Software Development Methodologies

Lecturer: **Raman Ramsin**

Lecture 8

Situational Method Engineering

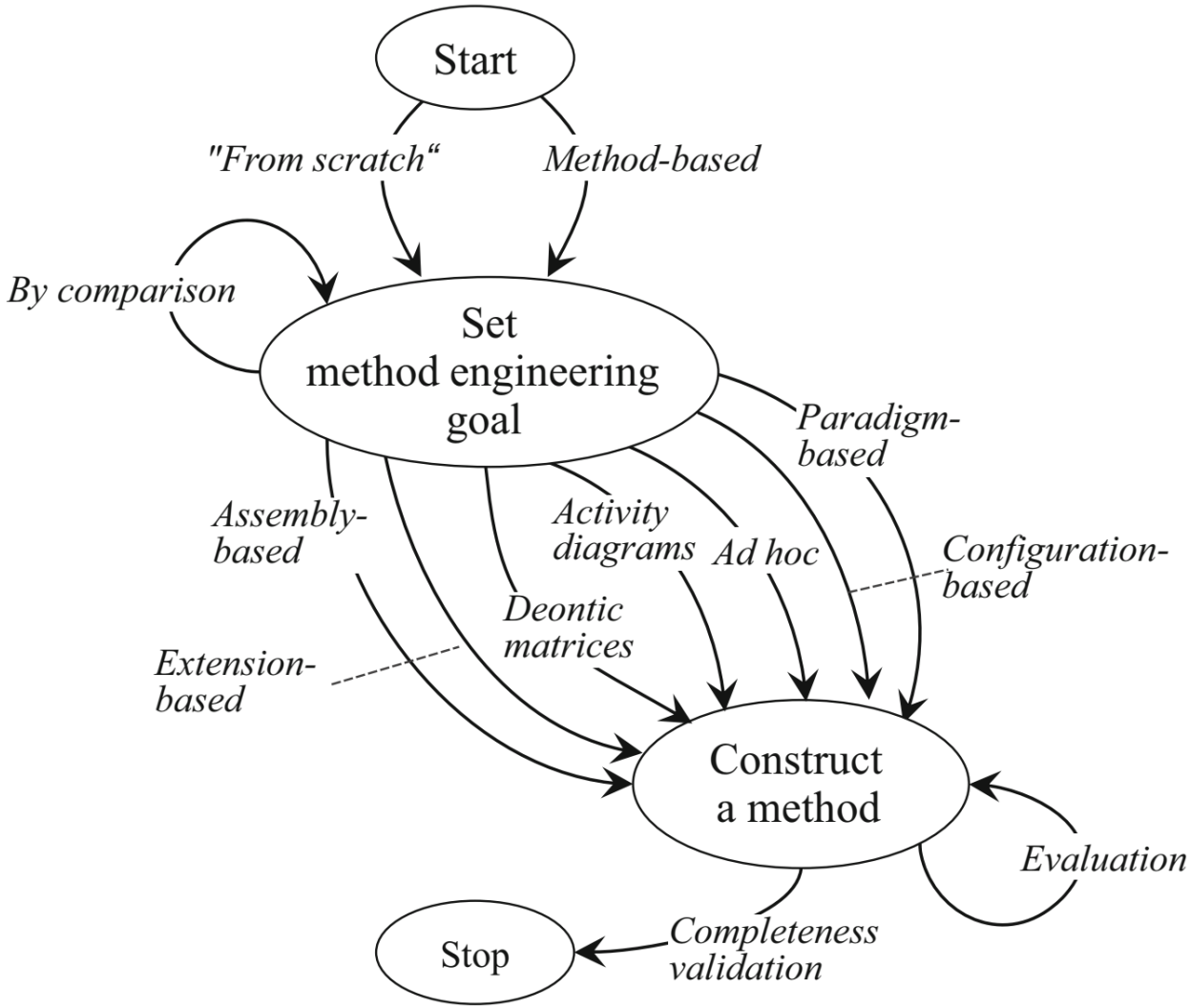
Methodology Engineering

- Motivated by the prevalent belief that no one methodology fits all situations.
- First introduced by Kumar and Welke in 1992 as a discipline aimed at constructing methodologies to match given organizational settings or specific development projects.
- Later came to be known as *Method Engineering*, a term proposed by Brinkkemper in 1996;
  - Definition: "The engineering discipline to design, construct, and adapt methods, techniques and tools for the development of information systems".
- The most well-known subfield is *Situational* Method Engineering (SME): concerned with the construction/adaptation of a methodology specifically attuned to the project at hand.

Methodology Engineering: Alternative Approaches

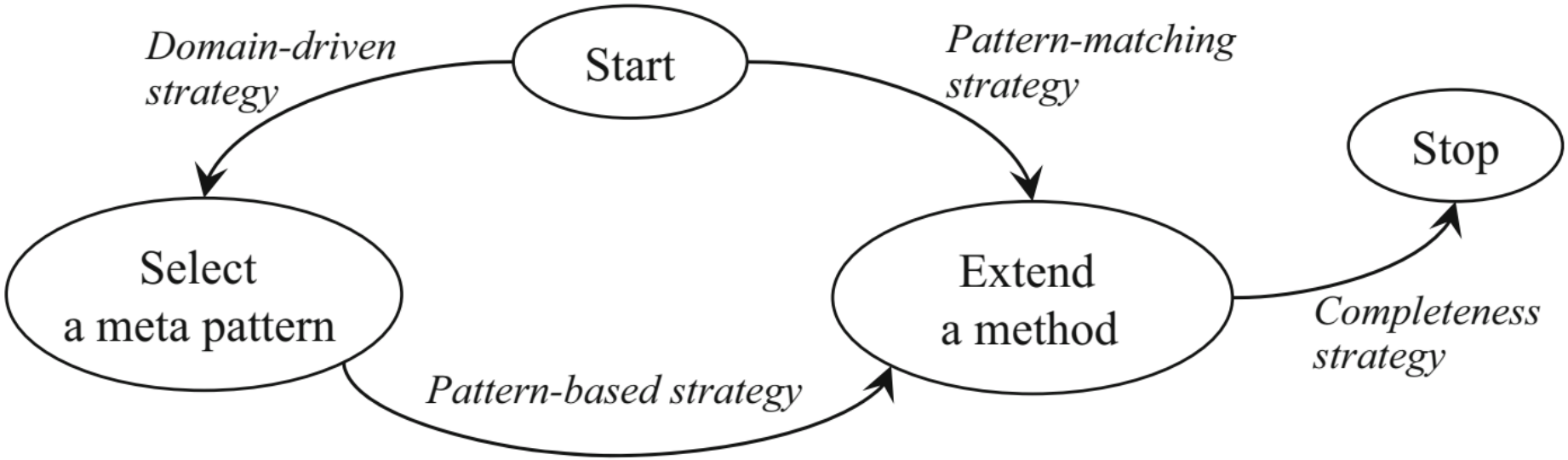
- ***Ad-hoc:*** Concerned with constructing a new methodology from scratch.
- ***Configuration-based:*** Concerned with configuring methodologies and method components based on the situational context at hand.
- ***Activity-diagram-based:*** Concerned with application of UML Activity Diagrams for representing process-focused actions by Activities.
- ***Deontic-matrices-based:*** Concerned with using matrices of values for representing relationships of Activity-Task pairs chosen from OPF repository.
- ***Paradigm-based:*** Concerned with instantiating, abstracting or adapting an existing meta-model in order to produce the target methodology.
- ***Extension-based:*** Concerned with enhancing an existing methodology with new concepts and properties by using extension patterns.
- ***Assembly-based:*** Concerned with constructing the target methodology or enhancing an existing methodology by reusing parts of other methodologies.

