

Software development process methodologies

Methodology is a set of models, methods, practices and tools.

The methodologies classified as

- «Heavy»/«Formal» : RUP, MSF
- «Light»/«Flexible» : Scrum, Agile, eXtreme Programming

Rational Unified Process

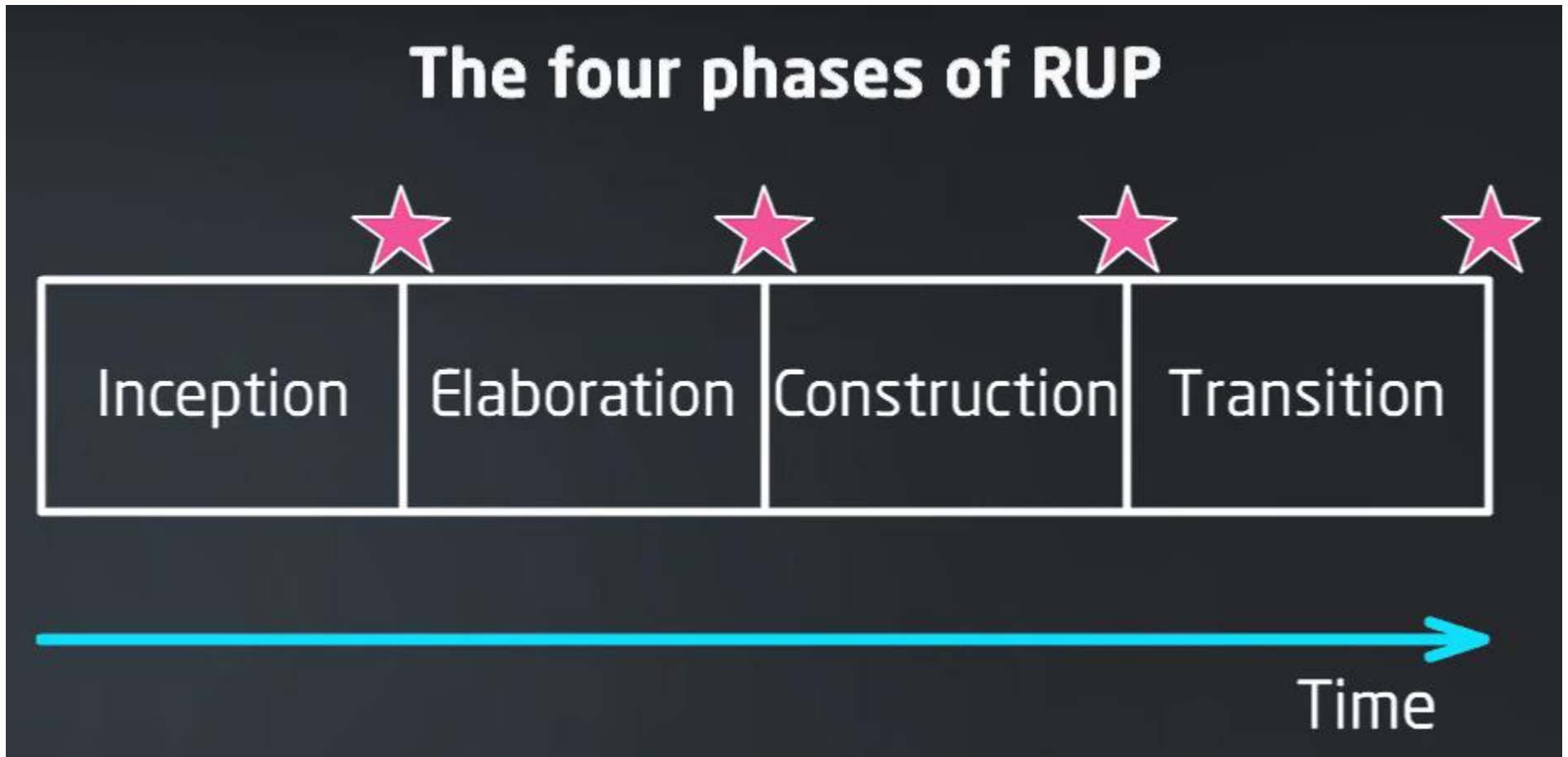
RUP – approach, which is

