DEMAND Community Event

Research on Quality Aspects of AI-based systems

Merel Veracx

Which quality aspects are crucial for the development of IT systems with an AI component.

An AI-based system is..

- .. a software solution that integrates algorithms that use data to learn rules or reactions automatically

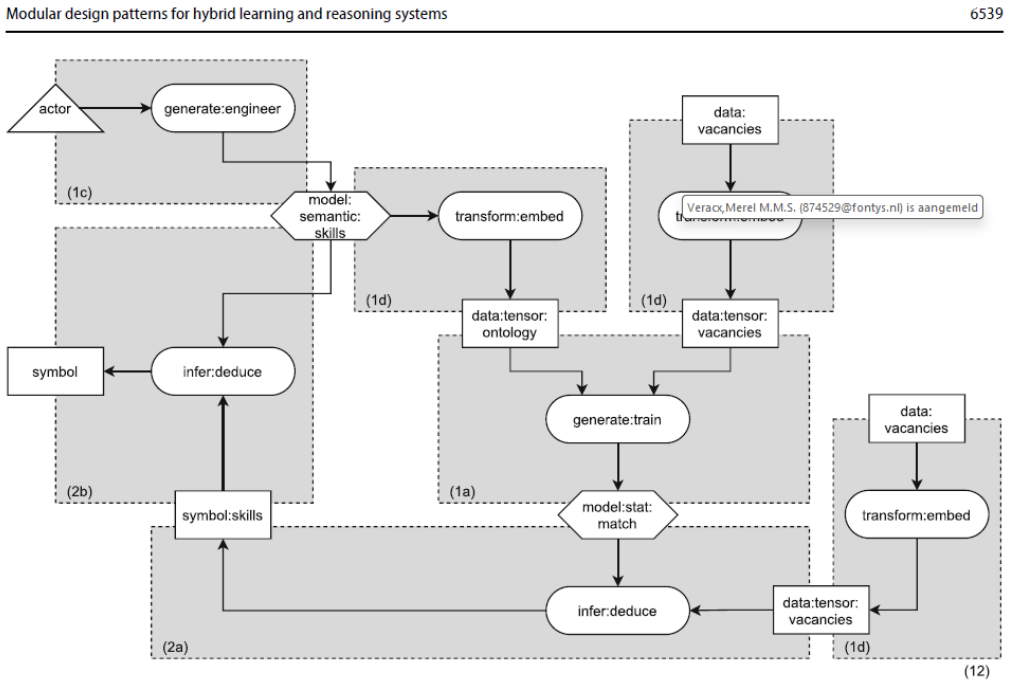
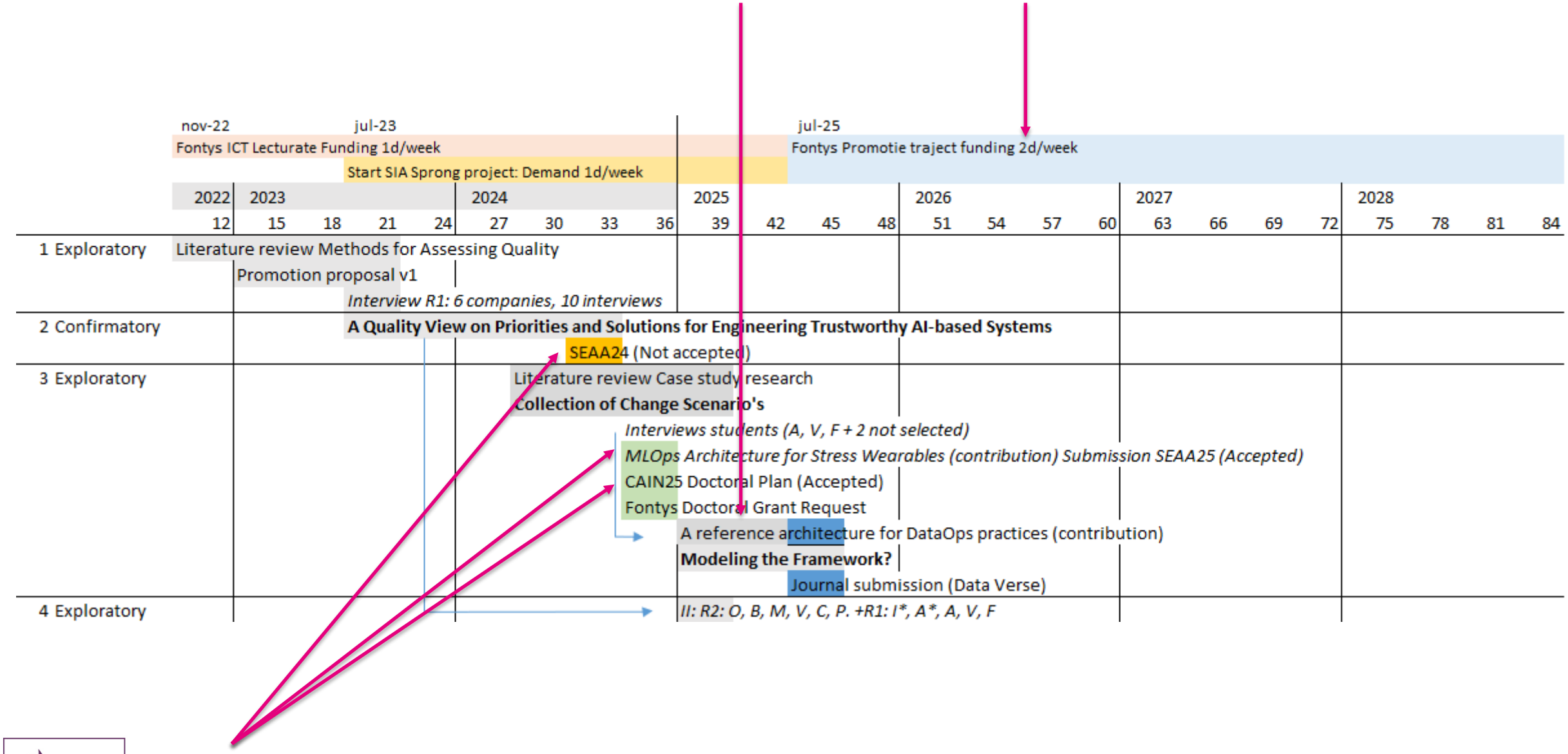