Generic Model for Situational Method Engineering



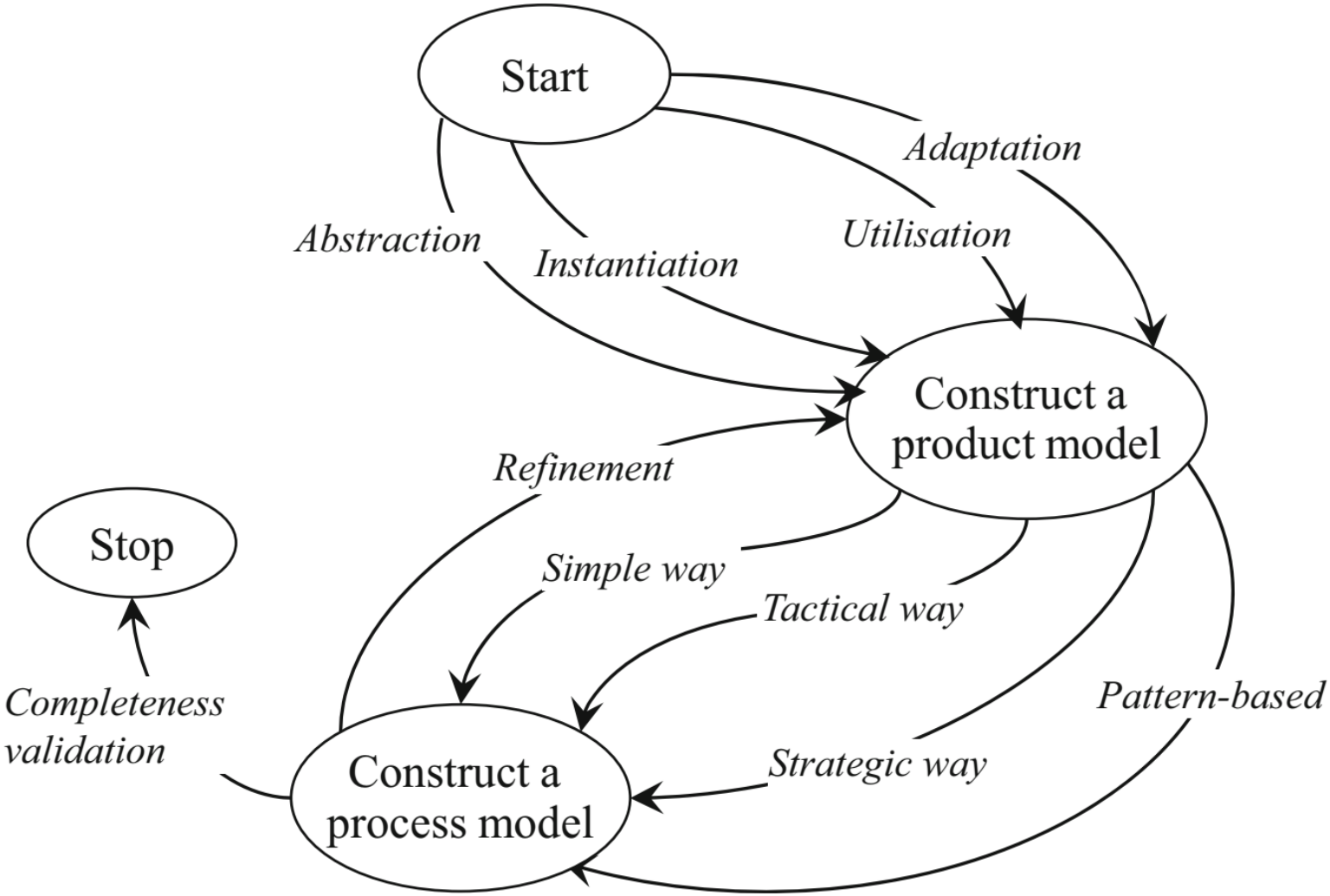
[Henderson-Sellers et al. 2014]

Extension-Based SME



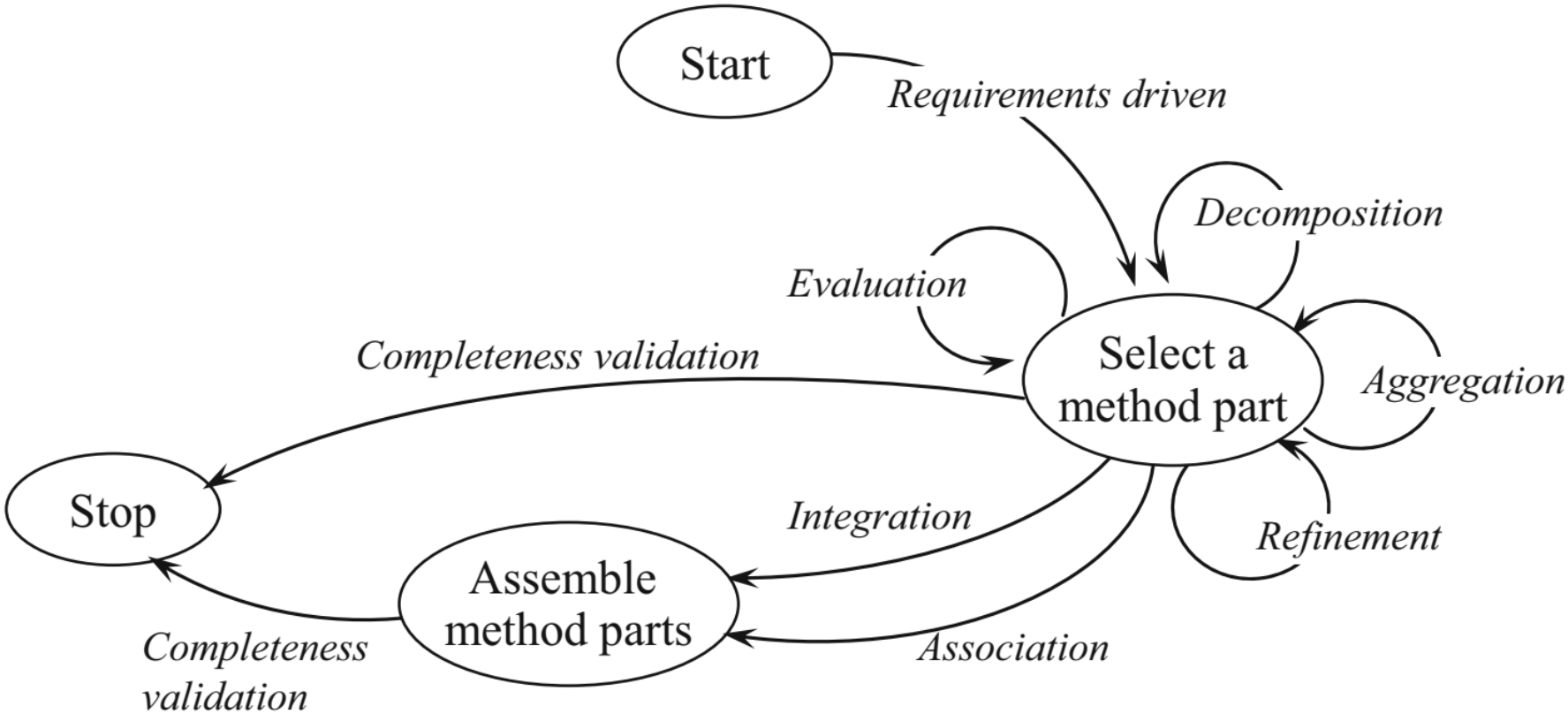
[Henderson-Sellers et al. 2014]