- Iterative
- Architecture centered
- Use-case based

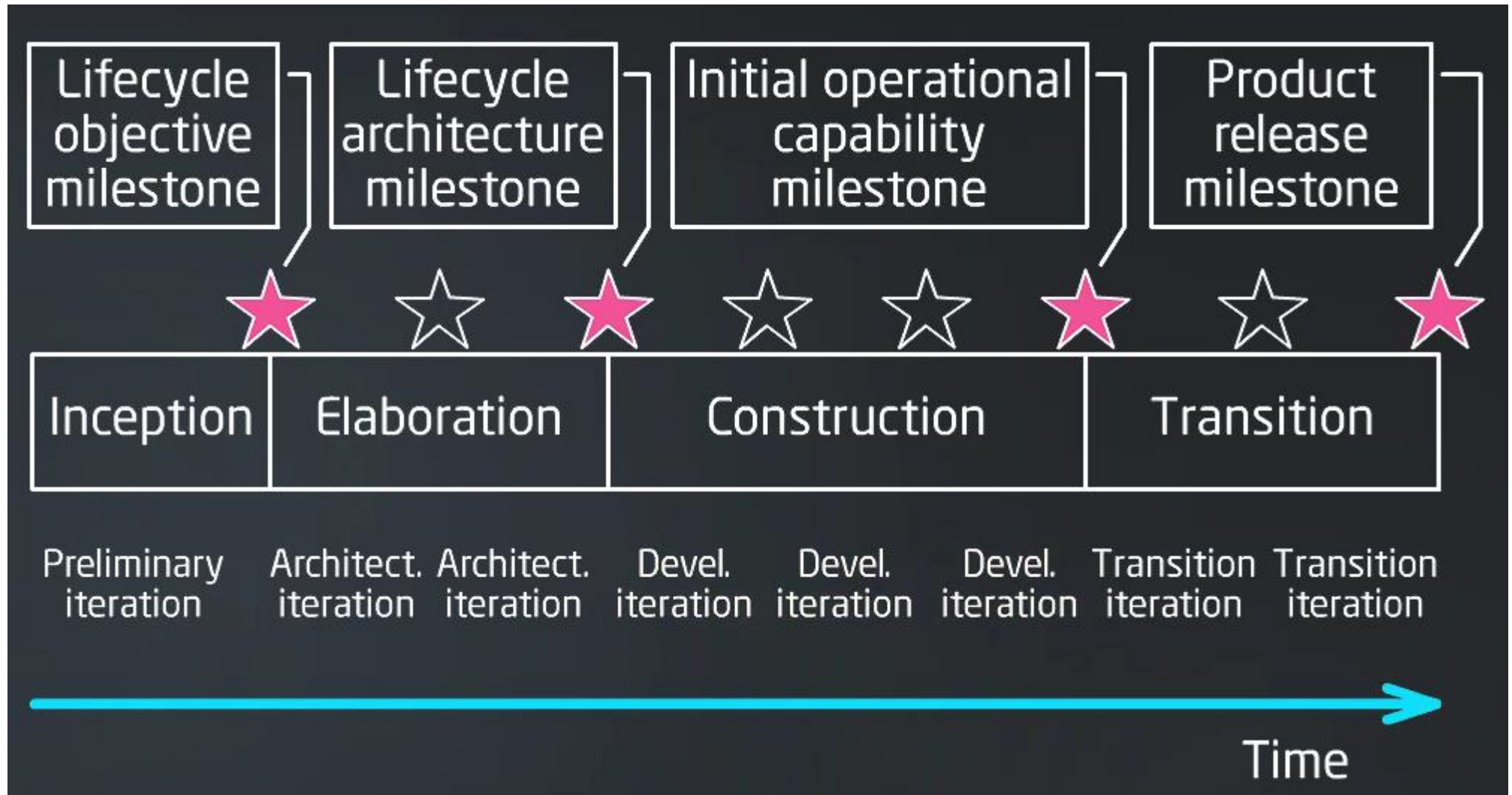
The four phases of RUP

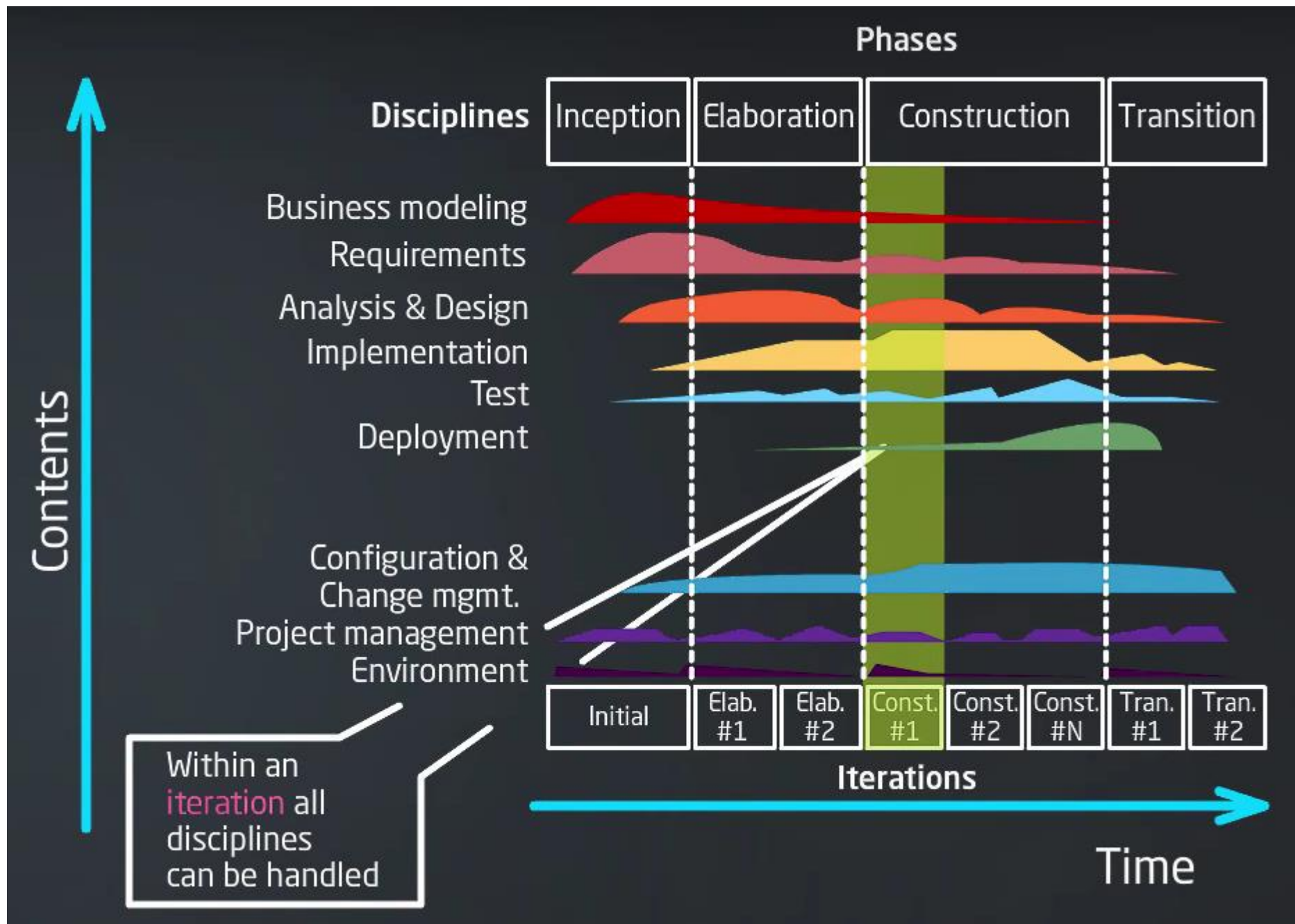
- Inception (What will be developed?)
- Elaboration (How it will be developed?)
- Construction (Product development)
- Transition (Product delivery)