Fig. 12 Use-case for skills matching


nov-22
jul-23
jul-25
Fontys ICT Lecturate Funding 1d/week
Fontys Promotie traject funding 2d/week
Start SIA Sprong project: Demand 1d/week
2022
2023
2024
2025
2026
2027
2028
12 15 18 21 24 27 30 33 36 39 42 45 48 51 54 57 60 63 66 69 72 75 78 81 84
1 Exploratory
Literature review Methods for Assessing Quality
Promotion proposal v1
Interview R1: 6 companies, 10 interviews
2 Confirmatory
A Quality View on Priorities and Solutions for Engineering Trustworthy AI-based Systems
SEAA24 (Not accepted)
3 Exploratory
Literature review Case study research
Collection of Change Scenario's
Interviews students (A, V, F + 2 not selected)
MLOps Architecture for Stress Wearables (contribution) Submission SEAA25 (Accepted)
CAIN25 Doctoral Plan (Accepted)
Fontys Doctoral Grant Request
A reference architecture for DataOps practices (contribution)
Modeling the Framework?
Journal submission (Data Verse)
4 Exploratory
II: R2: O, B, M, V, C, P. +R1: I*, A*, A, V, F

Overview

- Lessons learned from student projects and company interviews
- Most important quality: *embrace change*
- New change scenario's because of AI themes:
  - data, model and software integration
- Sneak preview: support modifiability with 3 architectural perspectives
- Research methodology:
  - iterative, empirical examination of software process improvement
  - across 5 tiers of behavior: individual, team, project, organization and business environment

Interview Study 02-2023 – 03-2024

- What are the challenges in integrating ML in your software system(s)?
- Which are your top priorities in SE and ML system qualities?