Paradigm-Based SME



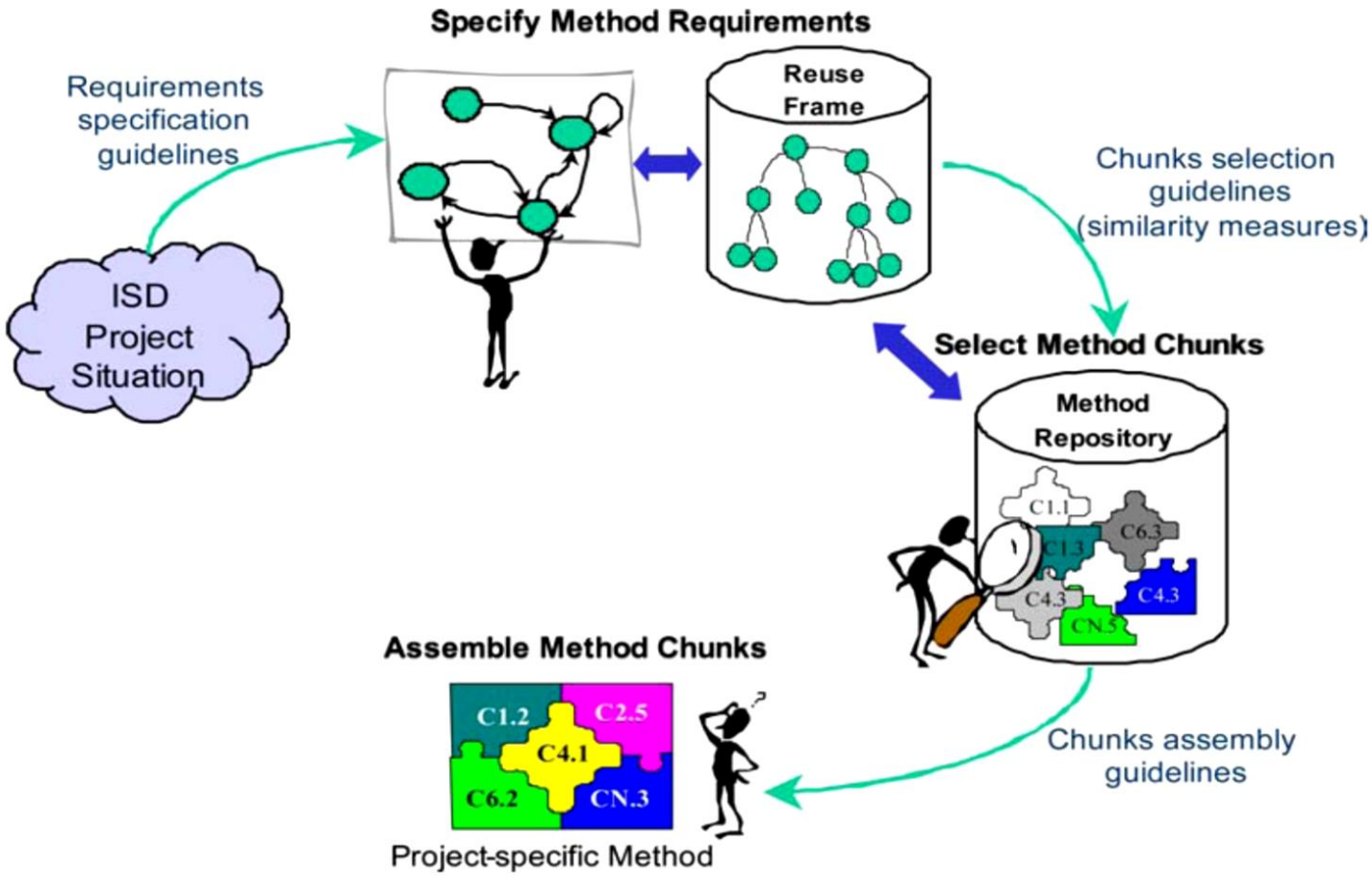
[Henderson-Sellers et al. 2014]

Assembly-Based SME



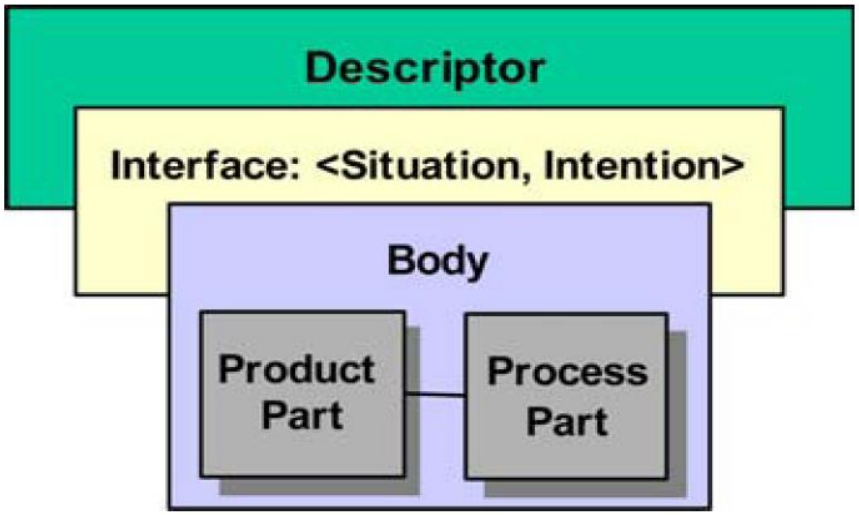
[Henderson-Sellers et al. 2014]

