

Empirical study to identify coordination- and methodology patterns in large-scale agile development

Moritz Schuell, 15.04.2019, Bachelor's Thesis Kick-Off

Chair of Software Engineering for Business Information Systems (sebis)
Faculty of Informatics
Technische Universität München
www.matthes.in.tum.de

Outline

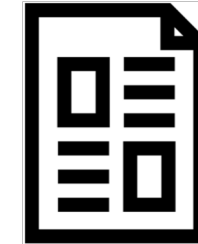
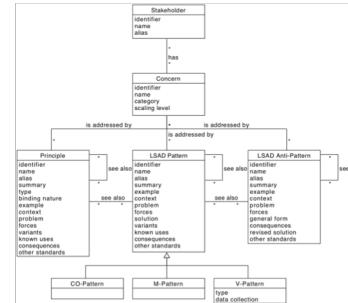


Motivation

Research Questions

Approach

Timeline



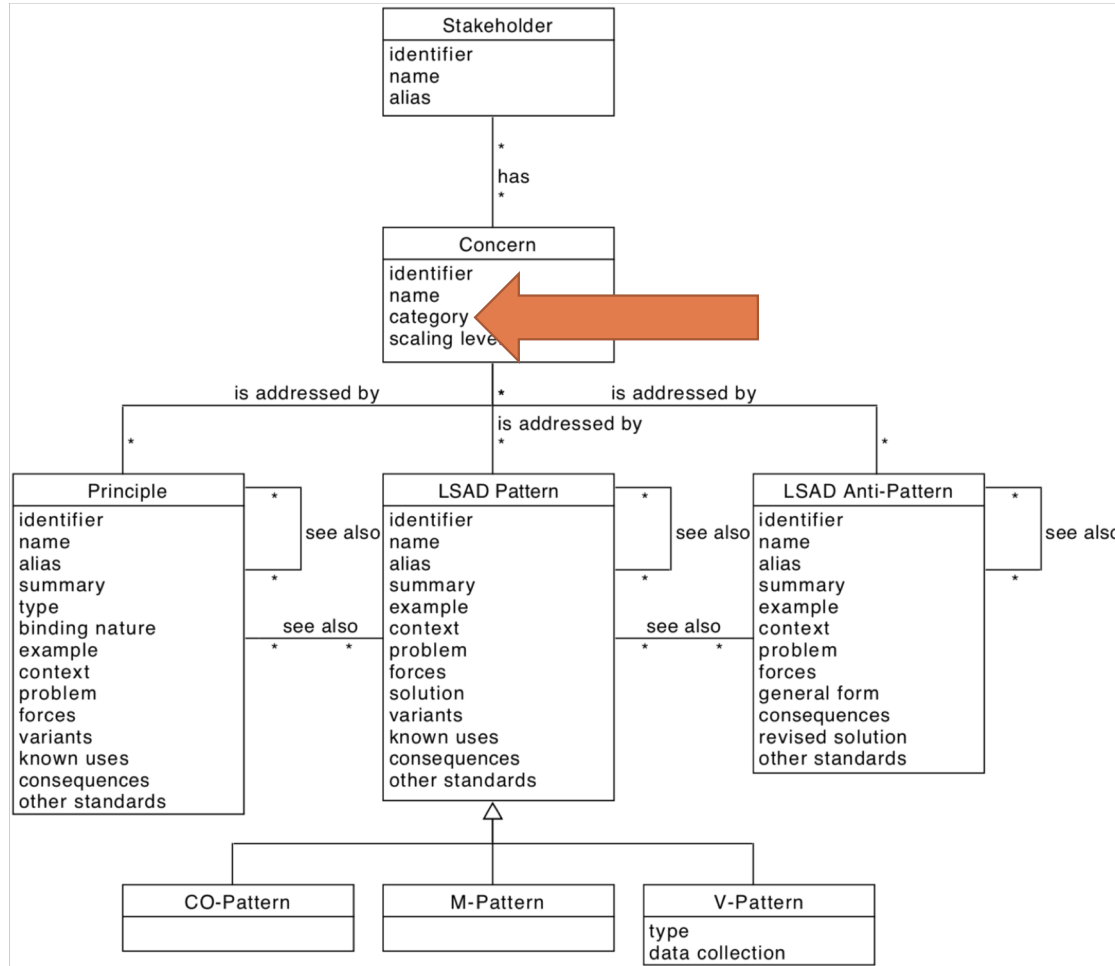
Agile methods are designed for small, co-located teams.