The four phases of RUP



The four phases of RUP

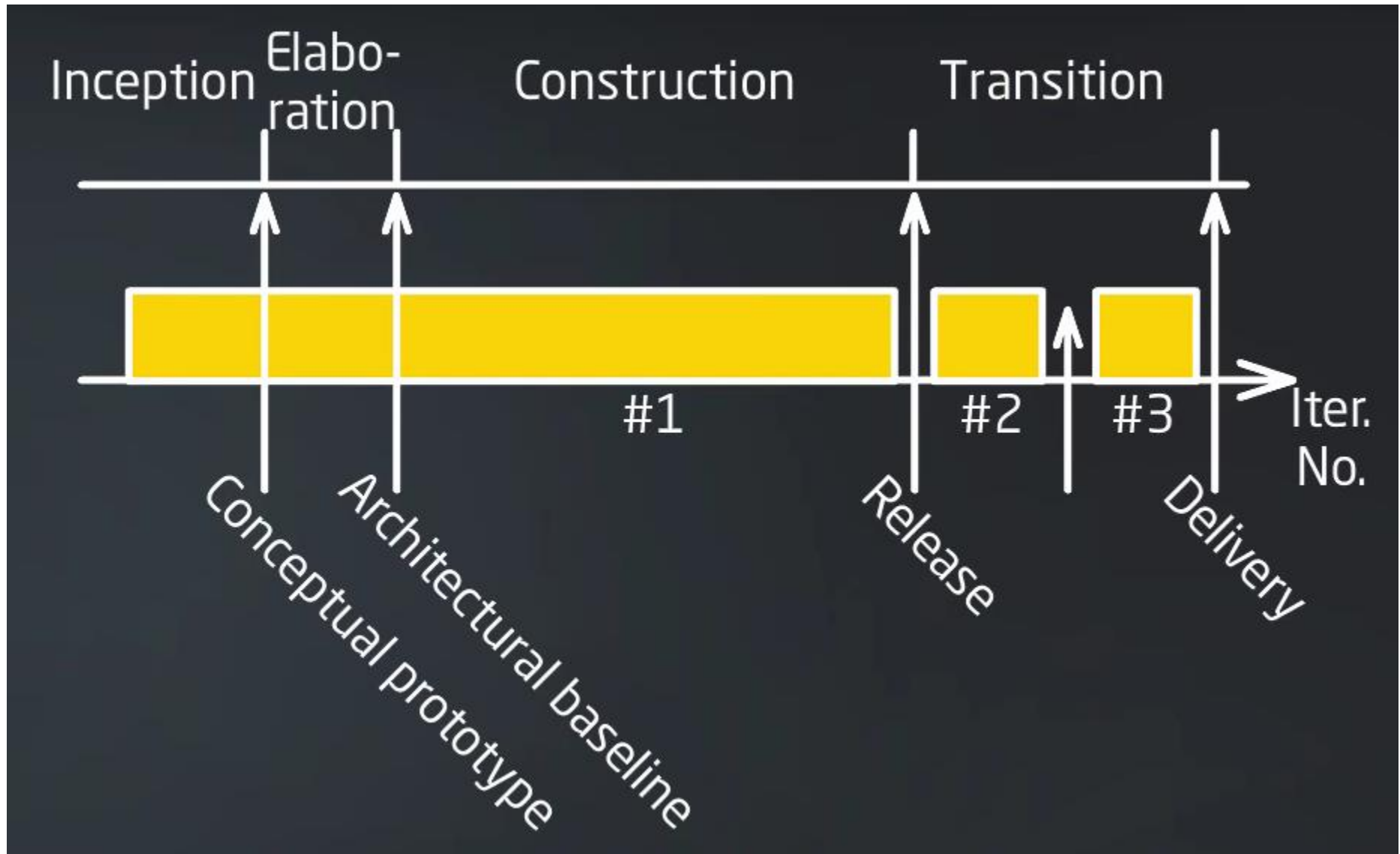




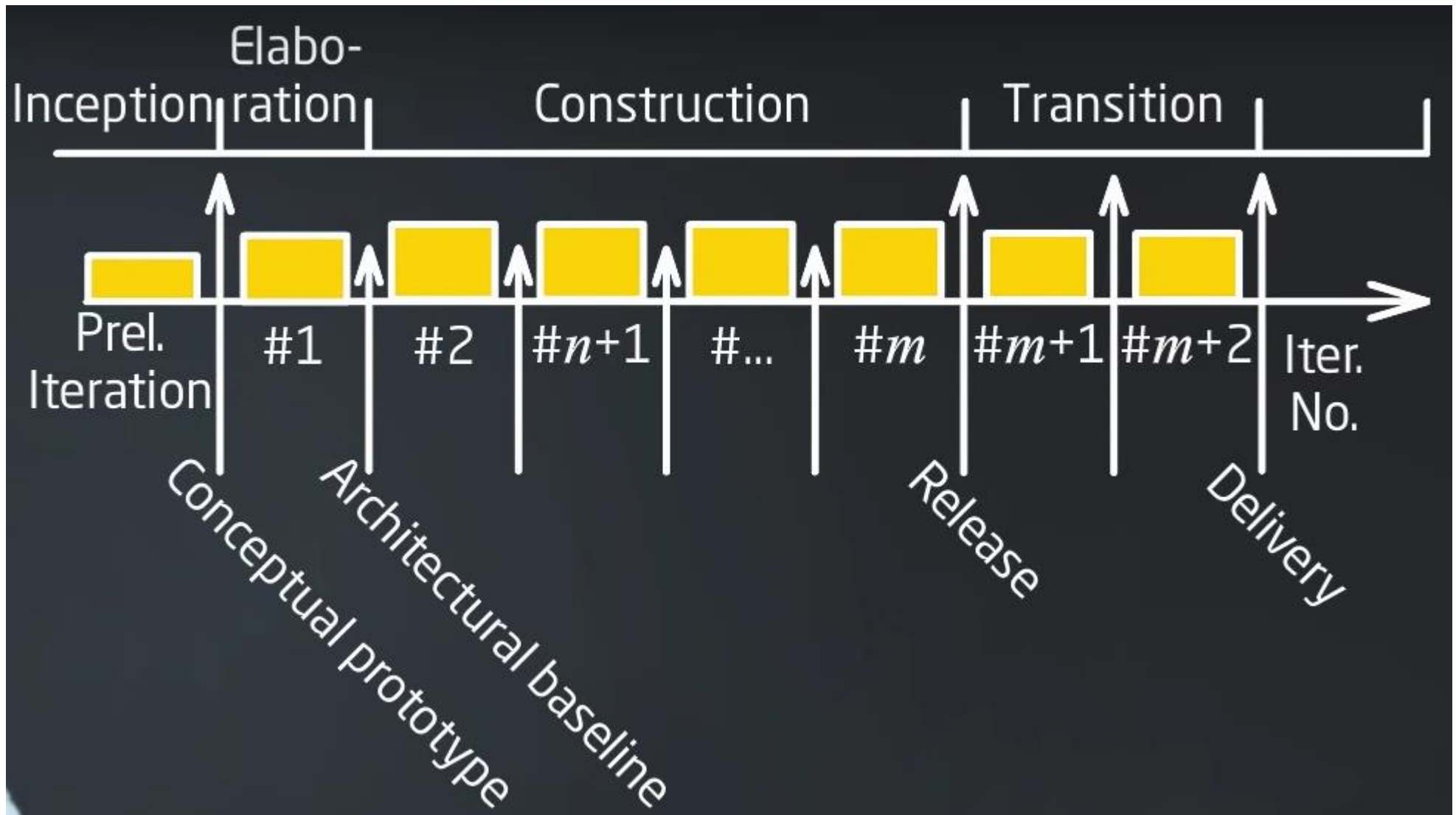
RUP practices

- Provide customer requirements satisfaction
- Concentrate on the program implemented
- Adjust to changes since project start
- Build a component system
- Build the foundation of implementable architecture ASAP
- Make quality a lifestyle