Assembly-Based SME



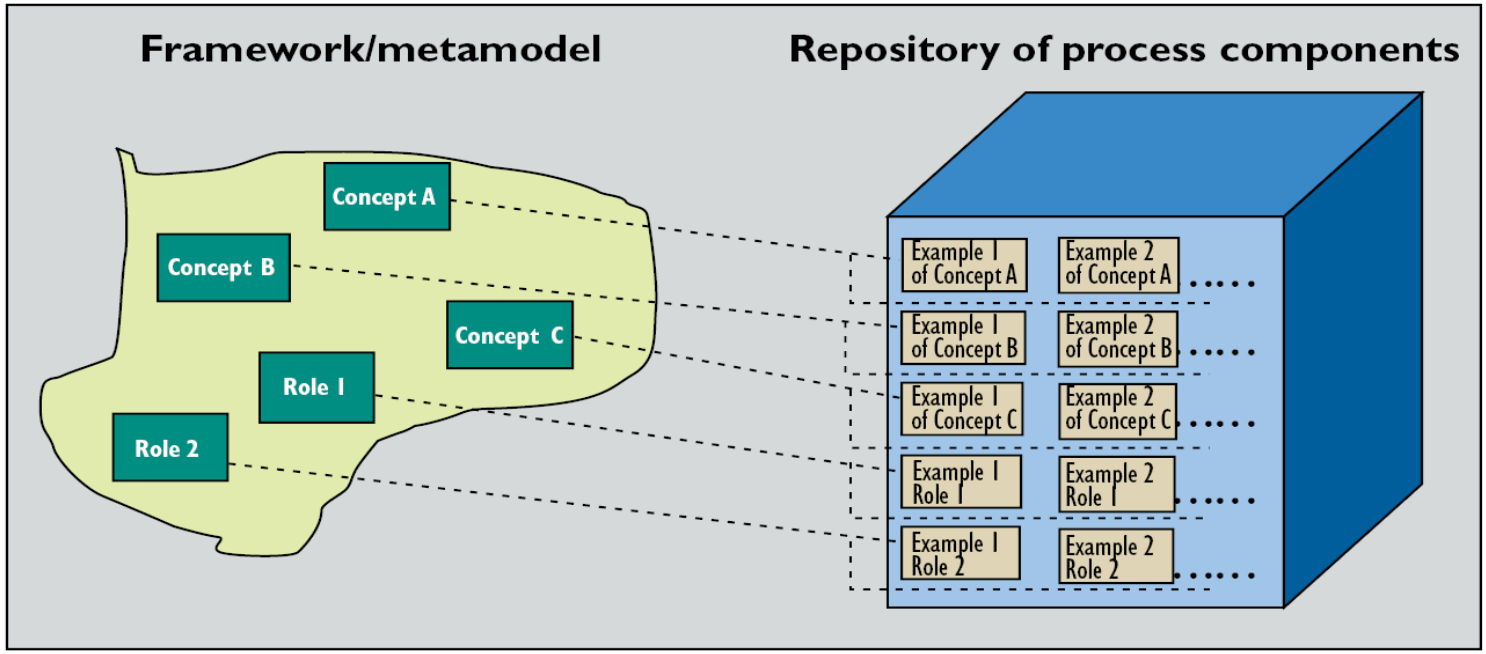
[Mirbel and Ralyté 2006]

Method Chunk



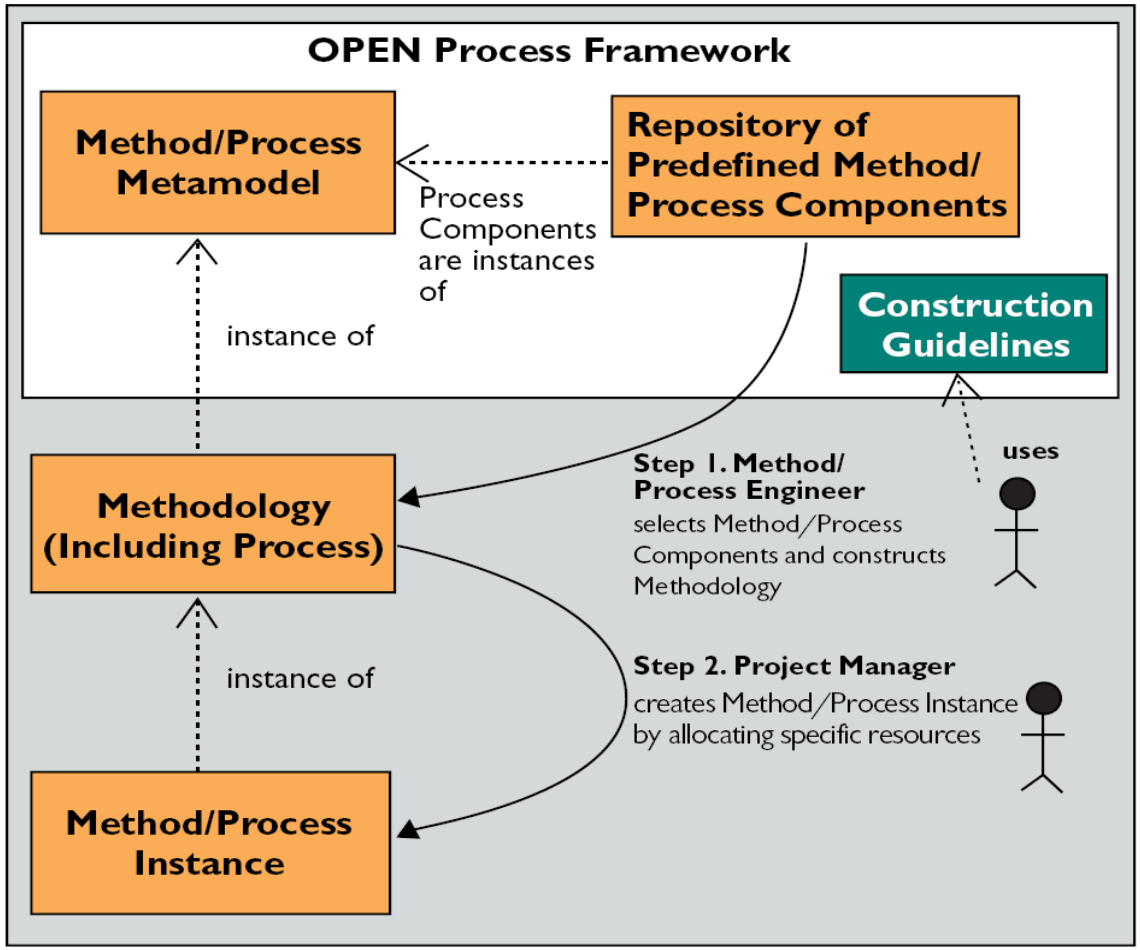
[Mirbel and Ralyté 2006]