Their shown benefits make them attractive to **larger companies** as well [1], but applying them on large-scale projects leads to several **challenges** [2].

Large-scale agile development **pattern language** created by sebis [3].

It will contain patterns that address these concerns.

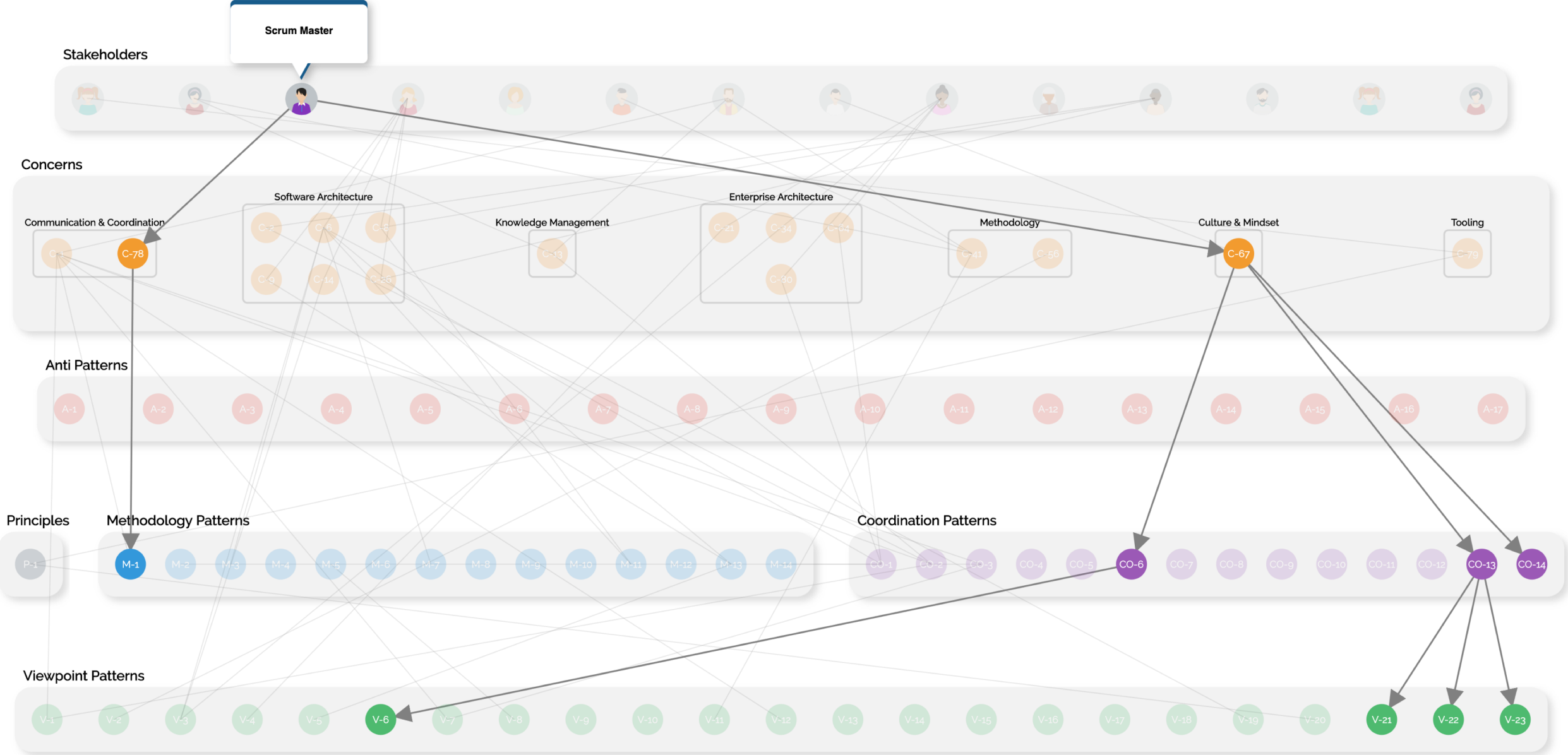
Pattern language has to be filled with **patterns** found in practice, that are used to address specific concerns.