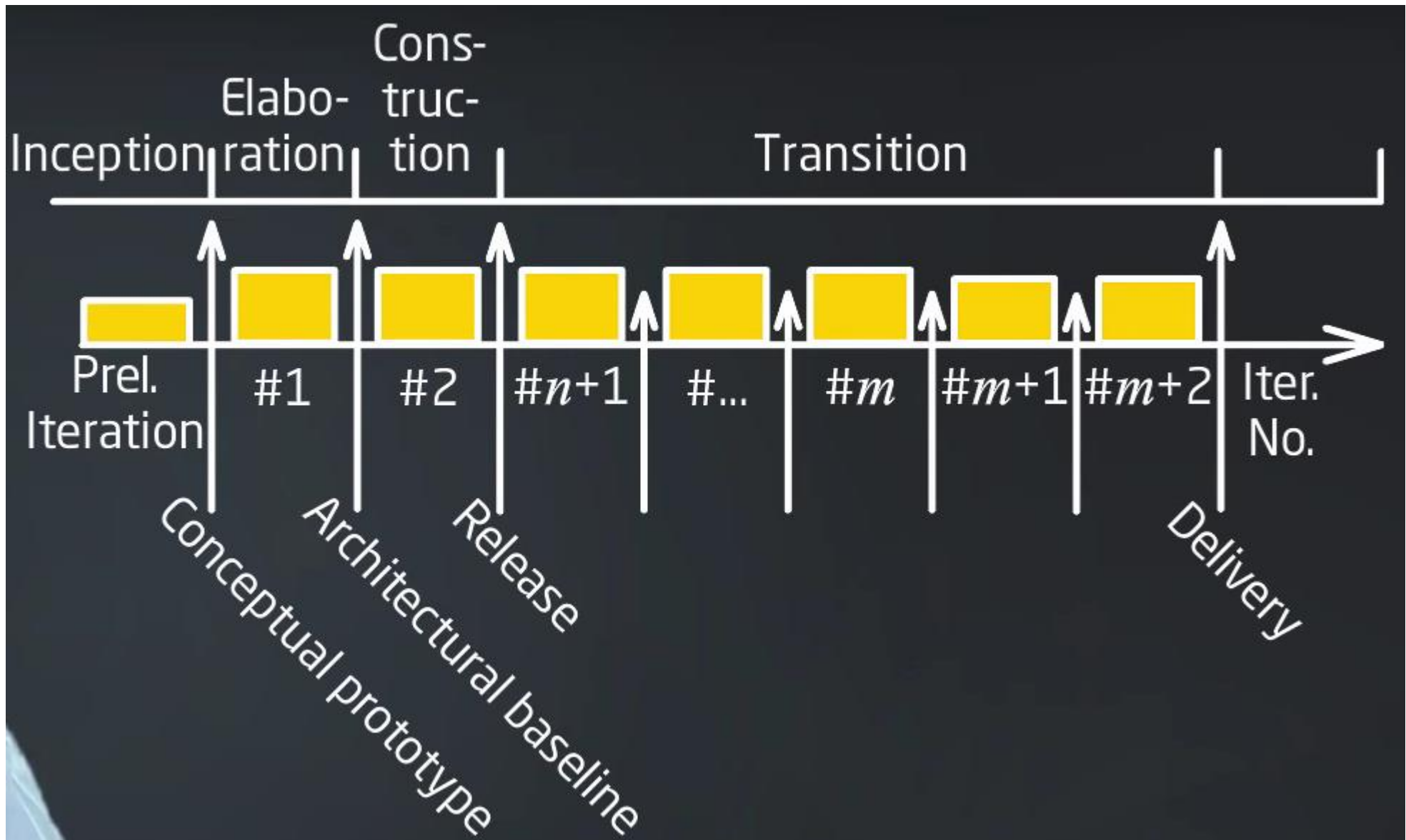
RUP waterfall lifecycle



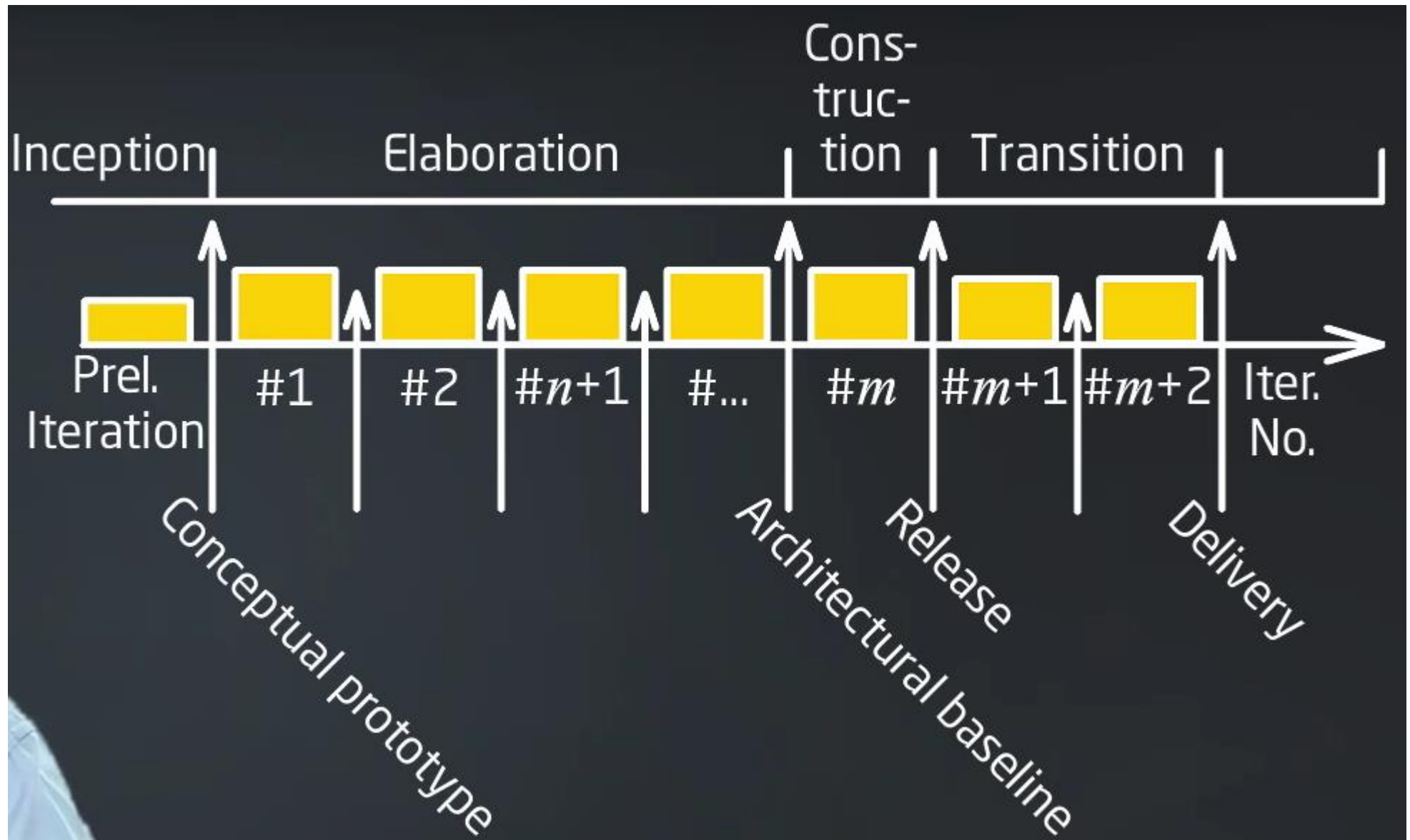
RUP: incremental lifecycle



RUP: incremental lifecycle 2



RUP: evolution lifecycle



RUP: organization

- RUP is iterative software (SW) development process framework
- RUP is organized in phases, iterations and workflows
- RUP structure includes roles, activities and artifacts
- RUP processes use manuals, patterns and SW user manuals
- RUP structure uses workflows of activities
- RUP includes a set of best practices

