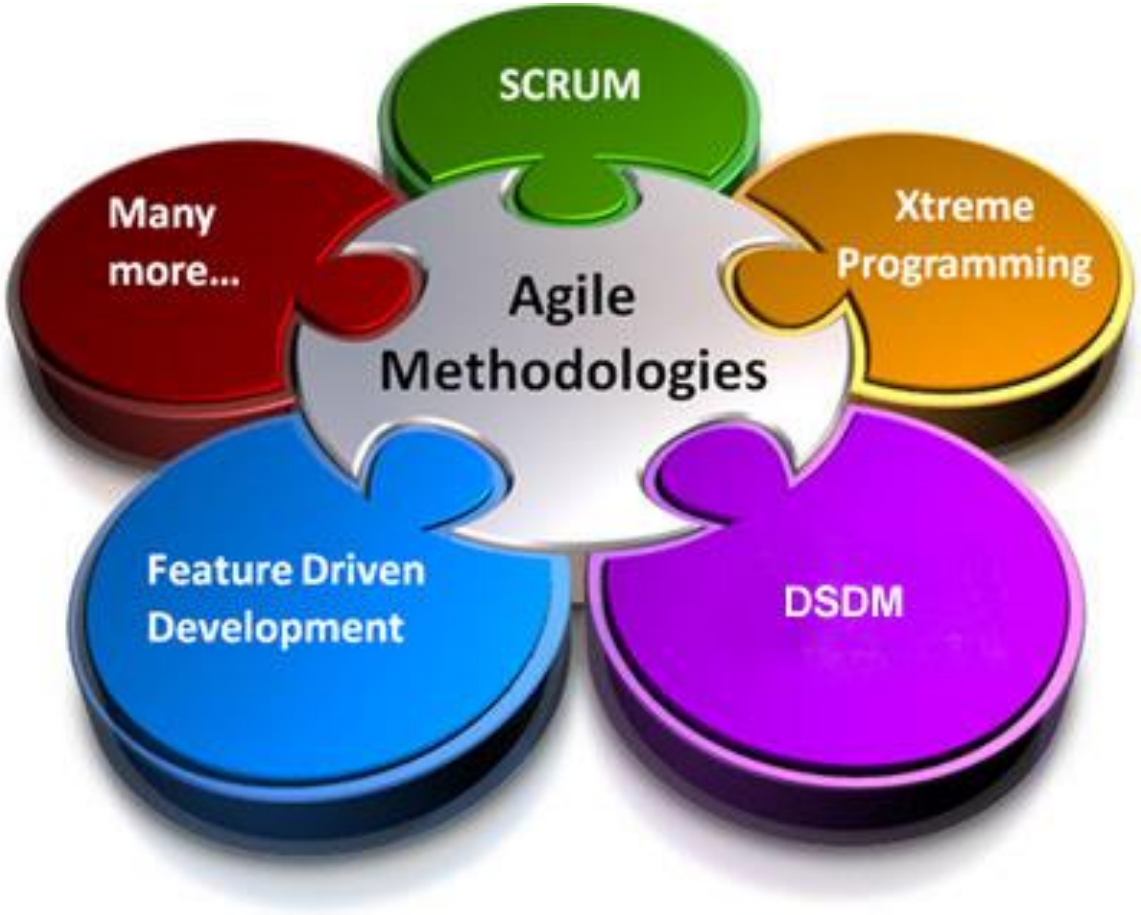




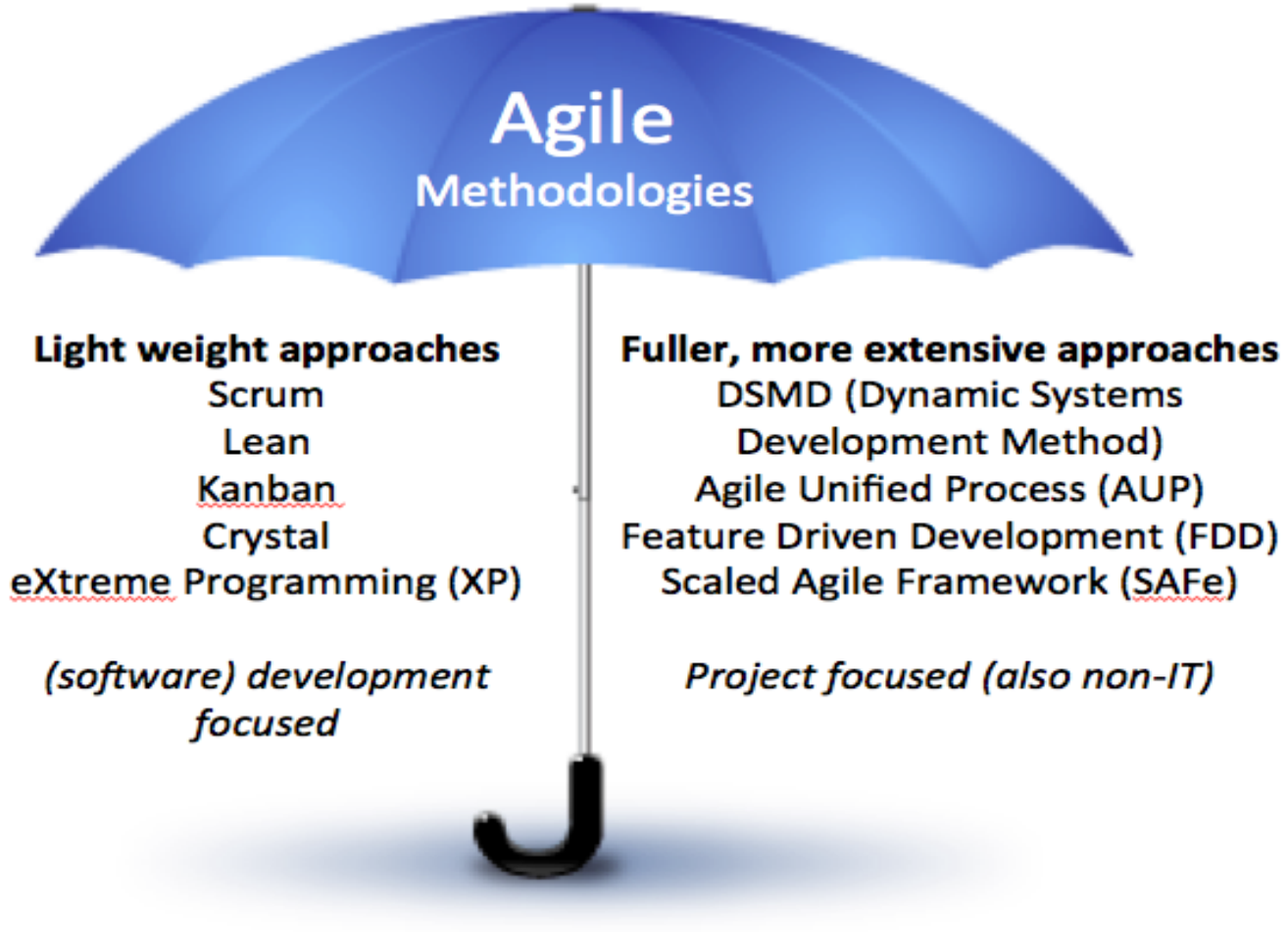
Waterfall vs Agile



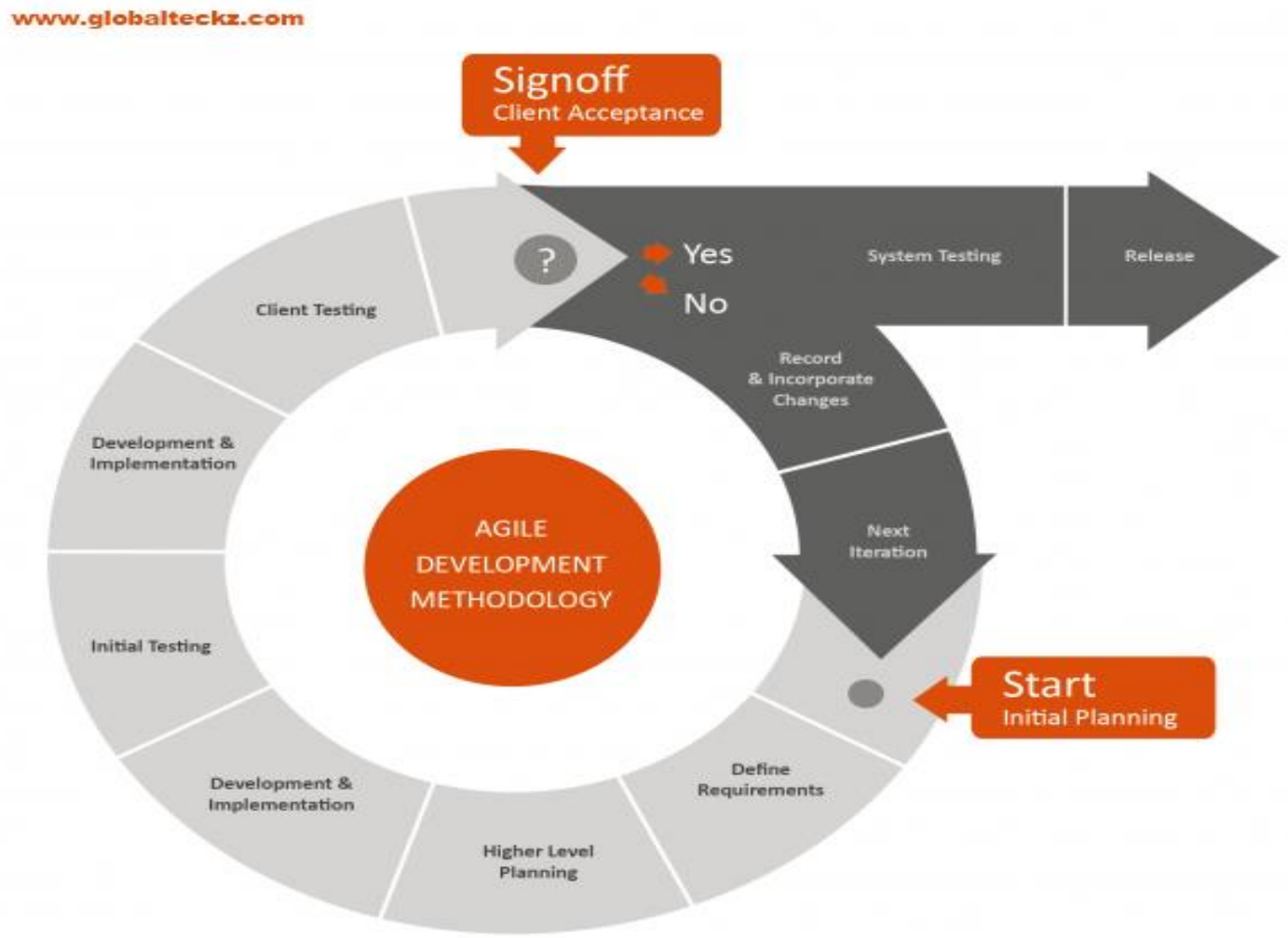
Agile Methodology



Light Weight and Fuller Agile



Characteristics