Pattern language has to be filled with **patterns** found in practice, that are used to address specific concerns.

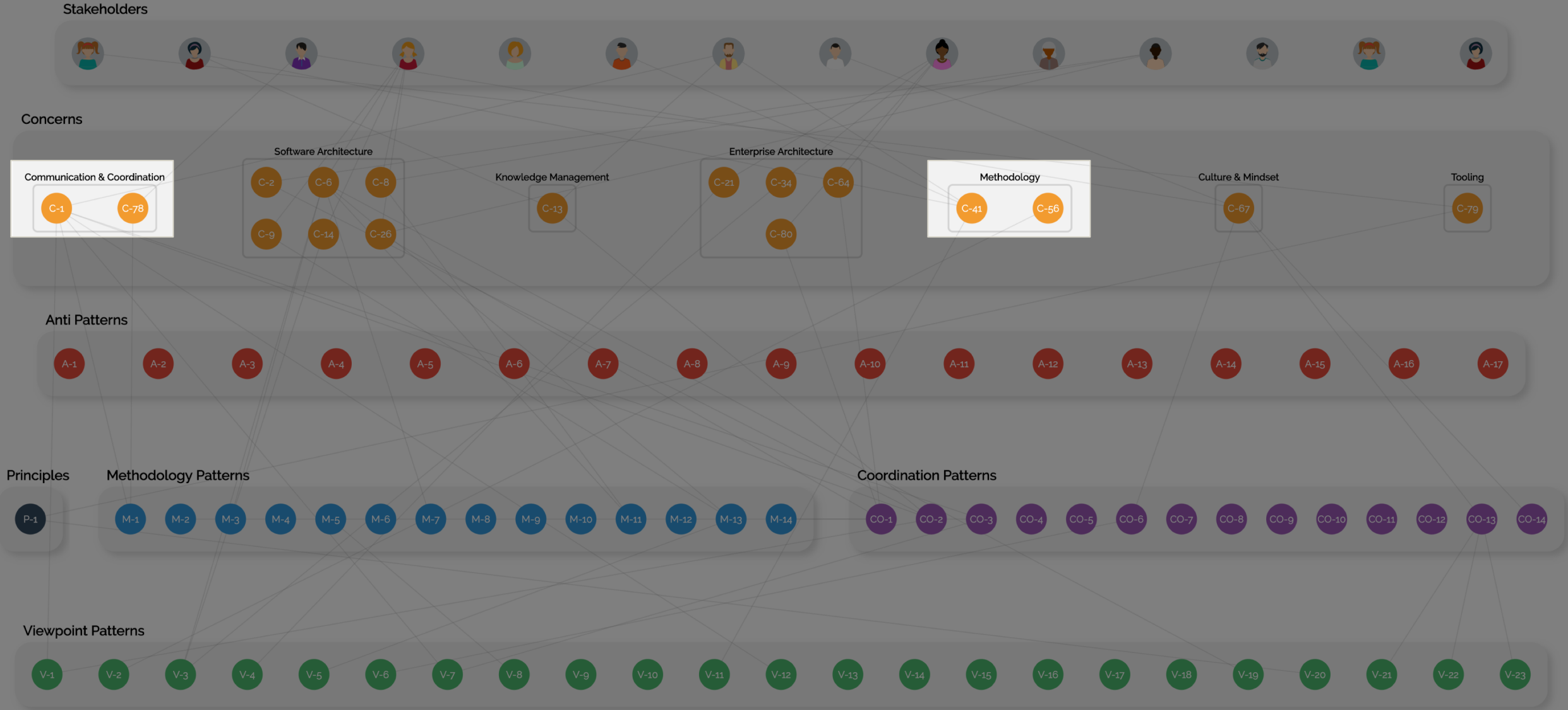
Focus for me is on patterns that address recurring **coordination (CO)** and **methodology (M)** concerns.