- 5 interviewees from 4 companies
  - Technical expertise: AI/IT architects, manager software, data scientist and AI engineer
  - All organisations based in the Netherlands
- 18 different projects mentioned

Interview analysis results

TABLE III
OVERVIEW OF ORGANISATIONS AND PROJECTS

Org. ID - Interviewee	Project ID	Description	Status	Quotes
P - AI Architect	PA	Visual Fault Detection in Product Manufacturing with Anomaly Detection	P	39
	PT	Realisation of a distributed, automated ML training platform	P	11
	PS	Self-service In Training Visual Fault Detection	I	5

		Challenges For Quality Requirements	Literature Ref.	Organis. Ref.
Business Understanding	B1	Creating awareness off, and understanding the feasibility of the application possibilities of AI for business challenges or accelerating processes and financing this innovation.	[14,16,18]	C,M,P,V
	B2	Improving collaboration across business and the various technical disciplines such as AI, data and software engineering	[13,14,16]	C,V
	B3	*Freedom from risk, fairness, human autonomy*: evaluation and awareness of the impact of automating a decision using AI	[13,14,16,20]	M,P
	B4	*Functional suitability and model correctness*: recording of the requirements where the acceptance	[13,14,15,16]	P

Interview analysis conclusion:
3 AI themes and focus on quality modifiability

- AI systems can only bloom successfully with **a solid data foundation** and stemming from **flexible software with renewed focus on the functional suitability.**
- We propose to update the AI engineering definition to also explicitly include data-engineering: AI Engineering is a combination of machine learning, **data engineering** and software engineering with the goal to build production-ready machine learning systems.

- Fact:
  - most of the cost of developing software systems occurs after their initial release, thus challenging software engineers to embrace change.
- The continual change that software undergoes during its lifetime is generally called evolution, and the degree to which it is easy or hard to change existing software is often called **modifiability**.
  - Other quality criteria are closely related: modularity, reusability, analyseability, testability, flexibility, adaptability, scalability, installability, and replaceability.
  - We use the broader term modifiability to refer to all these aspects of evolving IT systems.

An AI-based system is..

- .. a software solution that integrates algorithms that use data to learn rules or reactions automatically

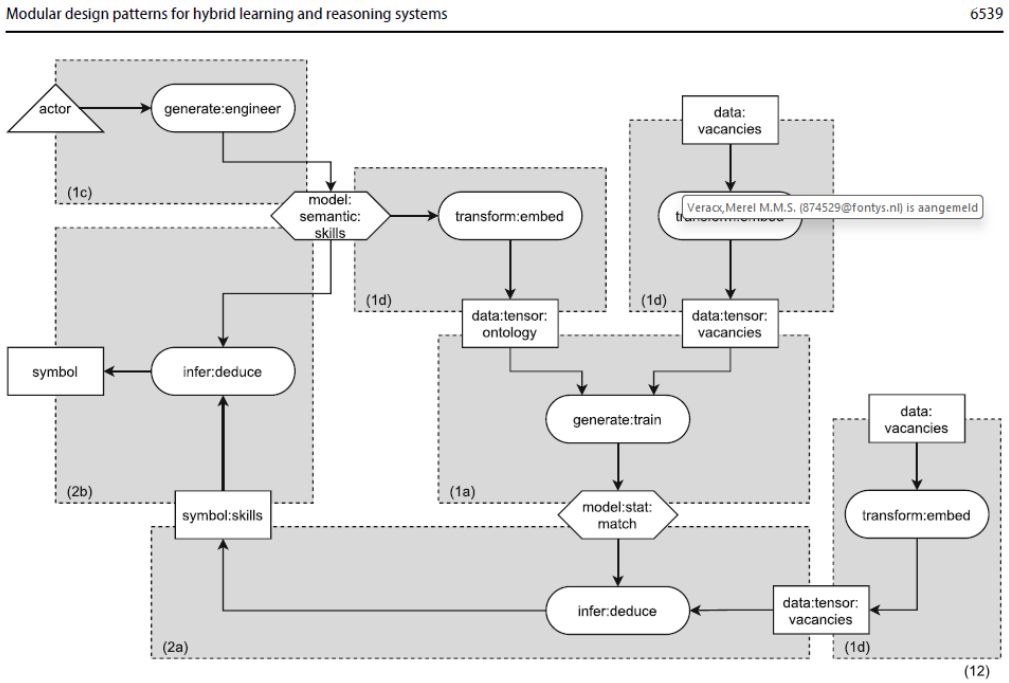


Fig. 12 Use-case for skills matching

These systems introduce new practices because..