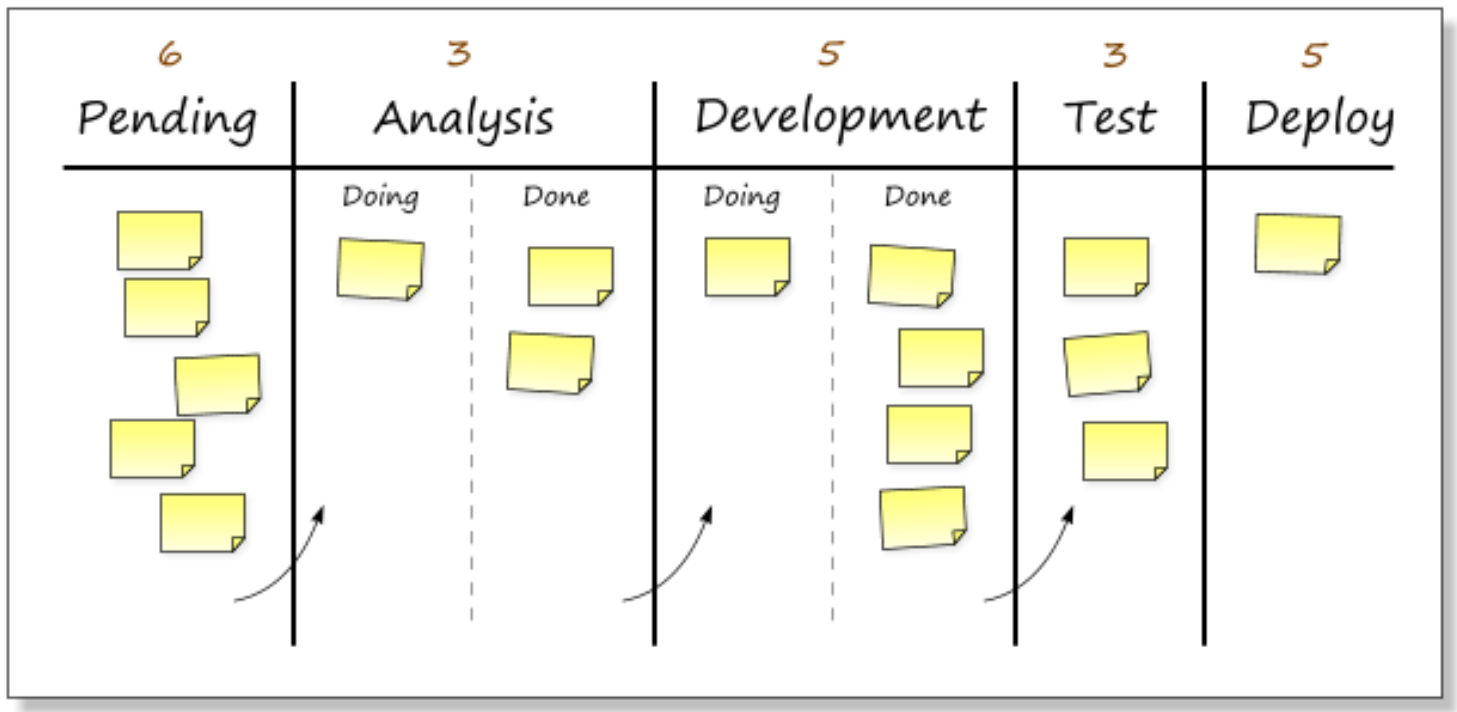
Existing Agile Methods

- Kanban
- Extreme Programming (“XP”)
- Agile Unified Process
- Scrum

Kanban

- Kanban is a new technique for managing a software development process in a highly efficient way.