Large-Scale Agile Development Pattern Graph



For more information visit the [Scaling Agile Hub](#) developed by sebis

Large-Scale Agile Development Pattern Graph



For more information visit the [Scaling Agile Hub](#) developed by sebis

RQ 1

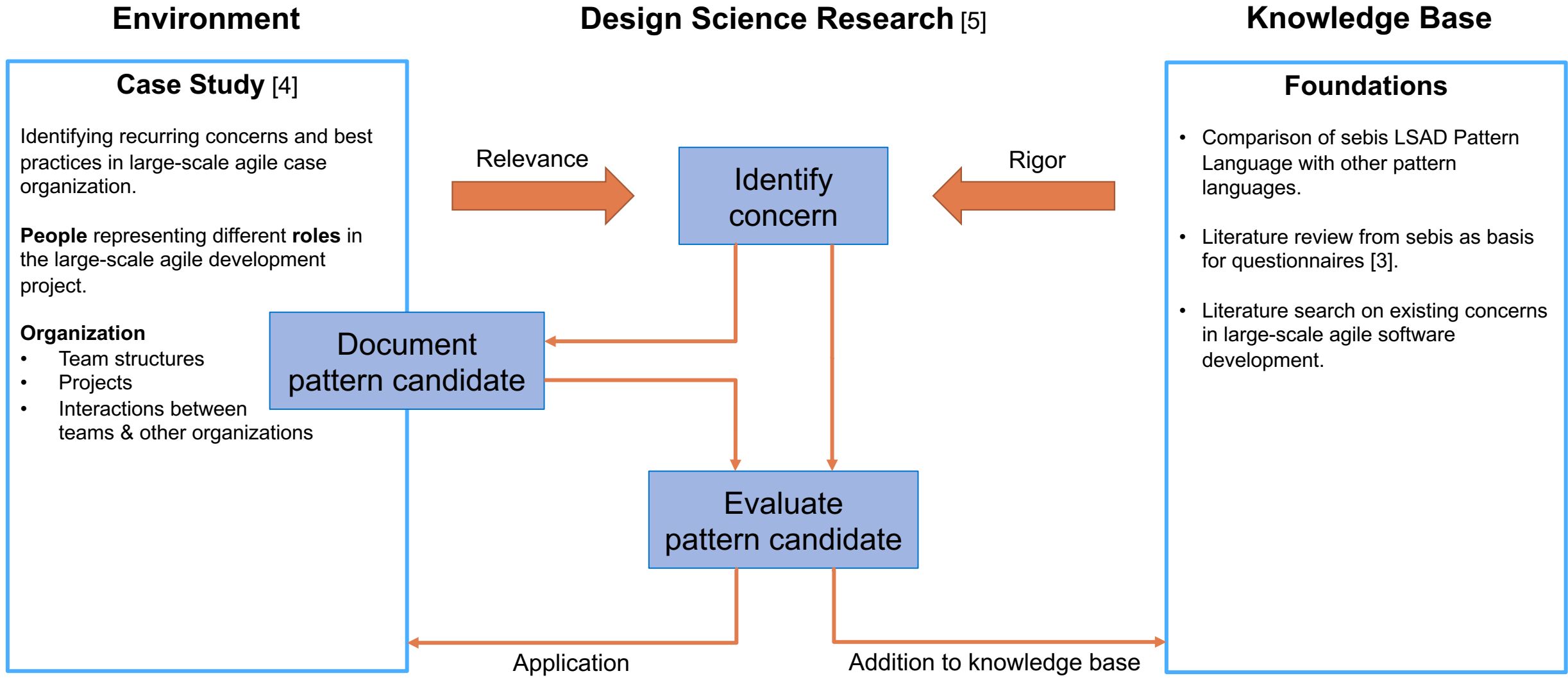
What are recurring coordination and methodology concerns in large-scale agile development?

