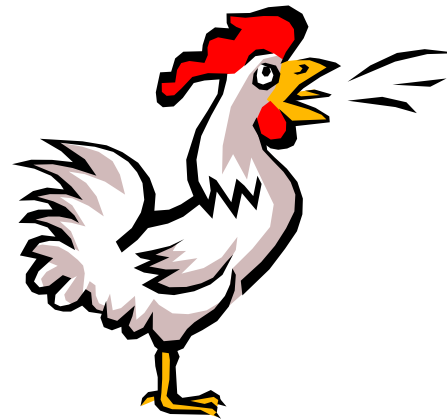
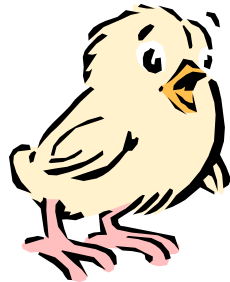


Iterative Development of Large Software-Intensive Systems



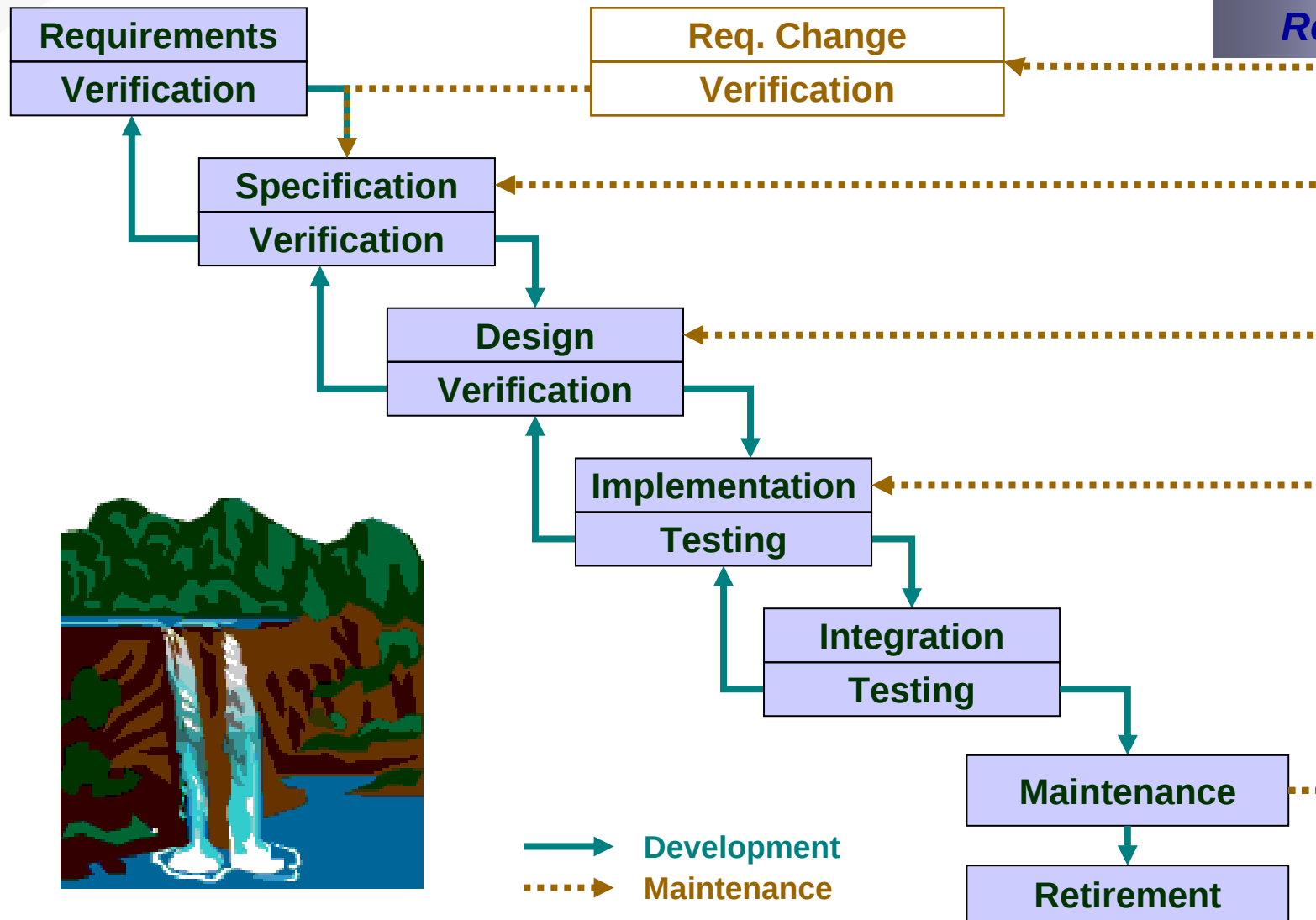
Ely Bonne,
Boaz Shani,
Amir Tomer

- **What's wrong with the waterfall model?**
- **The Unified Software Development Process**
- **USDP adaptation for RAFAEL**
 - **Process stages**
 - Activities
 - Work products
 - Reviews
- **Conclusions**