- .. they depend on frequently changing data, models and new integrations into existing software

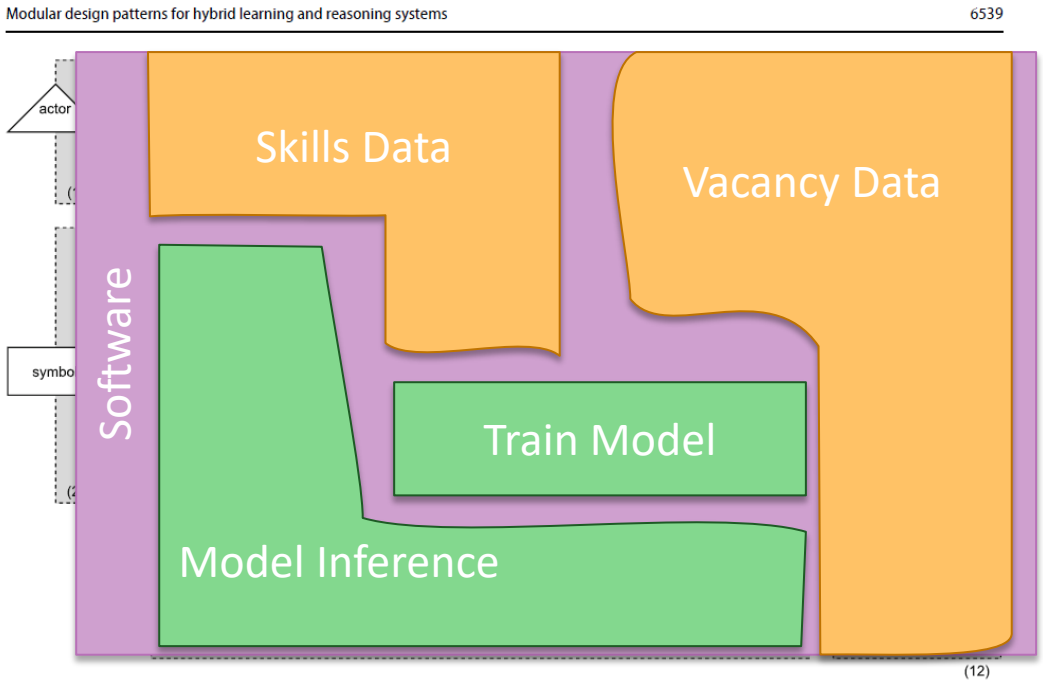


Fig. 12 Use-case for skills matching

- our 3 AI themes

The AI-based system changes because..