- In Japanese, kanban literally translates to "visual signal."
- For kanban teams, every work item is represented as a separate card on the board.
- Kanban underpins Toyota's "just-in-time" (JIT) production system.

Kanban Boards



Extreme Programming

Most prominent Agile Software development method

Prescribes a set of daily stakeholder practices

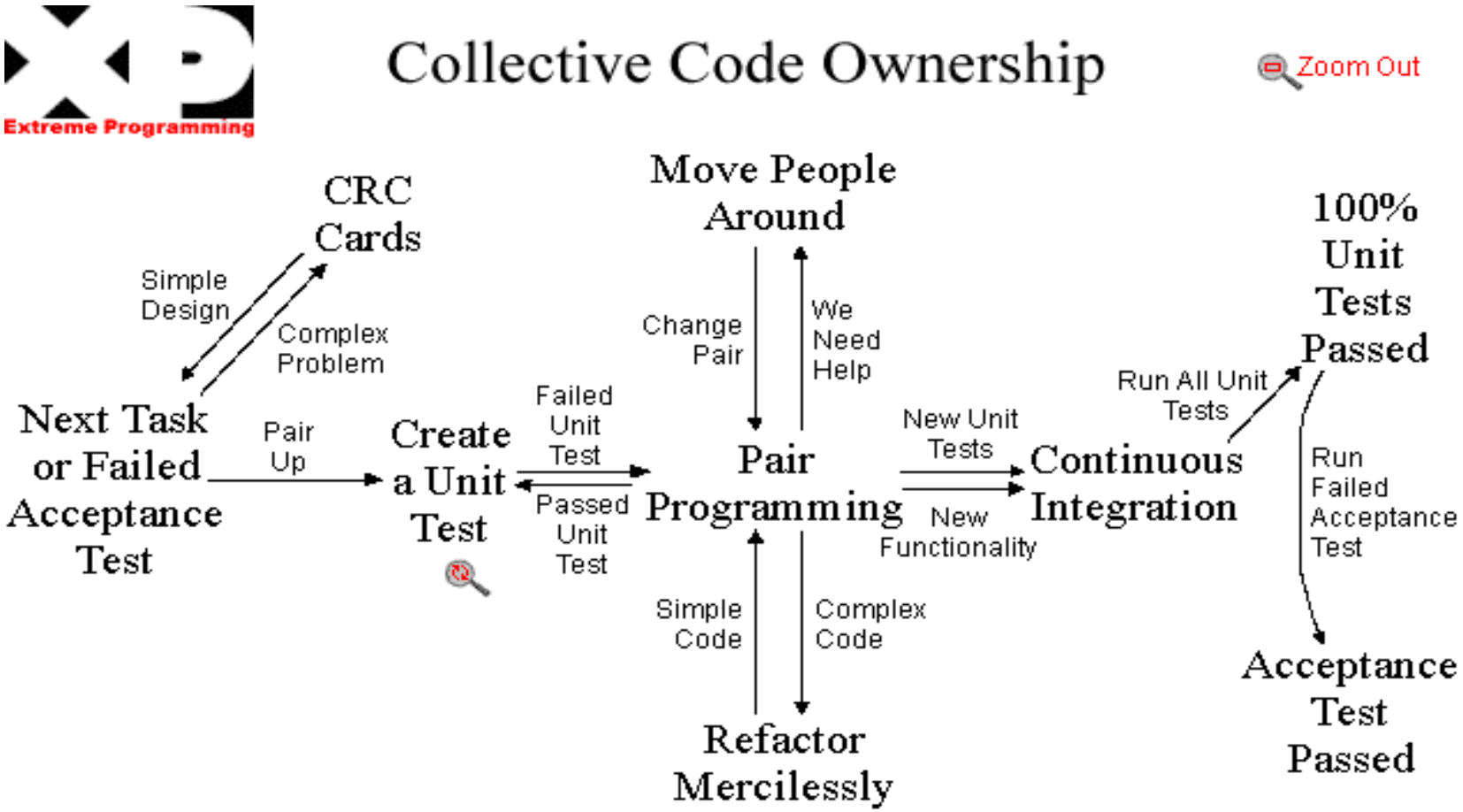
“Extreme” levels of practicing leads to more responsive software.