The Waterfall Model

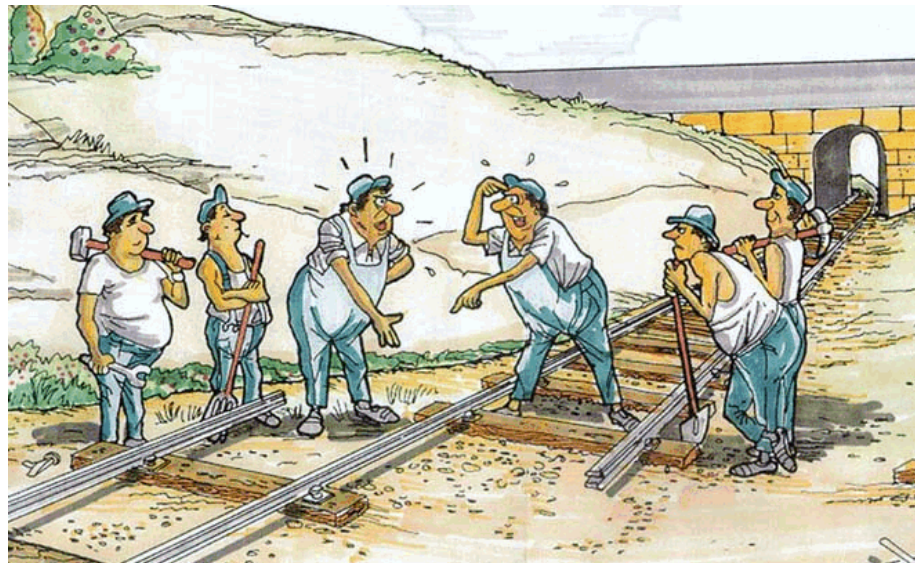
Royce, 1970



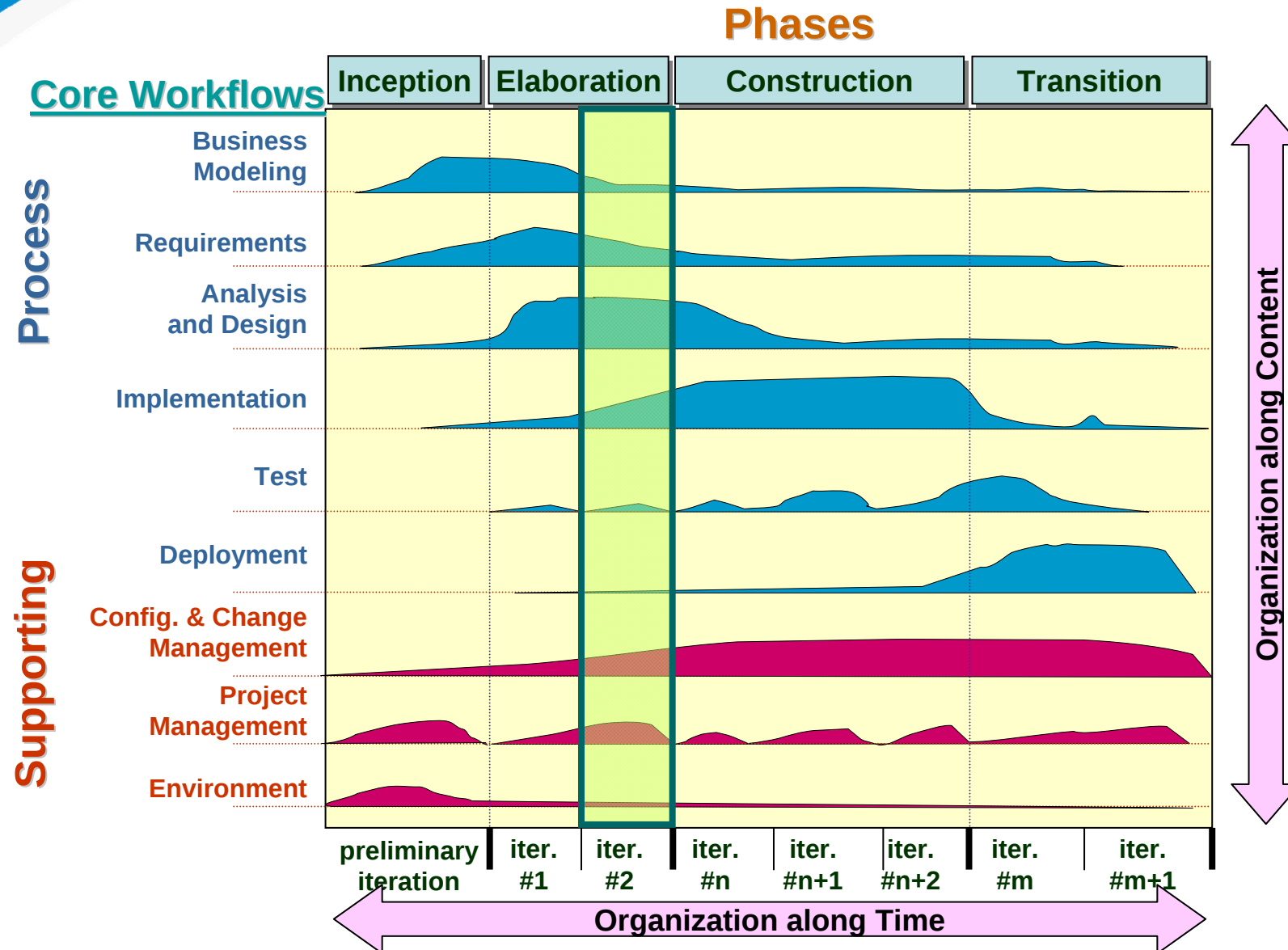
What's wrong with the Waterfall model?