RQ 2

What are good practices for addressing recurring coordination and methodology concerns in large-scale agile development?

RQ 3

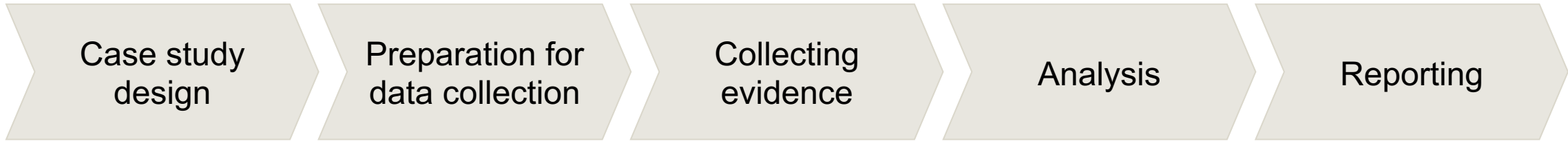
Which anti-patterns regarding coordination and methodologies should be avoided in large-scale agile development?



[3] Uludag, Harders & Matthes (2019)

[4] Runeson & Höst (2009)

[5] Hevner, March, Park & Ram (2004)



Questionnaire - Scaling agile practices

Information: Team-Name: _____ Location: _____
 Role: _____ Date: _____
 Interviewer: _____ Questionnaire-No.: 4.2

1. General questions about your role

1.1 Which role description applies to you?
 Head of department (IT) Head of department (Business) Division Manager (IT) Division Manager (Business)
 Business Architect Enterprise Architect Developer Product Owner
 Project manager (IT) Project manager (Business) Requirements Engineer Scrum Master
 Software Architect Solution Architect UX Designer

1.2 How many years have you been active in the field of scaled agile software development?
 1 - 2 years 3 - 5 years 6 - 10 years 11 - 15 years
 16 - 20 years > 20 years

1.3 How long has your company been using scaled agile software development?
 1 - 2 years 3 - 5 years 6 - 10 years 11 - 15 years
 16 - 20 years > 20 years

2. General questions about your team

2.1 How many members does your team have?
 1 - 5 6 - 10 11 - 15 More than 15

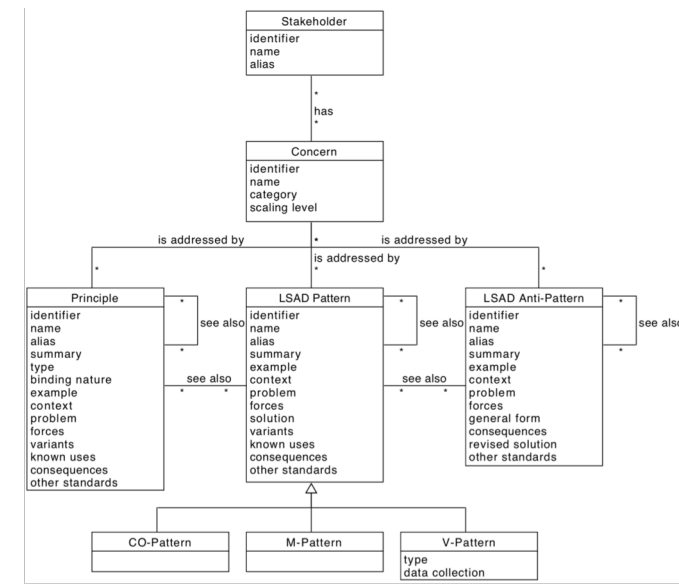
2.2 How many of them are developers?
 1 - 3 4 - 9 7 - 9 More than 9