Design manual



Rational Rose
user manual



Role



Designer

Activities



Use-case
analysis



Use-case
design

Artifacts

Responsible for



Use-case development



Use-case pattern



Plan project configuration
and change control



Create project CM environments



Change and deliver
configuration items



Manage Baselines
and releases

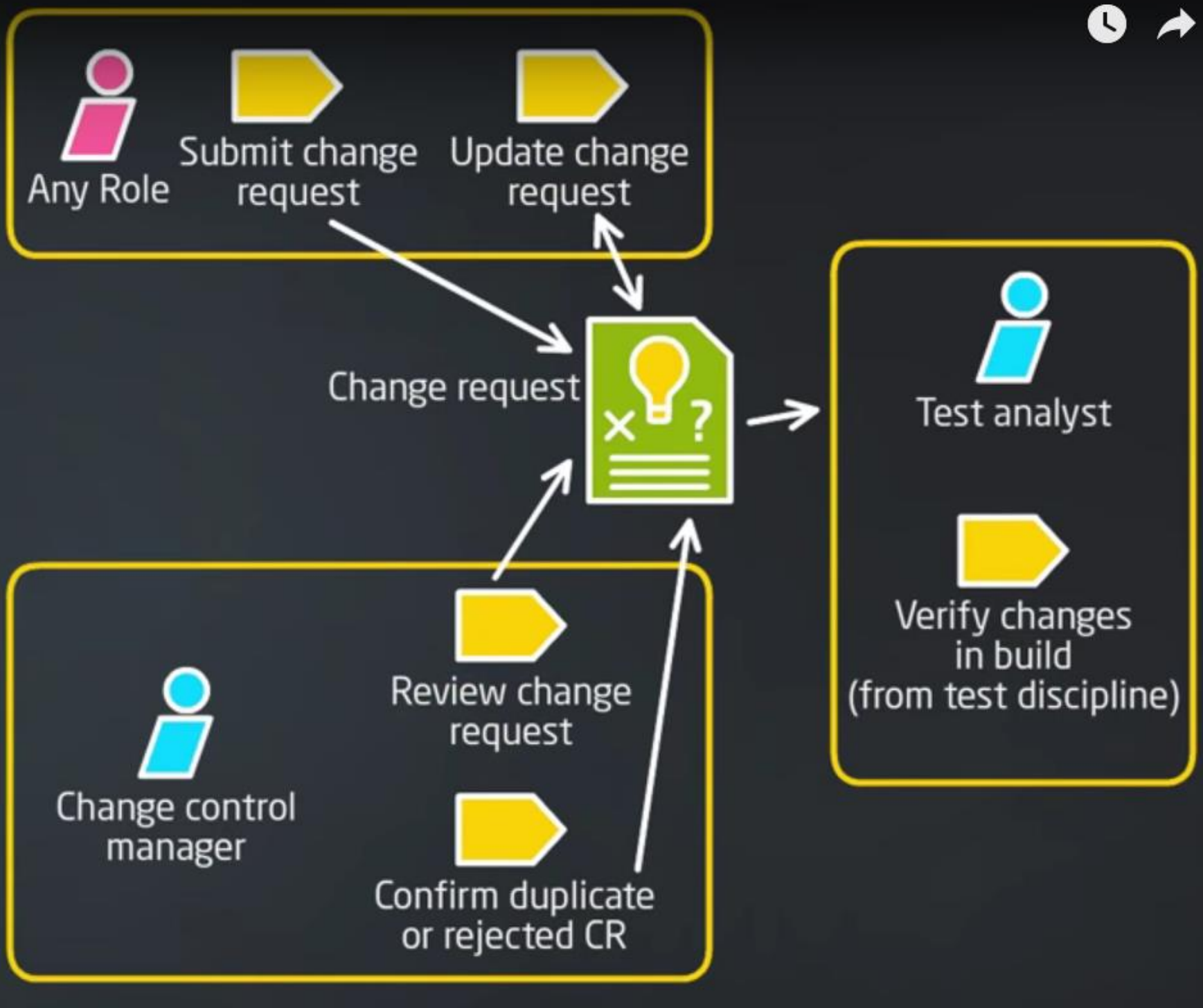


Monitor & Report
configuration status



Manage change
requests





Microsoft Solution Framework

MSF has two implementations:

- MSF Agile
- MSF Formal

Microsoft Operations Framework (MOF) is an addition

MSF = Build it right

MOF = Run it right

MSF elements

- Basic principles
- Models for teams and processes
- Disciplines of managing
- Key concepts (mindsets)
- Practices
- Recommendations
- Meta-model
- Implementation for MSF Agile
- Implementation for MSF Formal

MSF basic principles

- Partnership with client
- Foster open communication
- Work toward a shared vision
- Quality is everyday work for everyone (invest in quality)
- Stay agile, expect change
- Make implementation a habit
- Create value (focus on delivering business value)

MSF basic principles-2

- Distribution of responsibility, fixed statements
- Empower team members
- Focus on business priorities
- Single vision of the project
- Be flexible - be prepared for change
- Encourage free communication