- Risk mitigation postponed until late stages
- Document-based verification until late stages
- Attempt to stipulate unstable requirements too early
- Operational problems discovered too late
- Lengthy modification cycles and much rework

The
inevitable
result...



The Unified Software Development Process/ RUP SE



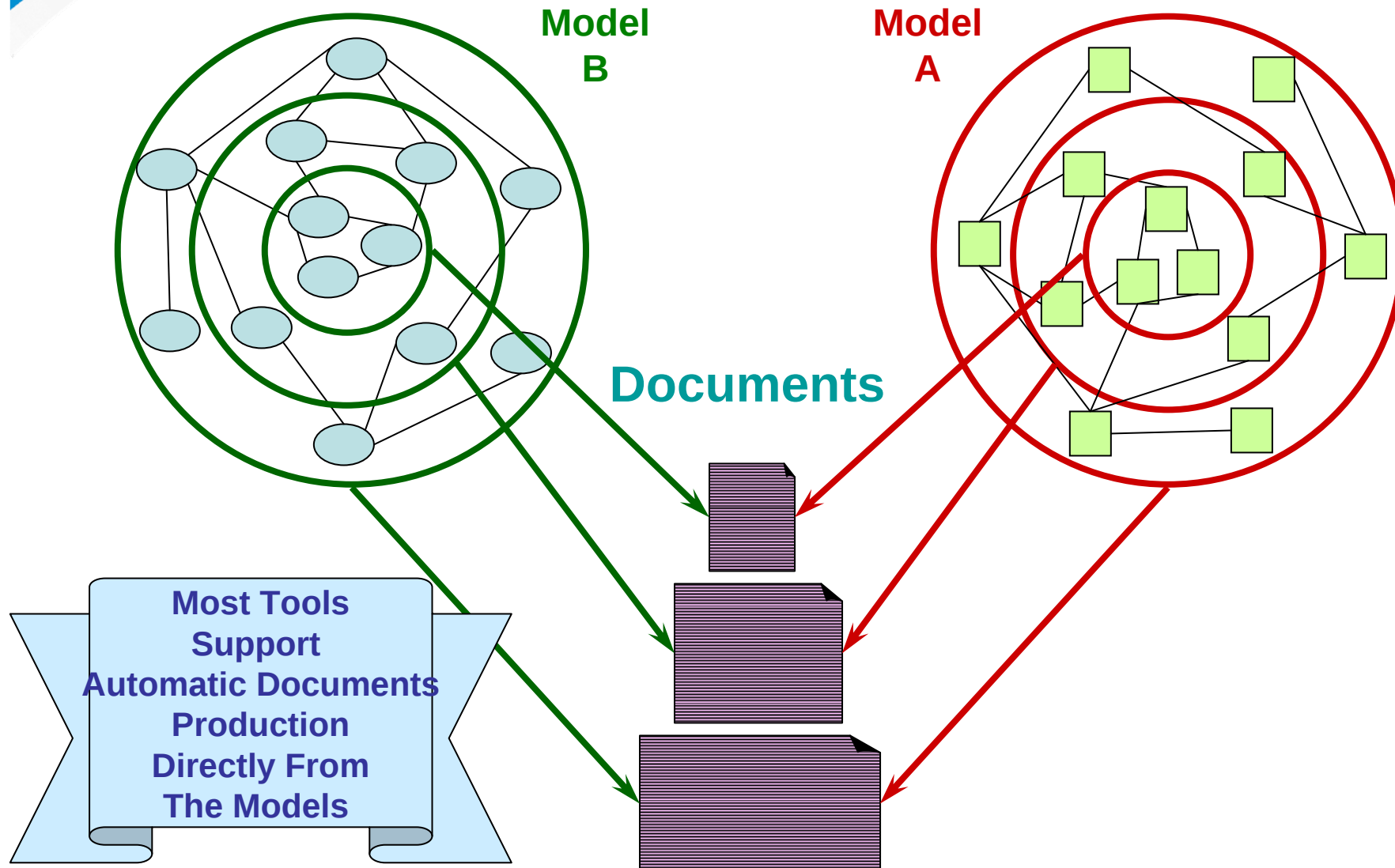
Source:
P. Kruchten, *The Rational Unified Process*,
Addison-Wesley, Reading, MA, 1998.