Assembly-Based ME In OPEN/OPF



[Henderson-Sellers 2003]

Assembly-Based ME In OPEN/OPF



[Henderson-Sellers 2003]

OPF Repository

- Contains a range of predefined instances for each class and subclass in the OPF metamodel; e.g.:
  - 30 predefined instances of Activity
  - 160 instances of Task
  - 200 instances of Techniques
  - 76 instances of Role

OPF: Task-Activity Matrix

Task	Activity					
	1	2	3	4	5	6
Code		x				
Construct the object model		x				x
Develop and implement resource allocation plan						
develop iteration plan	x					
develop timebox plan	x					
set up metrics collection program	x					
specify quality goals	x					
Evaluate quality			x	x		x
Identify CIRTs (Class, Instance, Role, or Type)		x				
Map roles onto classes		x(OOP)				
Test			x	x	x	
Write manuals and other documentation			x	x	x	x

key

1. Project planning
2. Modeling and implementation: OO analysis, design, programming
3. Verification and validation
4. User review
5. Consolidation
6. Evaluation

[Henderson-Sellers 2003]

OPF: Technique-Task Matrix

Techniques	Task 1	2	3	4	5	6	7
Abstract class identification		x					
Abstraction utilization		x			x		
Class internal design	x	x					
Class naming		x		x			
Collaborations analysis		x					
Complexity measurement				x			
Contract specification	x	x			x		
Coupling measurement				x			
CRC card modeling		x			x		
Generalization and inheritance identification		x					x
Implementation of services	x						
Implementation of structure	x					x	
Inspections				x			x
Interaction modeling		x					
Package and subsystem testing							x
Prototyping	x	x					
Relationship modeling	x	x					
Responsibility identification	x			x	x		x
Role modeling		x				x	
Service identification		x					
State modeling		x					
Textual analysis					x		
Timeboxing			x				
Unit testing							x
Walkthroughs				x			x

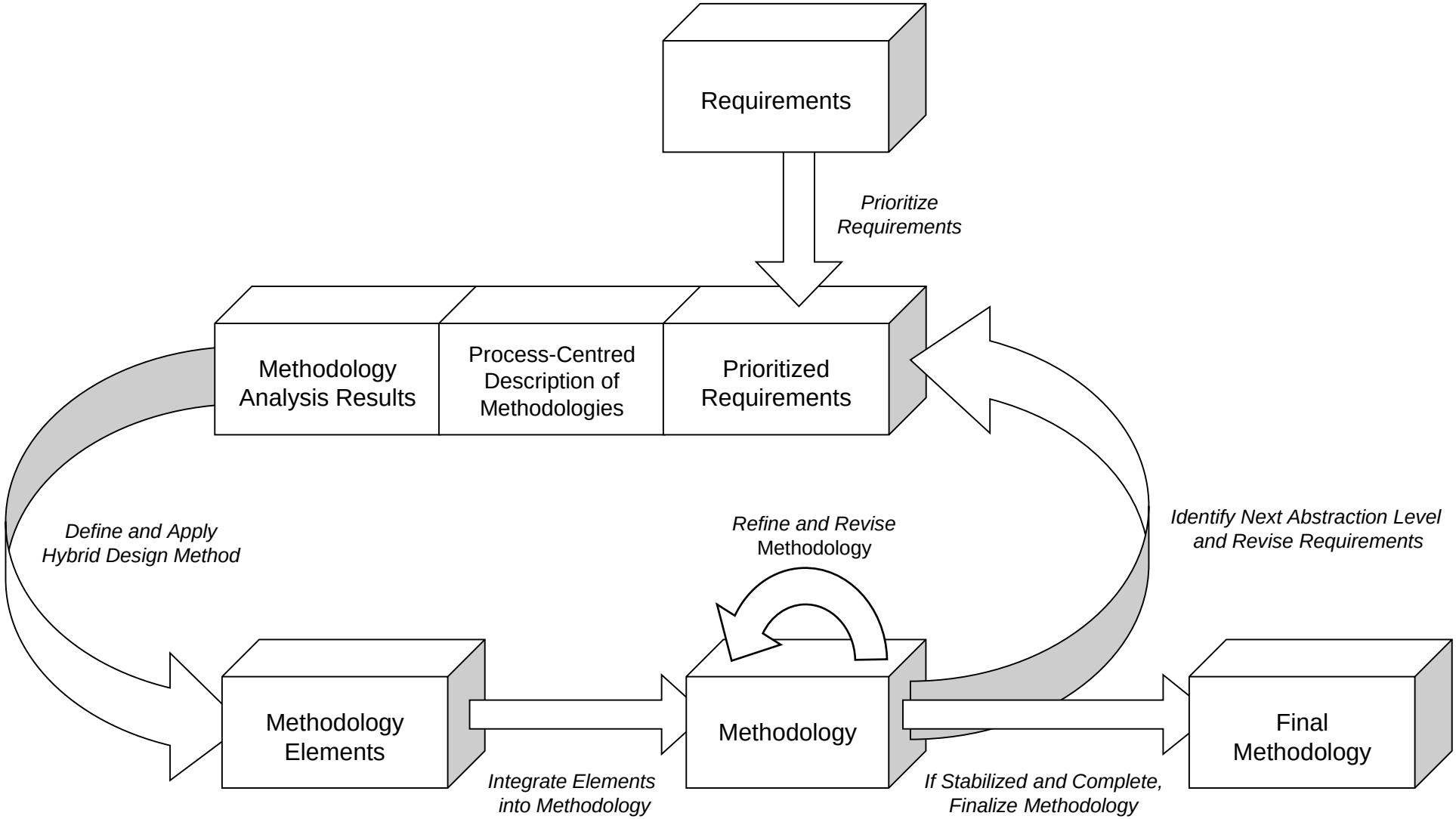
key
1. Code
2. Construct the object model
3. Develop and implement resource allocation plan
4. Evaluate quality
5. Identify CIRTs
6. Map roles onto classes
7. Test

[Henderson-Sellers 2003]