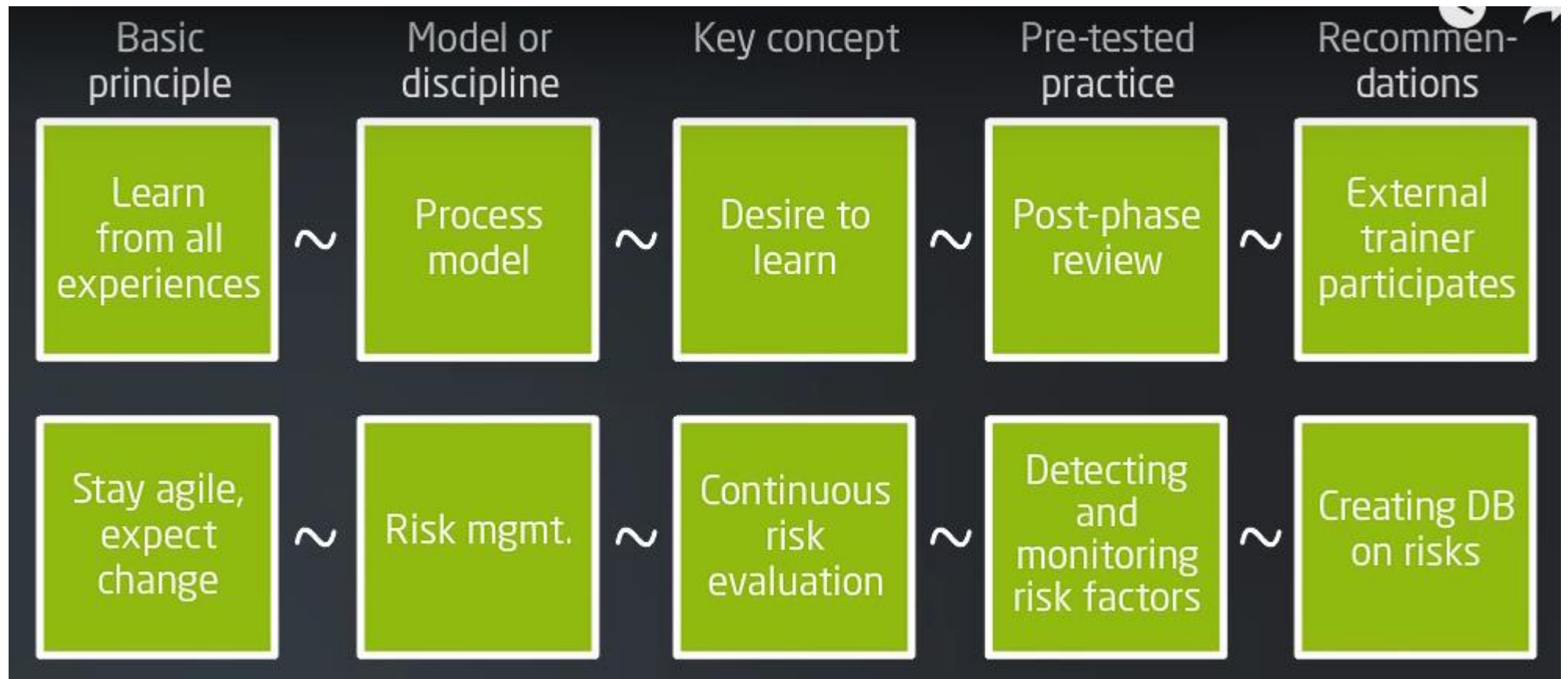
MSF roles

- Program management
- Product management
- Development
- Testing
- Release management
- Customer satisfaction

MSF project group

- Program manager
- Product manager
- Developer
- Quality Assurance
- Release manager
- User experience