Our Process vs. USDP

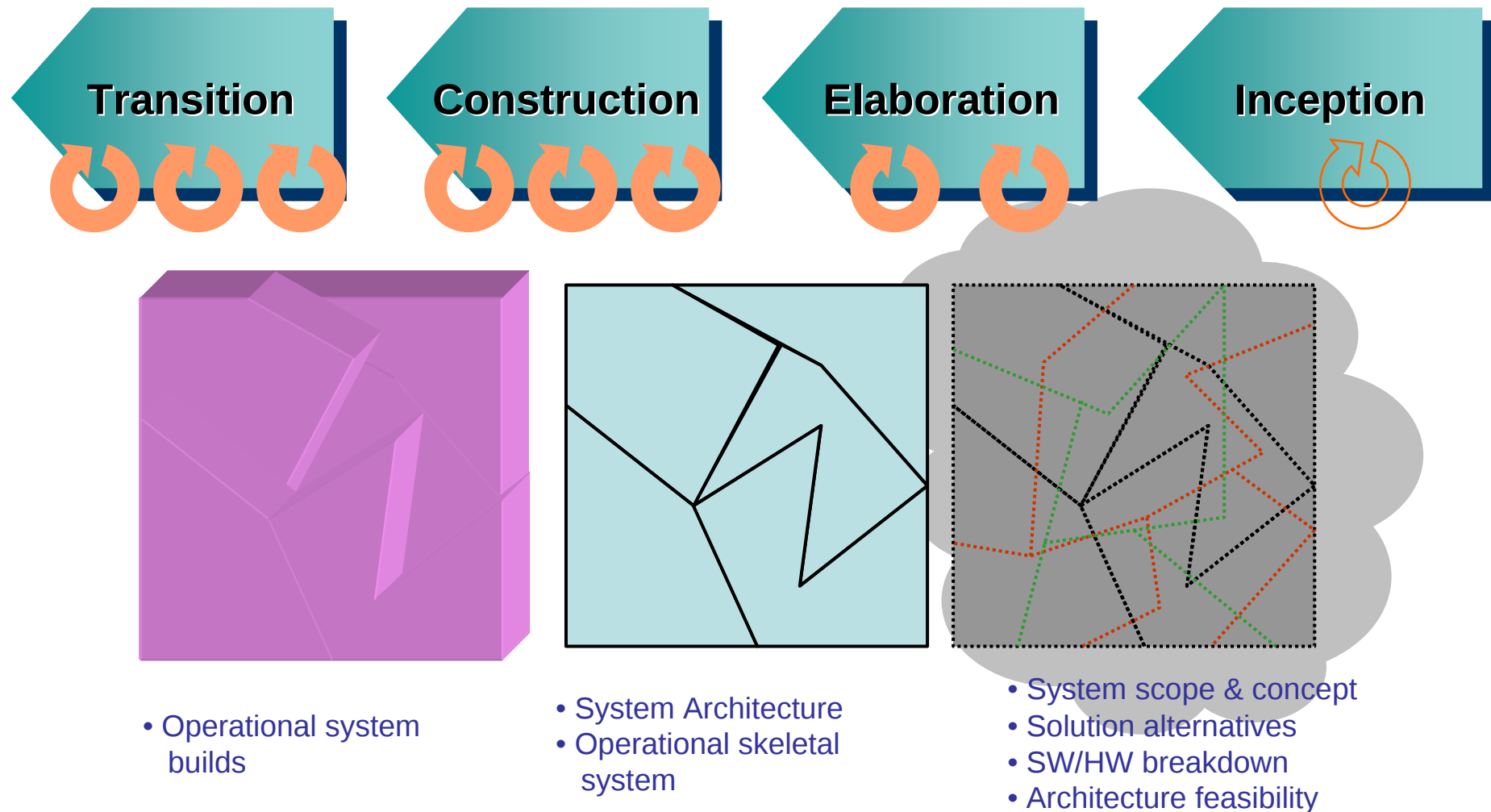
- **An "end to end" process covering both system and software engineering disciplines, defining the scope and work products of each and how they co-exist.**
 - Defines precisely the development stages, their work products and reviews.
- **Adjusted to large software-intensive systems, for government customers:**
 - fixed-price contract.
 - early definition of all system requirements.
 - continuous progress control.
 - compatibility to military standards.
 - compliance to CMM level 3.