2.3 What other roles are present in your agile team?
 Business Architect Enterprise Architect Developer Product Owner
 Project manager (IT) Project manager (Business) Requirements Engineer Scrum Master
 Software Architect Solution Architect UX Designer

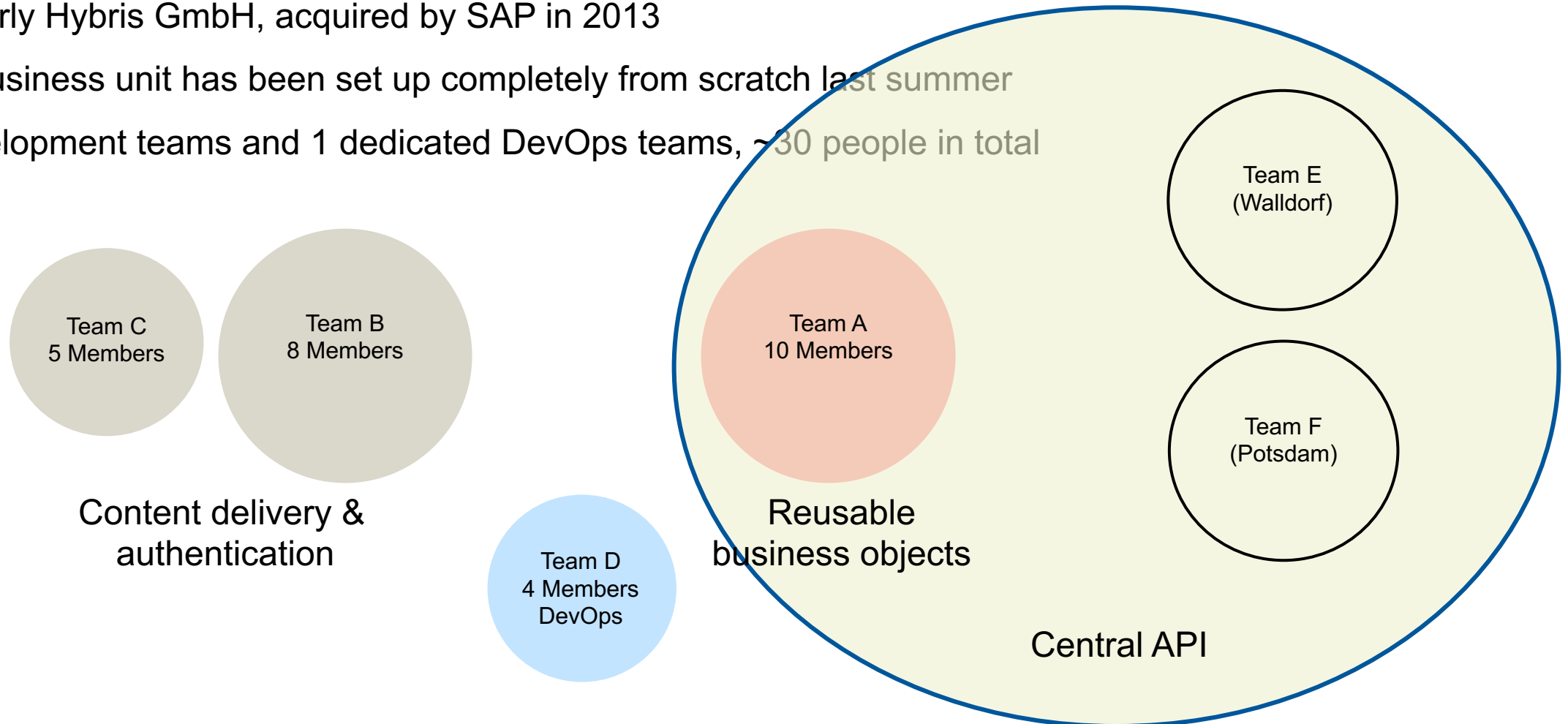
2.4 Is your team internationally or nationally distributed?
 International National

2.5 When did you start development of the current project / product?
 International National

2.6 When did your team start using agile development methodologies?
 International National



- SAP Cloud & Customer Experience Office in Munich
- Formerly Hybris GmbH, acquired by SAP in 2013
- The business unit has been set up completely from scratch last summer
- 3 development teams and 1 dedicated DevOps teams, ~30 people in total



What happened so far...