Changes are more realistic, natural, inescapable.

Extreme……


I'LL GO UP AND FIND OUT WHAT THEY NEED AND THE REST OF YOU START CODING !

Extreme Programming



Agile Unified Process

AUP is an iterative-incremental process consisting of four sub-processes or workflows

Phases of AUP:

Inception

Elaboration

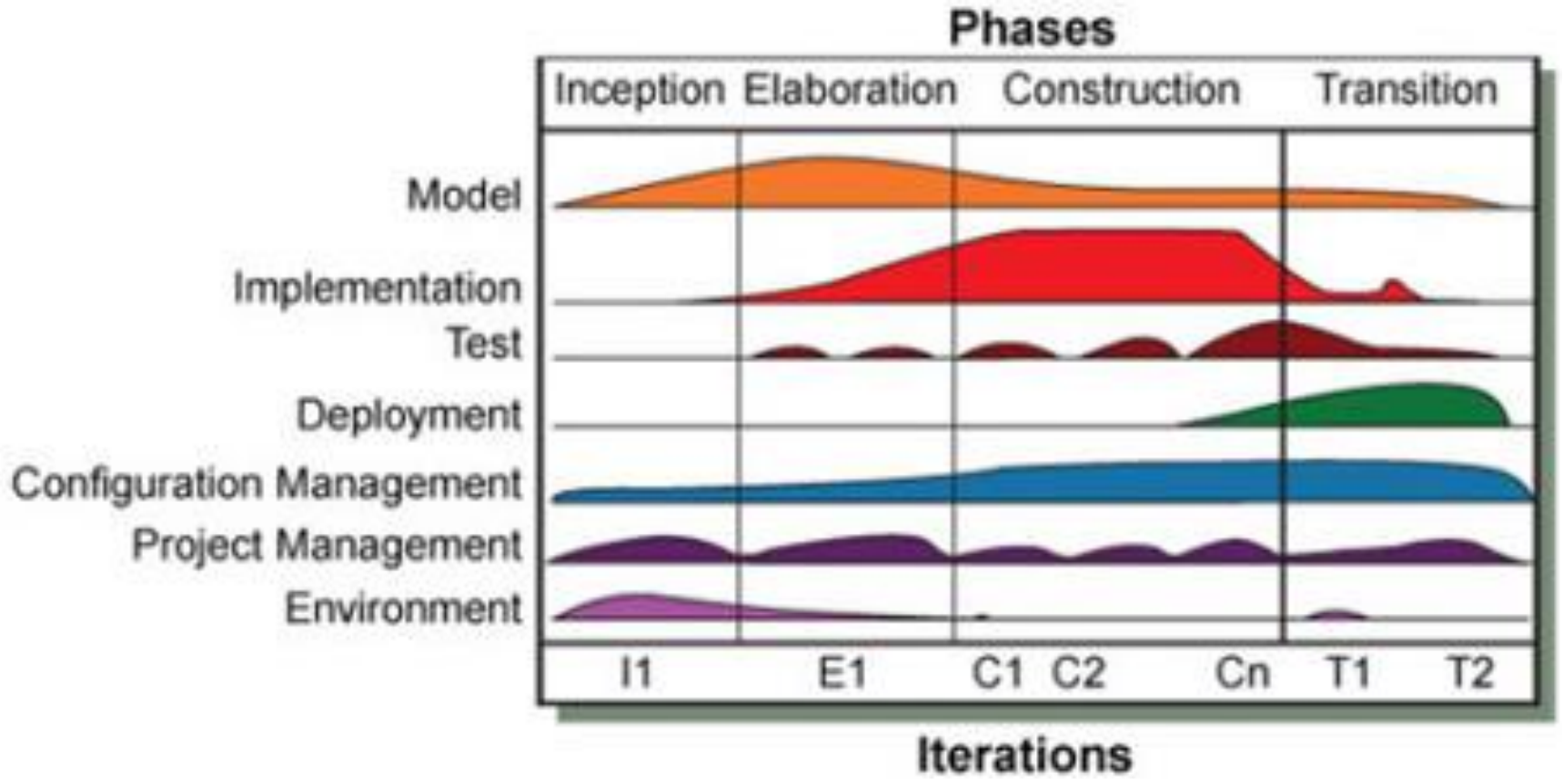
Construction

Transition

Disciplines of AUP

- Model
- Implementation