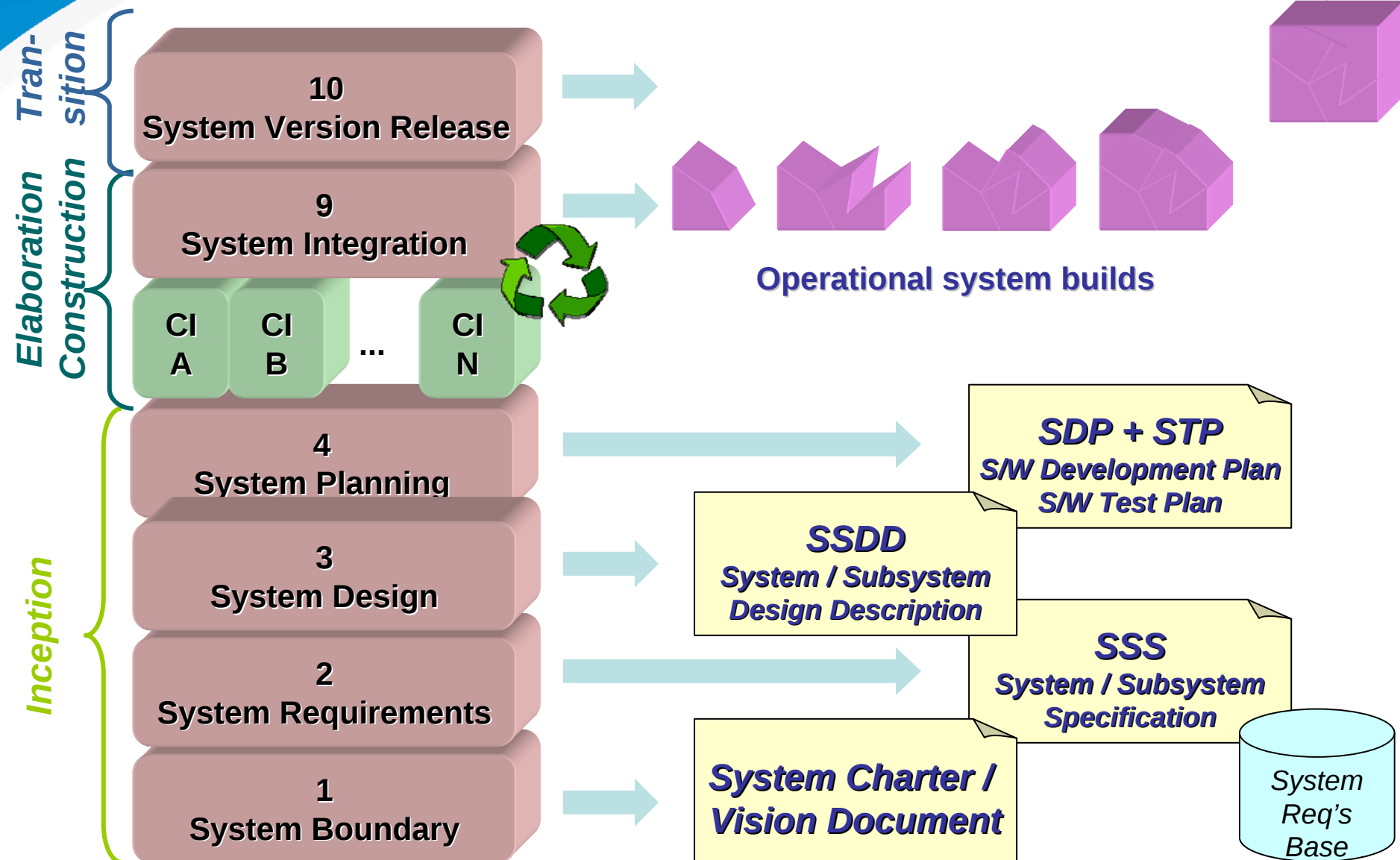
Models Based Development Using CASE Tools