- .. these new AI themes

New context practices

New “human in the loop” practices

New feedback practices

New deployment practices

Democratization of AI engineering techniques

New data

New models

New (AI) components

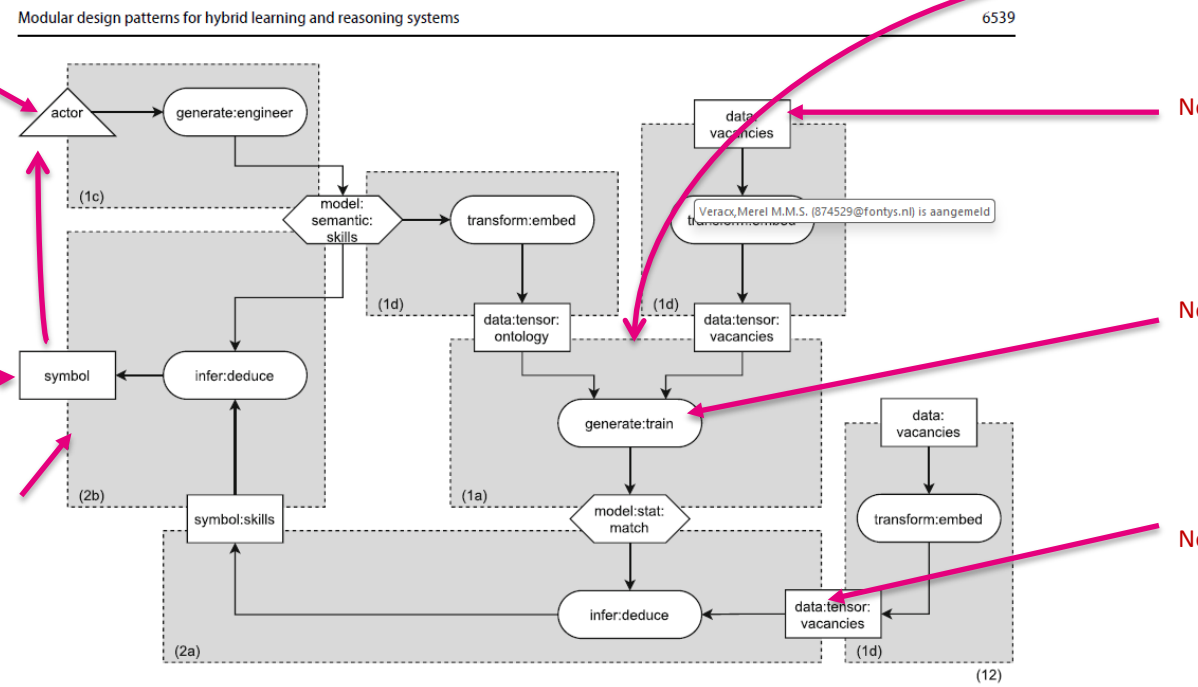
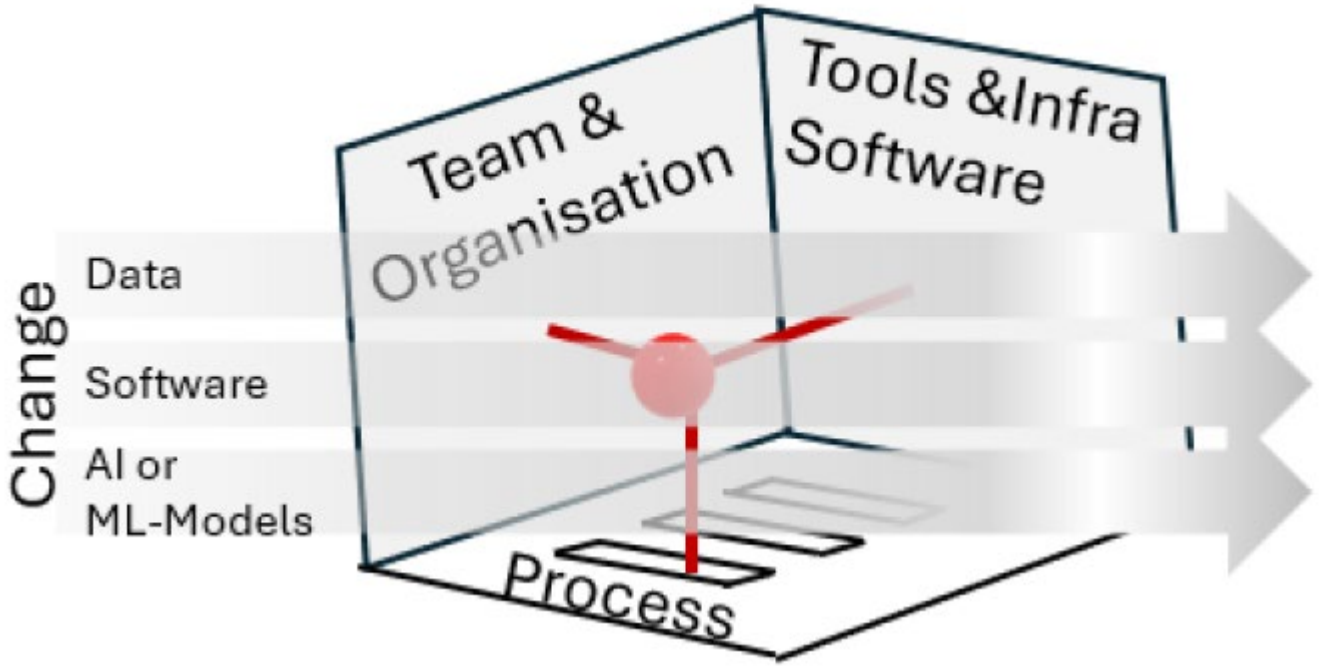


