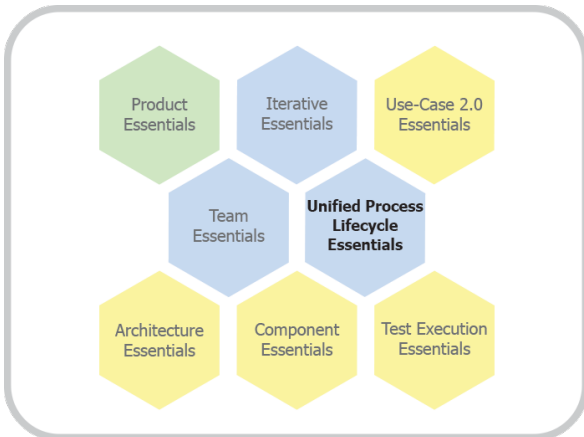


## Unified Process Lifecycle Essentials

Part of the IJI Essential Unified Process Practice Pack



*The Essential Unified Process (EssUP) focuses on the essentials to provide a pre-built assembly of eight easy-to-use practices that can be mixed and matched and used in different circumstances – all of them compatible with agile values and thinking. Use-case driven, iterative, component-based and architecturally-centric the practices provide an extensible framework for the addition of further practices.*

### Practice Overview



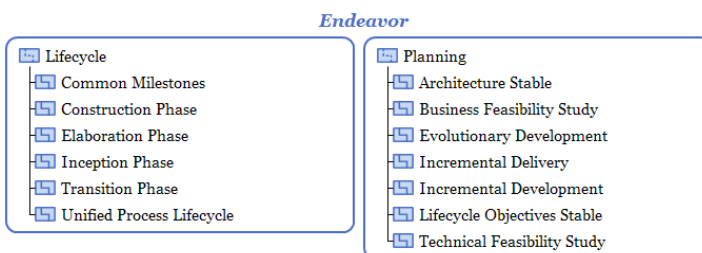
An agile, scalable way to control, plan and track software development projects. Use this practice to establish control over the lifecycle of an iterative development project.

This practice allows teams to:

- Establish a lifecycle for the project and plan effectively according to project circumstances.
- Share a set of common milestones with other projects and teams.
- Identify short-term objectives to reduce the levels of risk they face.
- Structure the plans into a sequence of well understood phases.
- Take full advantage of the benefits of iterative development.

### Lifecycle Planning Patterns

This practice contains a set of effective lifecycle planning patterns that help the team to:

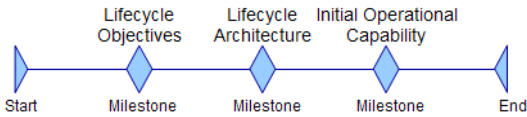


- Understand where the project is, and how well they are doing in addressing the risks.
- Adopt a standard control framework and establish appropriate objectives and milestones.
- Plan and iterate in a controlled manner.
- Balance the evolution of architecture and requirements alongside the development of a high quality software solution.

**Creating winning teams.**



## The Common Milestones

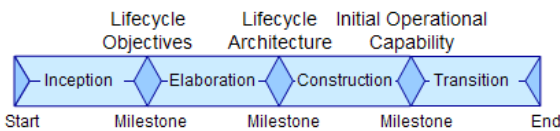


This pattern defines a set of milestones, or way points, suitable for the planning and tracking of all styles of iterative and incremental software development projects.

The pattern describes these three milestones for each product release cycle as follows:

- Lifecycle Objectives (LCO) – key scoping decisions are made about the product release. Operational requirements of the software are agreed.
- Lifecycle Architecture (LCA) – the software architecture is established and major associated risks are resolved.
- Initial Operational Capability (IOC) – the software is fully functional and preparations are made for transitioning the software to the customer and/or the live operating environment.

## The Lifecycle Phases



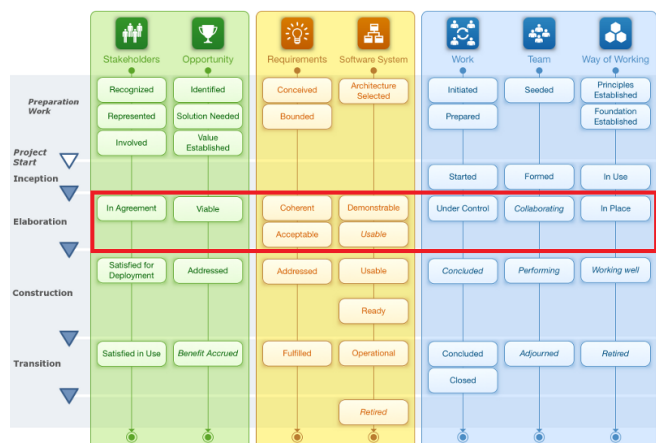
The practice refines the Common Milestones pattern by defining four project phases to progress a project successfully through the three common milestones.

The project or product release cycle is divided into four sequential phases, each with well-defined objectives:

- *Inception* - Confirm the scope and objectives and bring the business risks under control.
- *Elaboration* - Stabilize the plans and bring the architectural and technical risks under control.
- *Construction* - Build the product and bring the logistical, project execution risks under control.
- *Transition* - Deliver the product and bring the rollout risks under control.

## Controlled Alpha Evolution

By applying these phase and milestone patterns, the project can control the evolution of the main alphas.



Creating winning teams.