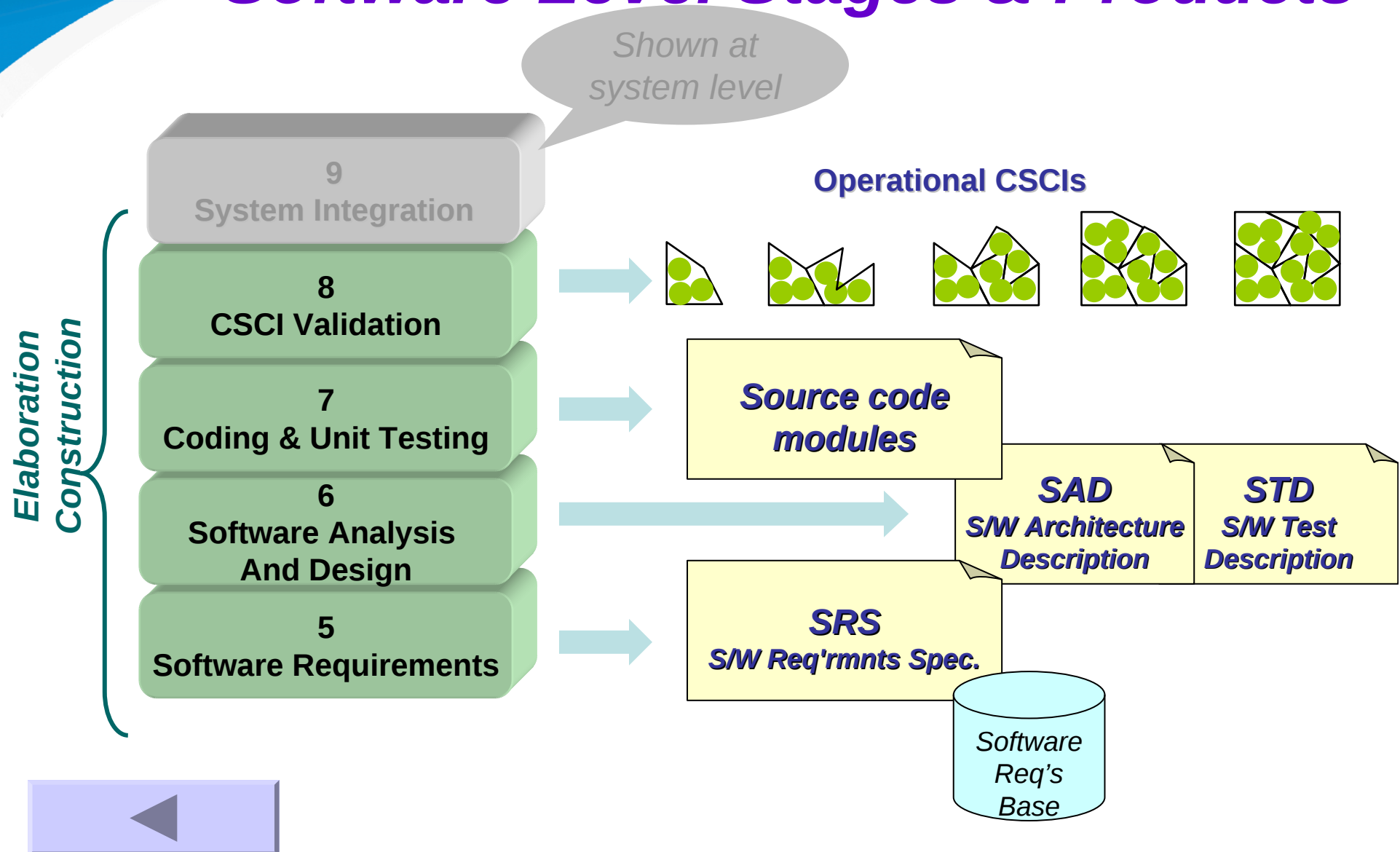
Iterative Development Overview



System Level Stages & Products



Software Level Stages & Products



Stage 1: System Boundary

Level: System
Phase: Inception

- **Purpose**
 - Provide a common understanding about the system
 - client
 - developers
 - other stakeholders
- **Activities**
 - Scope
 - Main purposes
 - Relationship with external systems
 - Main requirements
- **Products**
 - System charter / Vision document
 - Glossary (domain terms)



Stage 2: System Requirements

Level: System
Phase: Inception

- **Purpose**
 - Define system requirements
 - at the best level of knowledge
- **Activities**
 - Requirements elicitation from contract, proposal and other documents
 - Requirements classification and prioritization
- **Products**
 - SSS = System/Subsystem Specification
 - Use Case modeling recommended
 - Requirements base



Stage 3: System Design

Level: System
Phase: Inception

- **Purpose**
 - Provide a stable proposed solution and system architecture
- **Activities**
 - System architectural design
 - System breakdown into components (HW, SW, . . .)
 - including requirements allocation
 - Feasibility tests to select alternatives
- **Products**
 - SSDD = System/Subsystem Design Description
 - ICD = Interface Control Description
 - External / internal interfaces
 - Either separate or included in SSDD



Stage 4: System Planning

Level: System
Phase: Inception

- **Purpose**
 - Generating a general plan for the System development
- **Activities**
 - Risk Analysis
 - Iteration planning
 - General plan for following iterations
 - Activities, products, reviews
 - Risk allocation to iterations
 - Resource allocation
 - Testing concept and general planning
- **Products**
 - SDP = Software Development Plan / **SEMP**
 - STP = System / Software Test Plan



Stage 5: Software Requirements

Level: CSCI

Phase: Elab./Cons.

- **Purpose**
 - Provide a clear and detailed definition of the software requirements allocated to the appropriate CSCI
 - At the current iteration level
- **Activities**
 - Deriving software requirements from system requirements
 - Building/updating the Use Case model
- **Products**
 - Use Case model
 - Using CASE tools
 - SRS = Software Requirements Specification
 - Functional requirements (Use Cases)
 - Non-functional requirements
 - Software requirements base
 - Preferably as part of the system requirements base



Stage 6: Software Analysis & Design

Level: CSCI

Phase: Elab./Cons.

- **Purpose**

- Provide an architecture and design, in various aspects, of the current iteration
 - based on previous iterations + current requirements

- **Activities**

- Analysis and design modeling
 - Using UML models

- **Products**

- UML models
 - Class model
 - State charts
 - Sequence diagrams
- SAD = Software Architecture Description
- STD = Software Test Description

Detailed design

Top level design



Stage 7: Coding and Unit Testing

Level: CSCI

Phase: Elab./Cons.