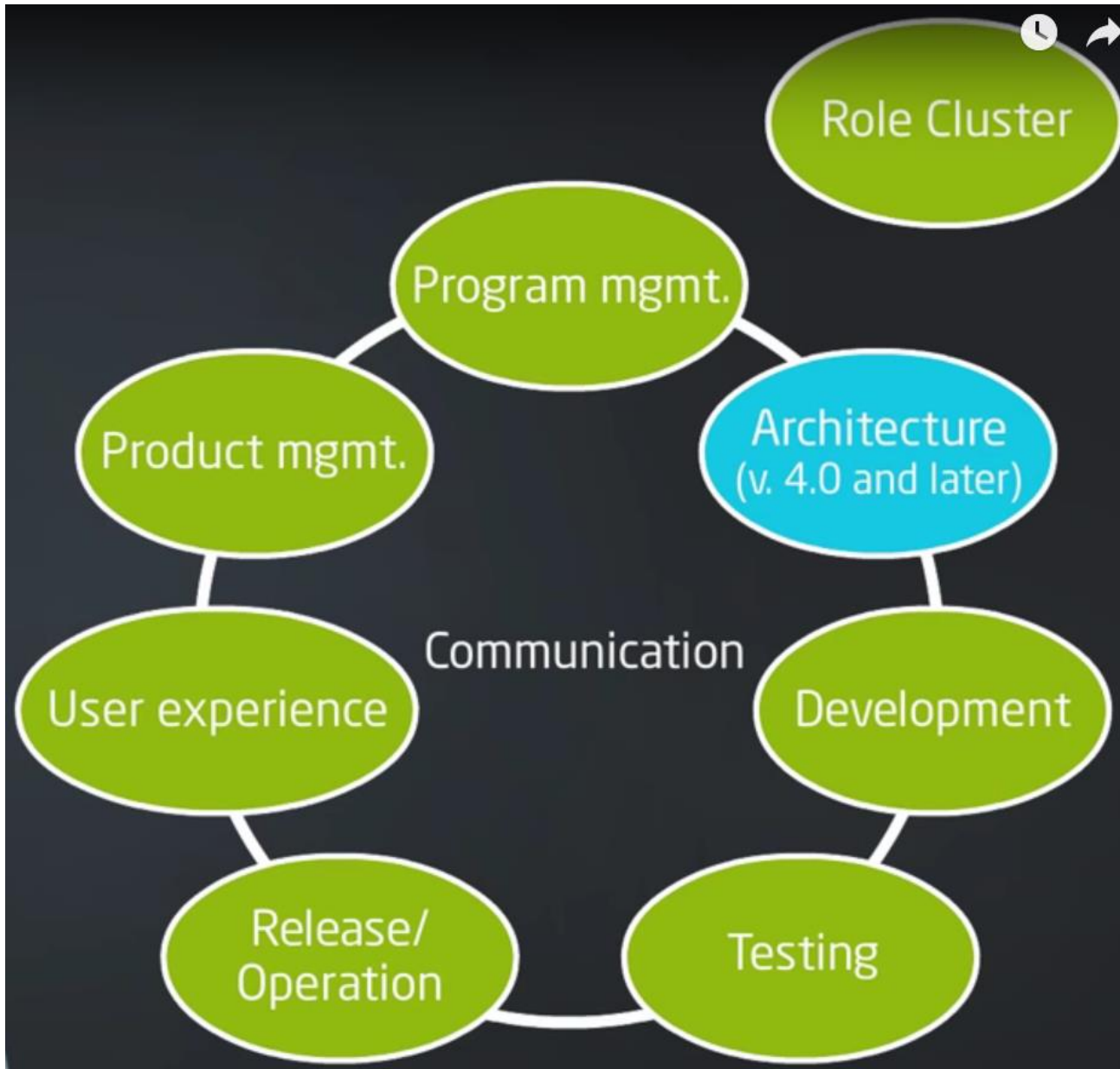
MSF: Elements and Relationships



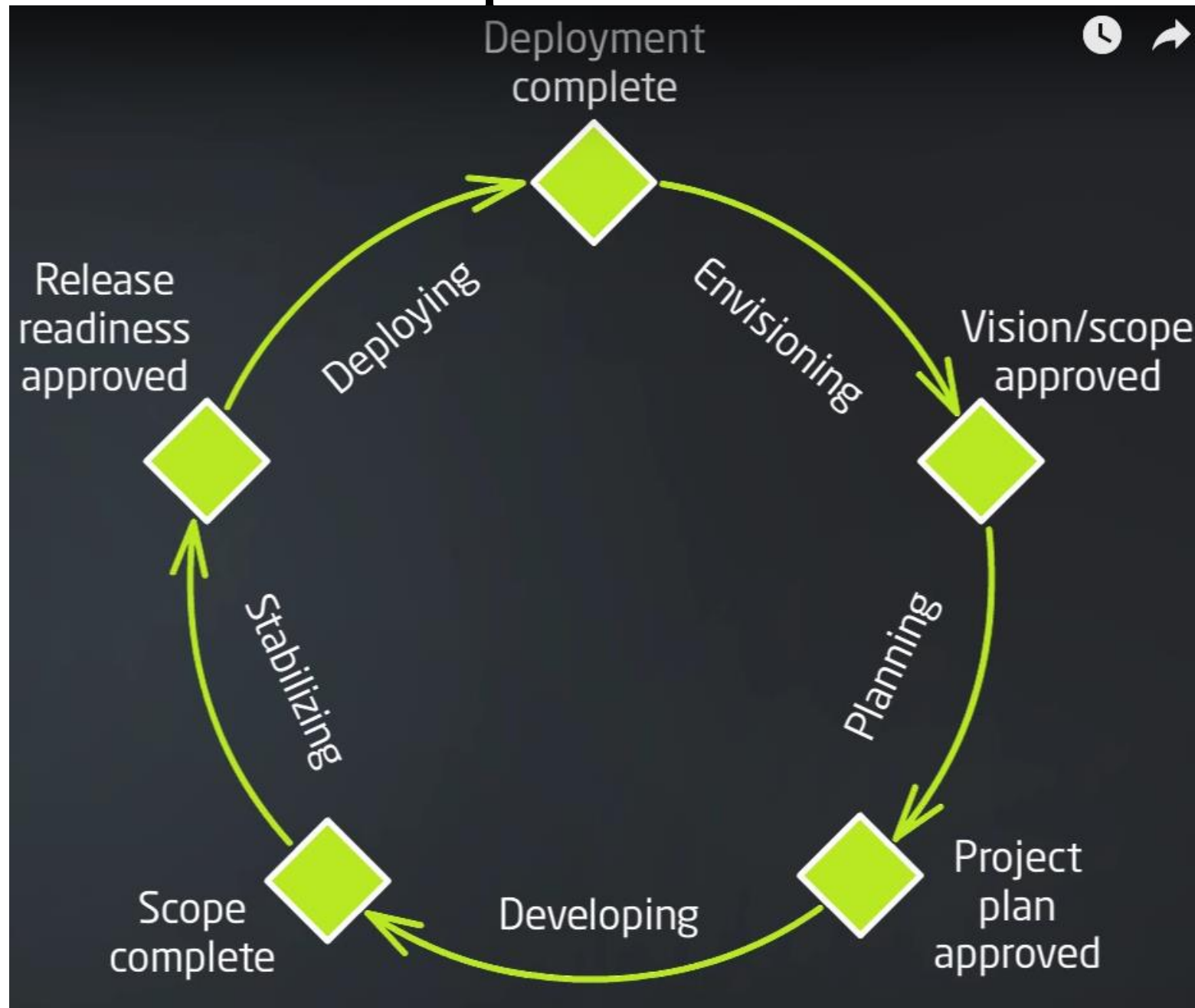
MSF: teamwork principles

- A team of equal
- Representing interests of all interested sides
- Adjust to match project scale («teams of teams»)

MSF: team model



MSF: process model



MSF: Role Compatibility Matrix

	Arc	MPrd	MPrg	Dev	Tst	UX	RM
Architecture		N	P	P	L	L	L
Product mpmt.			N	N	P	P	L
Program mpmt.				N	L	L	P
Development					N	N	N
Testing						P	P
User experience							L
Release mgmt.							

N – not recommended; P – possible; L – low probability