Fig. 12 Use-case for skills matching

Main research question:

How to support the evolution of AI-based systems with a holistic framework linking change scenarios to architectural perspectives and AI themes?



Next step:

The creation of a change catalogue for AI-enabled systems

What is a recent change?

What was the source?

What is the event that motivates to change the IT architecture?

4.2.2 Scenario 2: Software Error Isolation and Correction

This example scenario describes maintenance to the Flight Management System to correct a software error. The system clock cannot be set correctly on leap days (February 29).

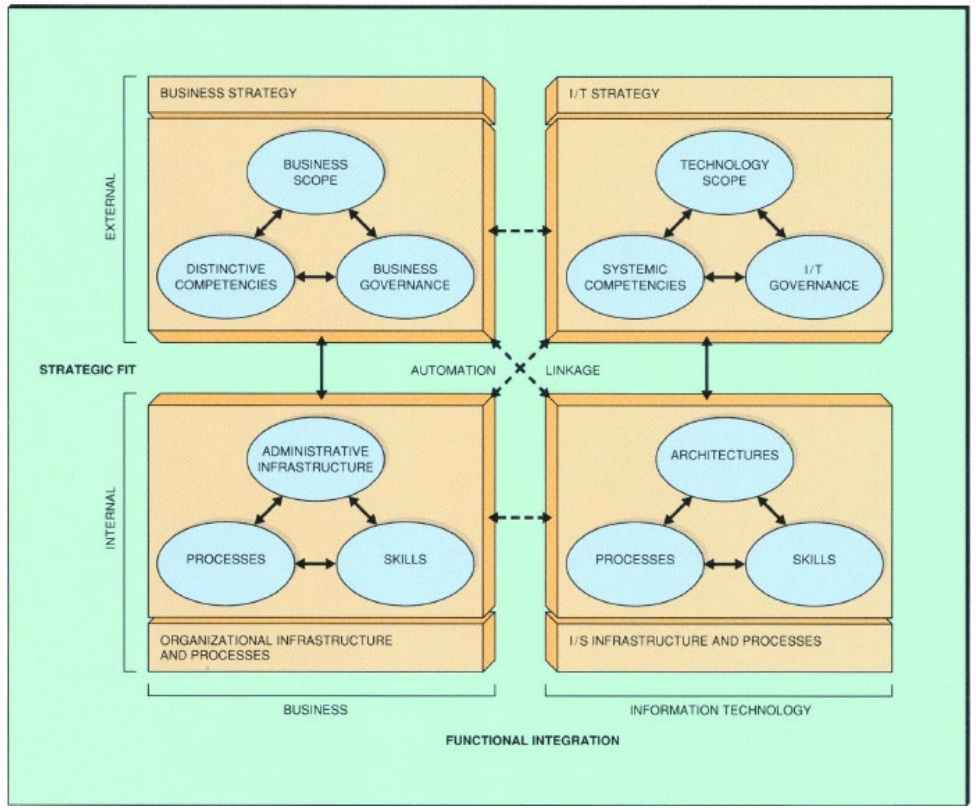
Scenario Part	Value
Source	Navigation System Program Manager
Stimulus	Direction to correct the system clock leap-day software error
Artifact	System Initialization package
Environment	CECOM Software Engineering Center (SEC) – It has been maintaining this system for 3 years. [organic, not original developer]
Response	Isolate the error. Modify the code and add automated tests. Test the Flight Management System using automated tests. Deliver the Flight Management System software to integration testing.
Response Measure	The error is isolated and affected the artifacts identified within 14 days. [calendar time] The affected code modules are all located in the System Initialization package. [affected artifacts] The SEC delivers the navigation system to integration testing within 30 days. [calendar time]