Hybrid Methodology Design

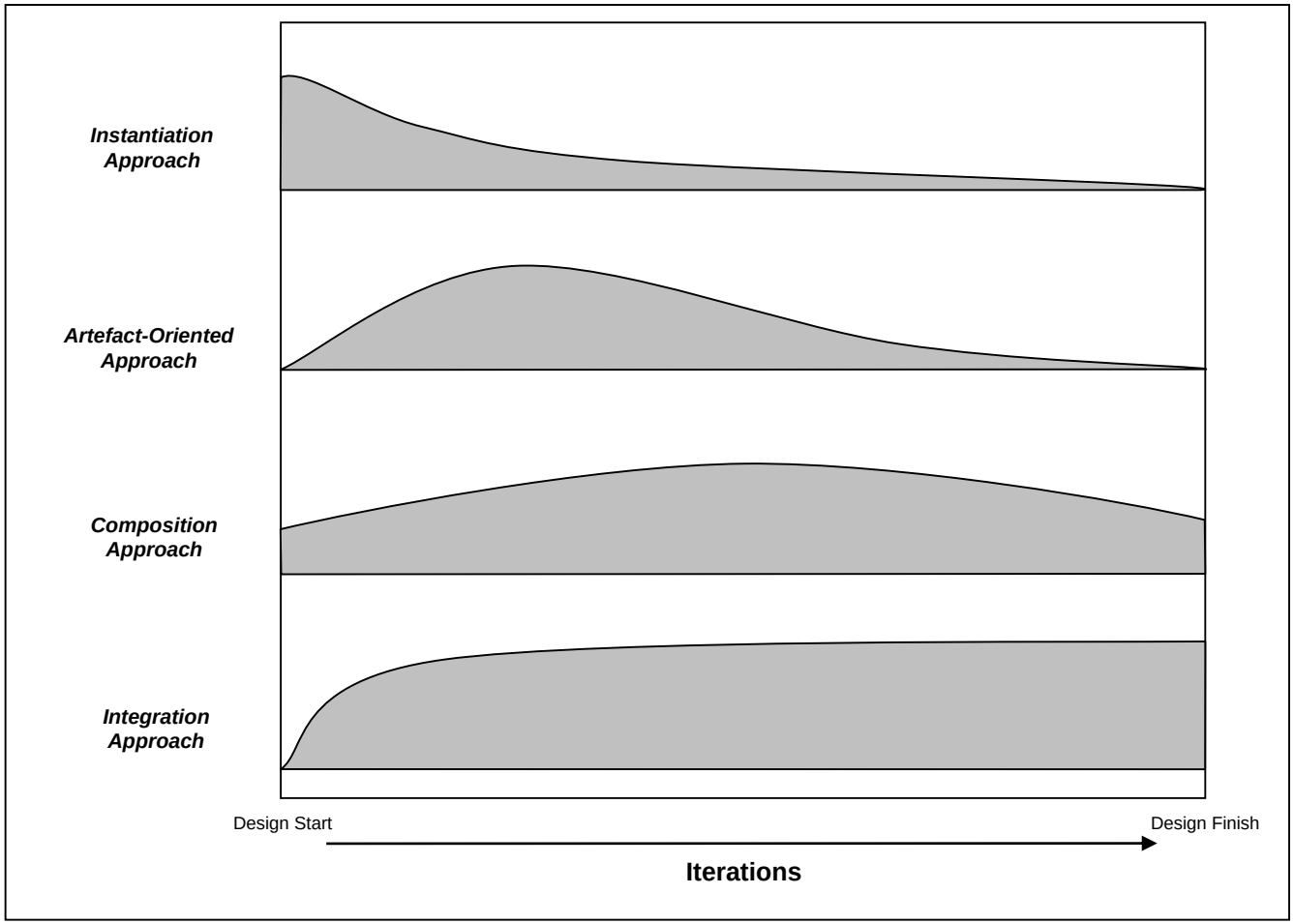
- Alternative Approaches:
  - *Instantiation approach:* instantiating an already available process metamodel
  - *Artefact-oriented approach:* devising a seamless complementary chain of artefacts and building the process around it
  - *Composition approach:* using one of the already available libraries of process patterns
  - *Integration approach:* integrating features, ideas and techniques from existing methodologies
- *Hybrid design approach:* using different alternatives from among the above-mentioned for different parts of the process and/or at different levels of abstraction

Hybrid Design Process



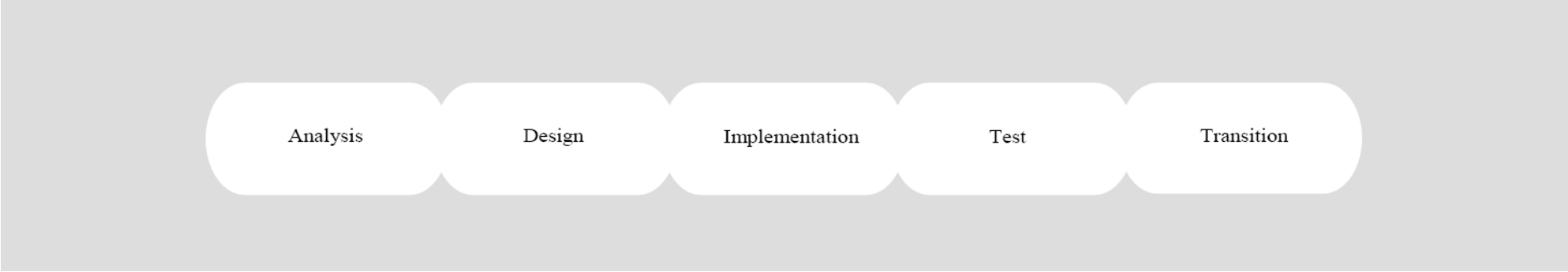
[Ramsin 2006]

Hybrid Design: Emphasis on Approaches during Design



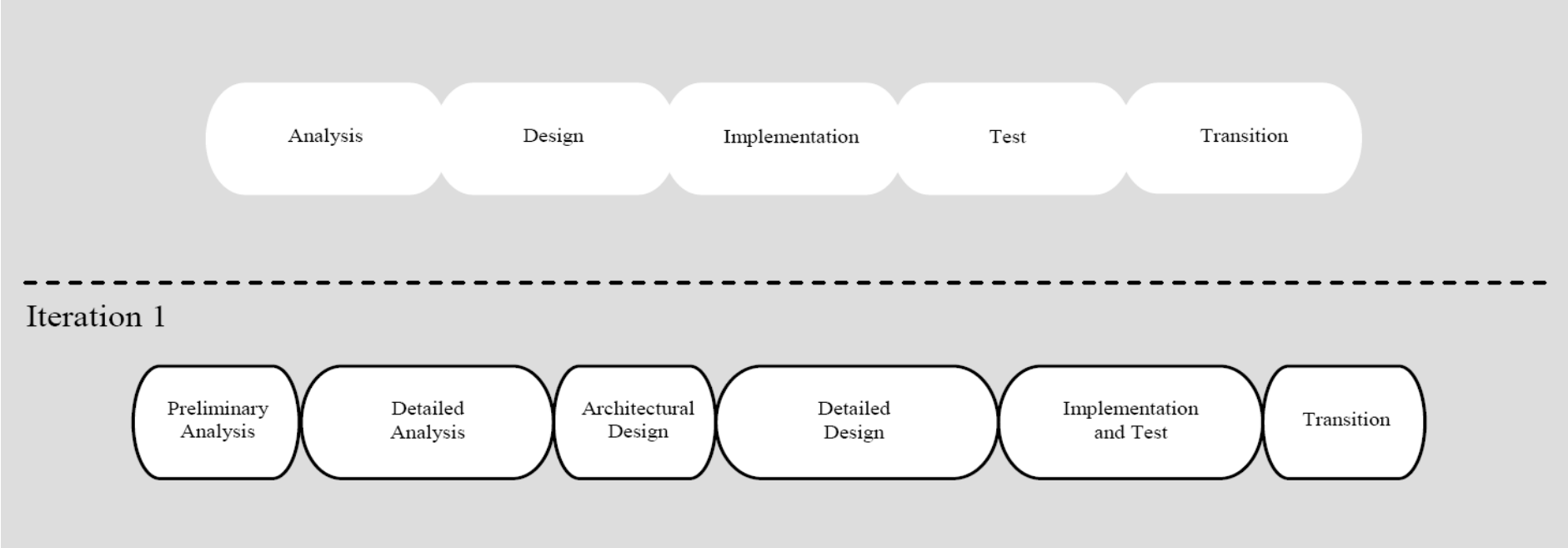
[Ramsin 2006]