Source	Scope & goal	Focus on agile development	Number of patterns	Pattern categories	Pattern examples
[Coplien 1995]	Collection of patterns for shaping a new organization and its development processes	Partially	42	(1) Process patterns; (2) Organizational patterns	- CODE OWNERSHIP - GATEKEEPER - FIRE WALLS
[Harrison 1996]	Collection of patterns for creating effective software development teams	No	4	-	- UNITY OF PURPOSE - DIVERSITY OF MEMBERSHIP - LOCK 'EM UP TOGETHER
[Beedle et al. 1999]	Collection of Scrum patterns	Yes	3	-	- SPRINT - BACKLOG - SCRUM MEETINGS
[Taylor 2000]	Collection of patterns for creating product software development environments	No	9	(1) Establishing a Production Potential; (2) Maintaining a Production Potential; (3) Preserving a Production Potential	- DELIVERABLES TO GO - PULSES - BOOTSTRAPPING
[Coplien and Harrison 2004]	Collection of organizational patterns that are combined into a collection of four pattern languages	Yes	94	(1) Project Management; (2) Piecemeal Growth; (3) Organizational Style; (4) People and Code	- SKILL MIX - DEMO PREP - FEW ROLES
[Elssamadisy 2008]	Collection of patterns for successfully adopting agile practices	Yes	38	(1) Feedback Practices; (2) Technical Practices; (3) Supporting Practices; (4) The Clusters	- REFACTORING - CONTINUOUS INTEGRATION - SIMPLE DESIGN
[Beedle et al. 2010]	Collection of the most essential best practices of Scrum	Yes	11	-	- DAILY SCRUM - SPRINT BACKLOG - SPRINT REVIEW
[Välimäki 2011]	Enhancing performance of project management work through improved global software project management practice	Partially	18	(1) Directing a Project; (2) Starting up a Project; (3) Initiating a Project; (4) Controlling a Stage; (5) Managing Stage Boundaries; (6) Closing a Project; (7) Managing Product Delivery; (8) Planning	- COLLOCATED KICK-OFF - CHOOSE ROLES IN SITES - ITERATION PLANNING
[Mitchell 2016]	Collection of patterns to address agile transformation problems	Yes	54	(1) Patterns of Method; (2) Patterns of Responsibility; (3) Patterns of Representation; (4) Anti-Patterns	- LIMITED WIP - KANBAN SANDWICH - CONTROLLED FAILURE
[ScrumPloP 2019]	Body of pattern literature around agile and Scrum communities	Yes	234 (10)	(1) Value Stream; (2) Team; (3) Sprint; (4) Process Improvement; (5) Product Organization; (6) Distributed Scrum; (7) Scaling Scrum; (8) Scrum Core; (9) Misc	- SCRUM MASTER - SCRUM OF SCRUMS - PORTFOLIO STANDUP

Together with Nina we created a **tabular comparison** of the sebis LSAD pattern language with other pattern languages.