© 1994 – Kazman "Software architecture" Modifiability Quality Scenario

How did the IT architecture change? (Artifact, Environment, Response, Response Measure)

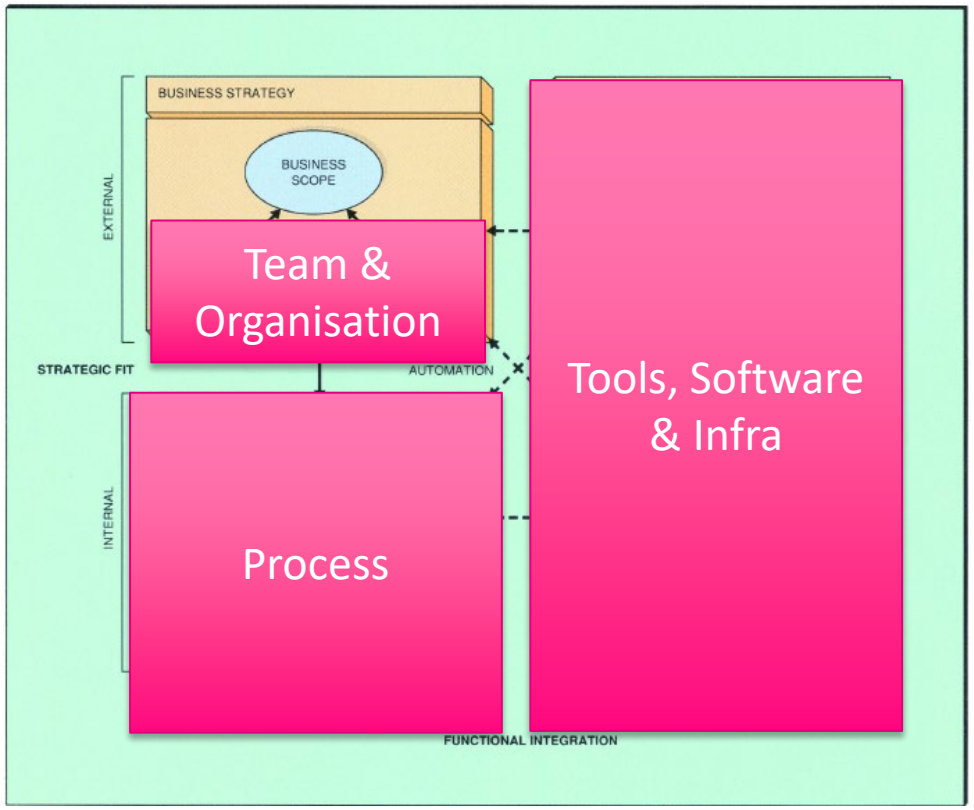
What is the feature needed by the trigger?	What was the current state of the IT architecture when this feature was identified?	What was changed in the IT architecture to address the feature needed?

How to link change scenario's to which architectural perspectives?