Hybrid Design: Sample Iterations



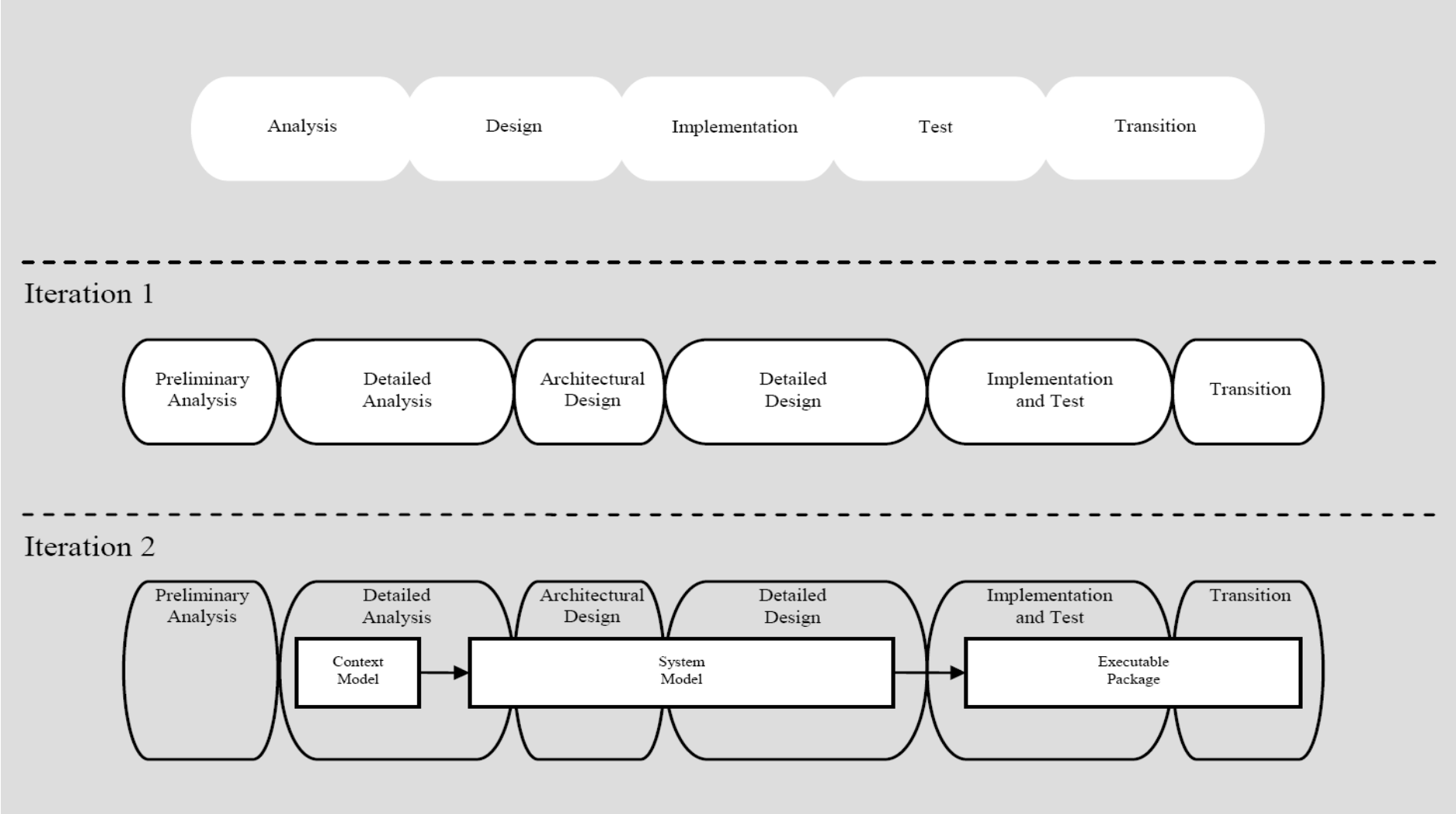
[Ramsin 2006]

Hybrid Design: Sample Iterations



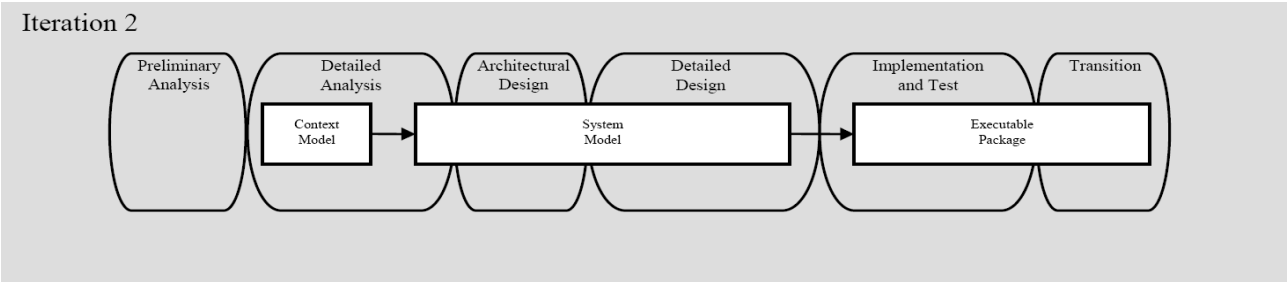
[Ramsin 2006]