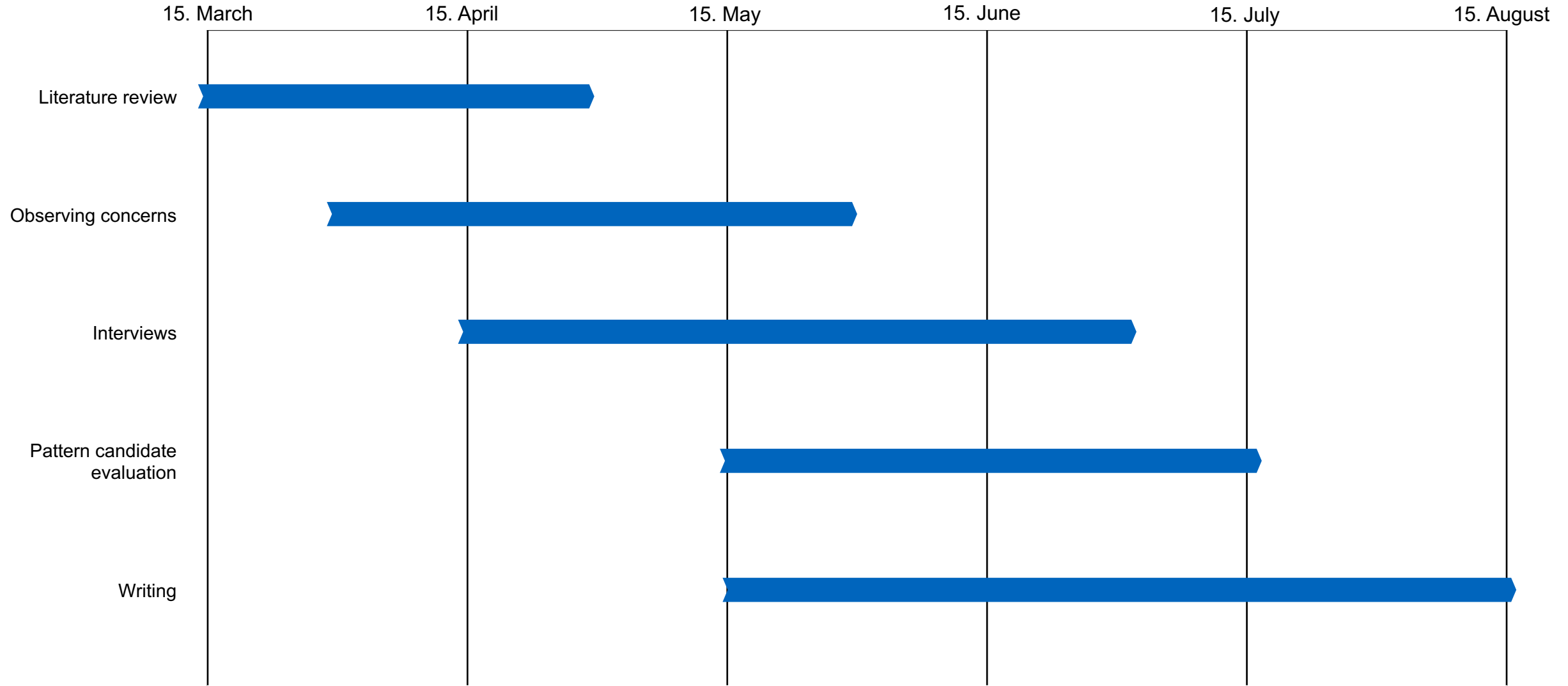
Most of the other pattern languages, while being focused on agile, **do not take scaling into account.**

The differentiation between **pattern types** is unique to the sebis language.

The first version of the interview questionnaires is finished.

The questionnaires are focused on personal **role and experiences** rather than information about the company.

Timeline



- [1] Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. *Journal of Systems and Software*, 119, 87-108.
- [2] Uludag, Ö., Kleehaus, M., Caprano, C., & Matthes, F. (2018). Identifying and Structuring Challenges in Large-Scale Agile Development based on a Structured Literature Review.
- [3] Uludag, Ö., Harders, N.-M., & Matthes, F. (2019). Documenting Recurring Concerns and Patterns in Large-Scale Agile Development.
- [4] Runeson, P., & Höst, M. (2009). Guidelines for conducting and reporting case study research in software engineering. *Empirical software engineering*, 14(2), 131.
- [5] Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS quarterly*, 28(1), 75-105.



Moritz Schüll

Technische Universität München
Faculty of Informatics
Chair of Software Engineering for Business
Information Systems

Boltzmannstraße 3
85748 Garching bei München

Tel +49.89.289.17132
Fax +49.89.289.17136

moritz.schuell@tum.de
www.matthes.in.tum.de