- **Purpose**
 - Implementation of software modules
- **Activities**
 - Coding
 - Programming, compilation, link
 - Individual module testing
 - Informal documentation
 - Test coverage and completion approved by STL
- **Products**
 - Approved code modules



Stage 8: CSCI Validation and Approval

Level: CSCI

Phase: Elab./Cons.

- **Purpose**
 - Approve the CSCI's readiness for system integration
- **Activities**
 - Integrating the modules comprising the CSCI
 - Performing the tests specified in the STD
- **Products**
 - Approved CSCI version, ready for system integration
 - STR = Software Test Report
 - Detailed results of STD tests



Stage 9: System Integration and Testing

Level: System
Phase: Elab./Cons.

- **Purpose**
 - Accomplish the iteration with an operational approved version of the system
- **Activities**
 - Integration
 - System testing, according to system test specifications
- **Products**
 - Approved operational version of the system, ready for delivery



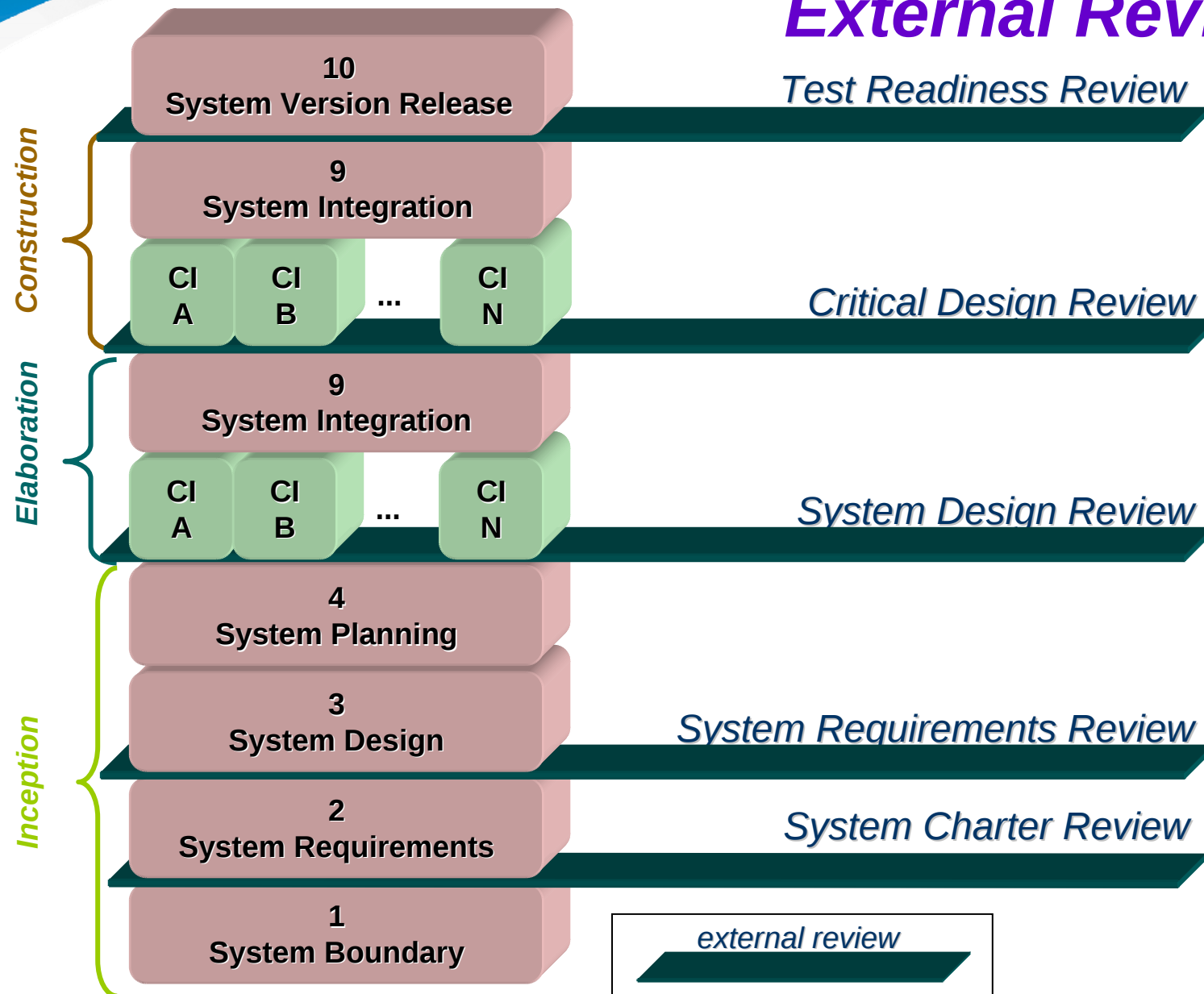
Stage 10: System Version Release

Level: System
Phase: Elab./Cons.