- Test
- Deployment
- Configuration Management
- Project Management
- Environment

AUP


Phases
Inception
Elaboration
Construction
Transition
Model
Implementation
Test
Deployment
Configuration Management
Project Management
Environment
I1
E1
C1
C2
Cn
T1
T2
Iterations

Scrum

It is an Agile S/w development method for project management

Characteristics:

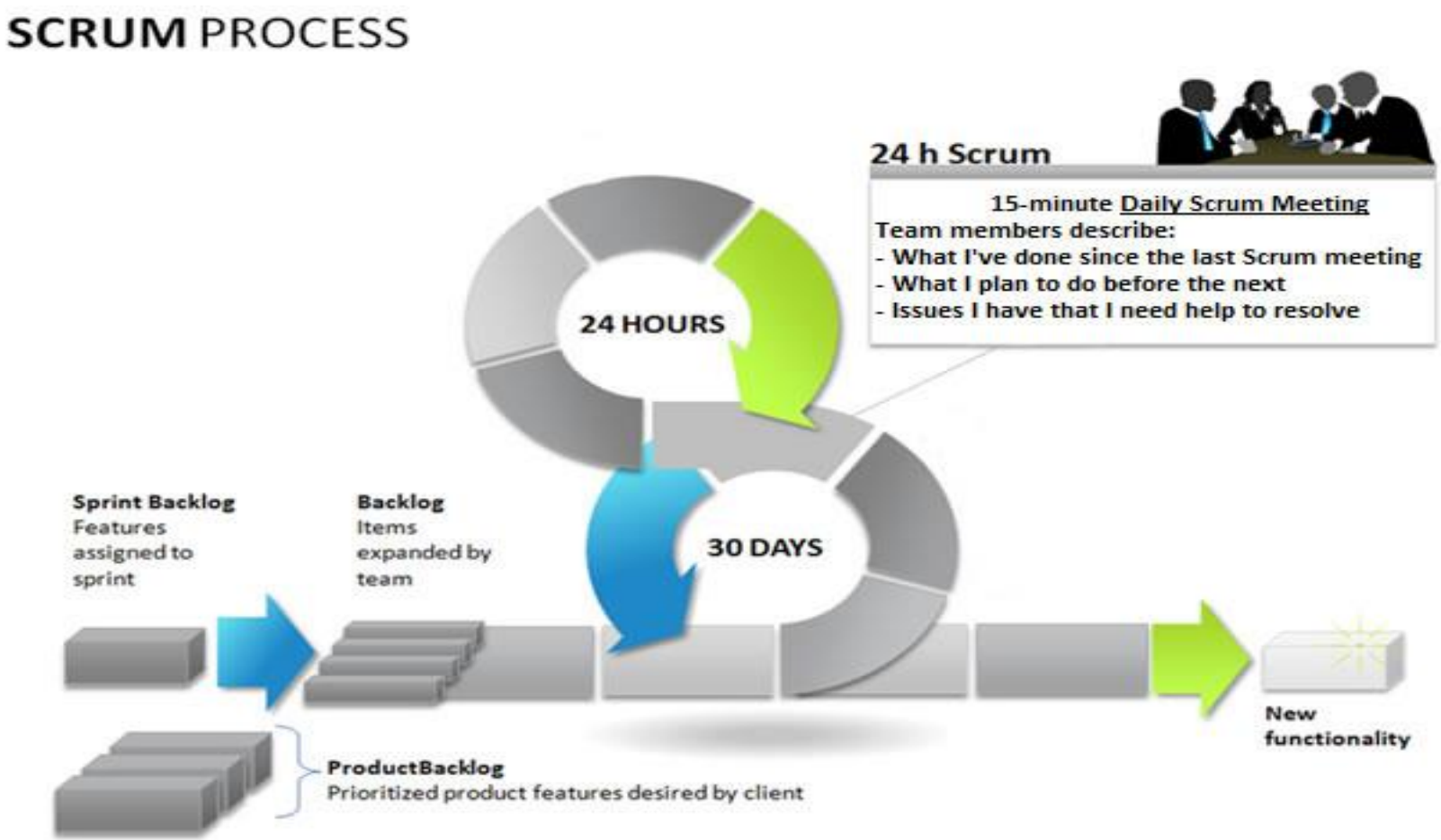
Prioritized work is done.