Hybrid Design: Sample Iterations



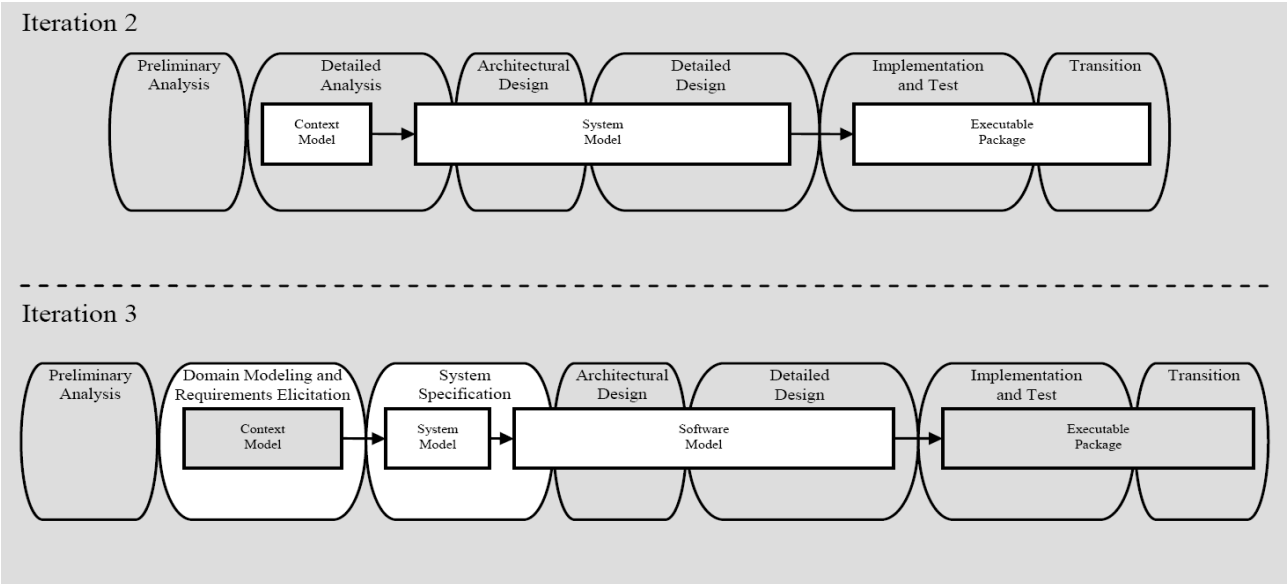
[Ramsin 2006]

Hybrid Design: Sample Iterations (Contd.)



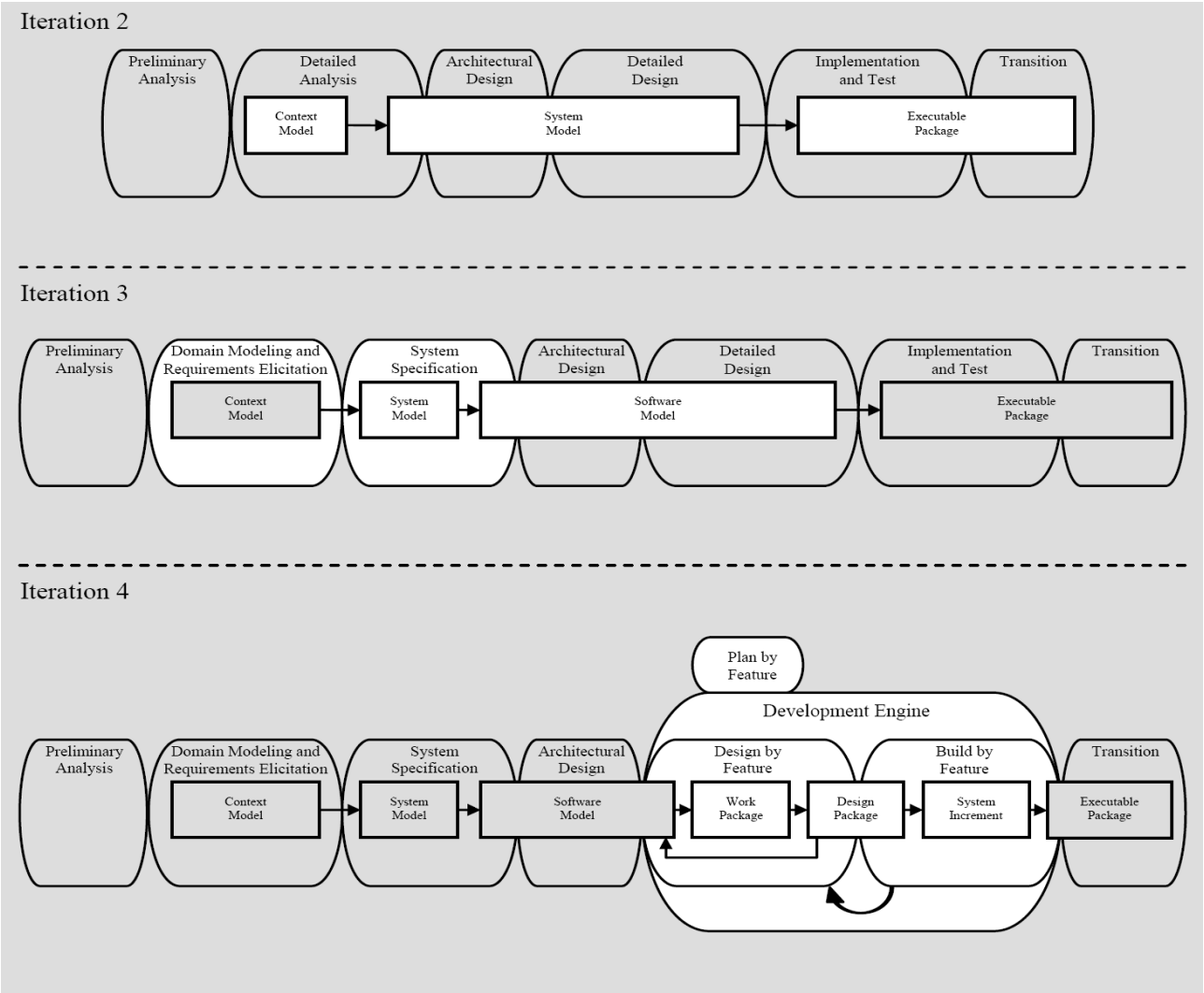
[Ramsin 2006]

Hybrid Design: Sample Iterations (Contd.)



[Ramsin 2006]

Hybrid Design: Sample Iterations (Contd.)



[Ramsin 2006]

References

- Henderson-Sellers, B, Ralyté, J, Ågerfalk, P.J., Rossi, M., *Situational Method Engineering*. Springer, 2014.
- Mirbel, I., Ralyté, J., Situational method engineering: combining assembly-based and roadmap-driven approaches. *Requirements Engineering 11,* 1 (March), 58–78, 2006.
- Ralyté, J., Deneckére, R., Rolland, C., Towards a generic model for situational method engineering. In *Proceedings of CAiSE 2003 (LNCS 2681)*, 95-110, 2003.
- Ralyté, J., Rolland, C., Deneckére, R., Towards a meta-tool for change-centric method-engineering: A typology of generic operators. In *Proceedings of CAiSE 2004 (LNCS 3084)*, 202-218, 2004.
- Ramsin, R., *The Engineering of an Object-Oriented Software Development Methodology*. Ph.D. Thesis, University of York, York, UK, 2006.