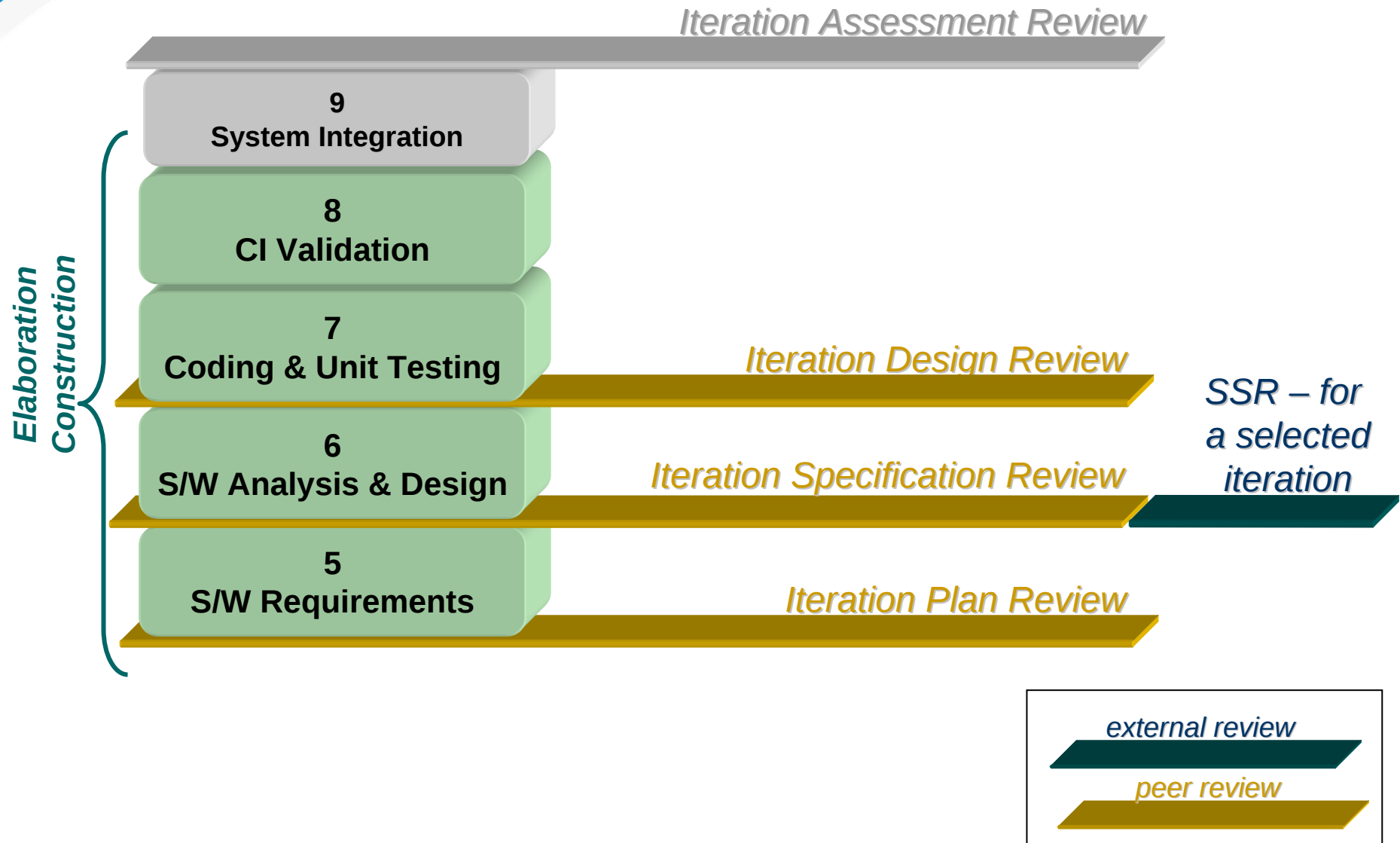
- **Purpose**
 - Deploy an approved operational system version to the customer
- **Activities**
 - FAT
 - System installation at customer's site
 - SAT
 - ILS activities
- **Products**
 - Approved version of the system, operational at customer's site



External Reviews



Peer Reviews



Conclusions

- USDP-based iterative software-intensive system development has many advantages over the waterfall model
- The iterative process may be adapted and tailored to host most of MIL-STD-498 and CMM/CMMI terminology
- The presented iterative process complies with RAFAEL software development procedures
- The tailored process is well accepted by engineers, managers and customers
- The process appears to be suited for the development of multi-disciplinary systems as well



Contacts

Boaz Shani
Senior C³I Systems Engineer
RAFAEL
P.O.Box 2250/23
Haifa, 31021
ISRAEL
(+972)-4-879-4220
boazsh@rafael.co.il

Ely Bonne
Head of Software
Engineering Section
RAFAEL
P.O.Box 2250/8H100
Haifa, 31021
ISRAEL
(+972)-4-879-2243
elyb@rafael.co.il

Dr. Amir Tomer
Director of Systems & Software
Engineering Processes
RAFAEL
P.O.Box 2250/1P
Haifa, 31021
ISRAEL
(+972)-4-879-5989
tomera@rafael.co.il