Completion of backlog items

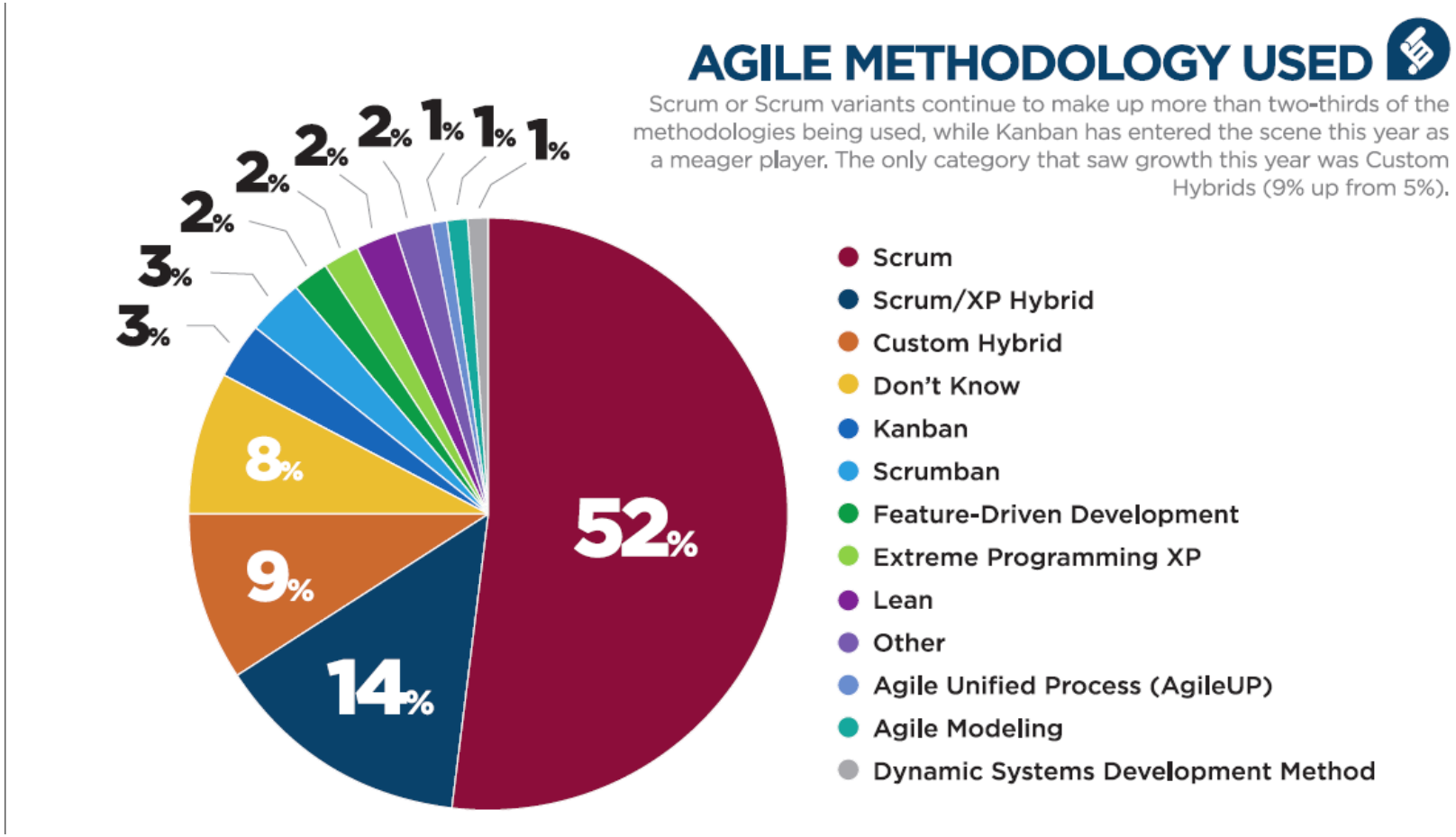
Progress is explained

Agile Software Development

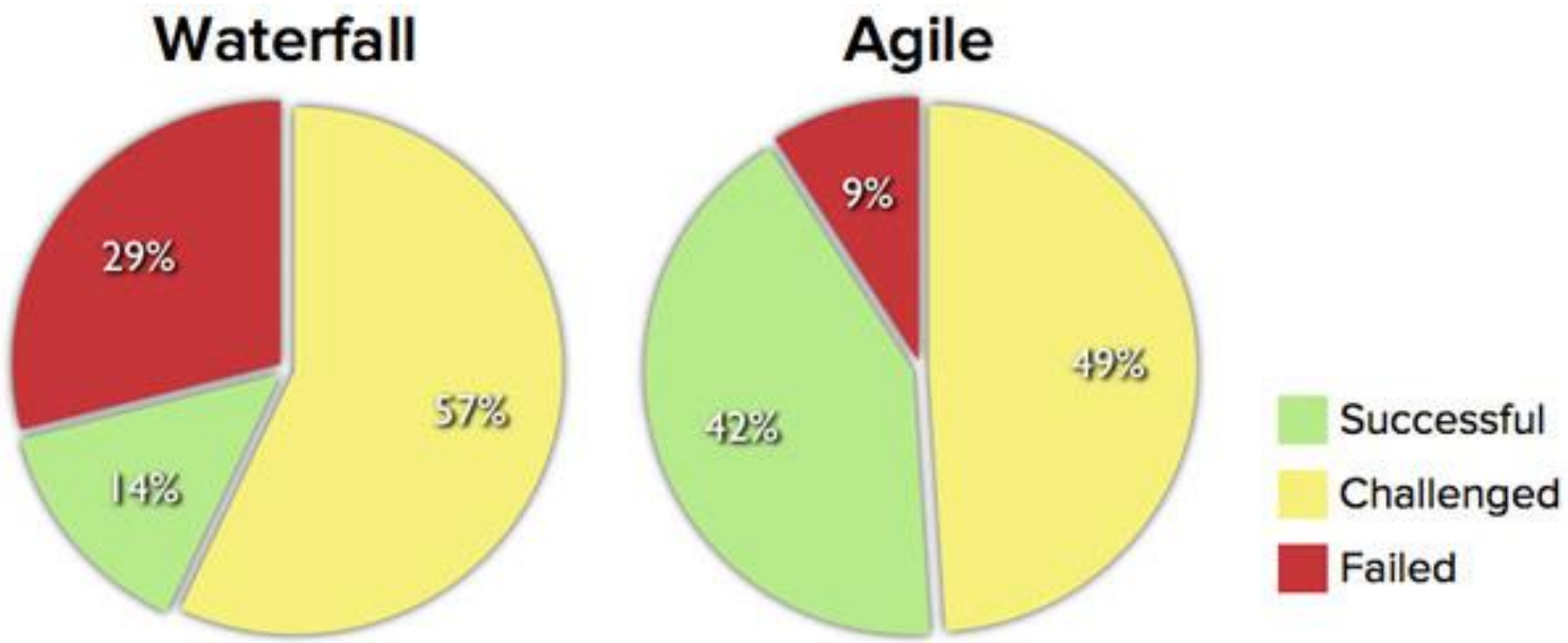
SCRUM Process



Agile Methodology Distribution



Why Agile ?



Source: The CHAOS Manifesto, The Standish Group, 2012.

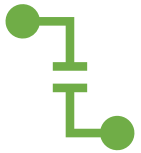
Conclusion

- Synthesises the existing literature.
- Each method is described in terms of process, roles, responsibilities, practices, adoption and experiences.
- Enables a selection criteria for comparing methods and pointing out their differences.

References

[1]. Abrahamsson P, Salo O and Ronkainen J. Agile software development methods (Review and analysis).

[2]. Scott W Ambler. Agile model driven development.



[3]. Cohen D, Lindvall M, Costa P. Agile software development.

Questions ?