© 1992 -- John Henderson, N. Venkatraman – Strategic Alignment Model

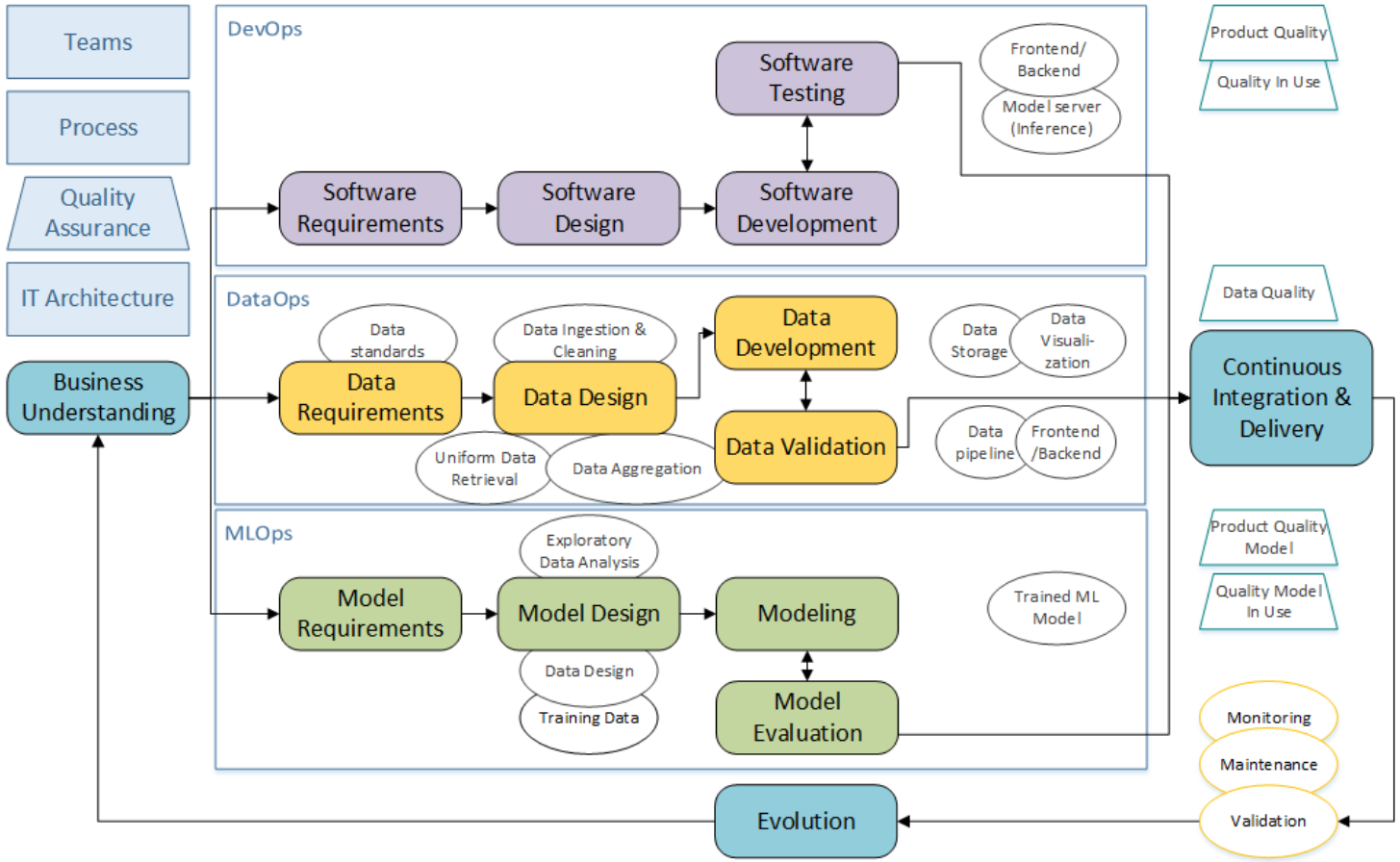
How to link change scenario's to which architectural perspectives?



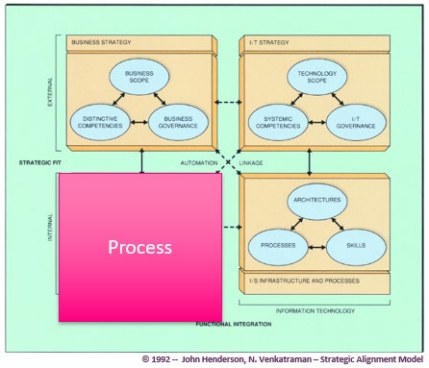
© 1992 -- John Henderson, N. Venkatraman – Strategic Alignment Model

Work in progress: sneak preview

Architectural perspective: process

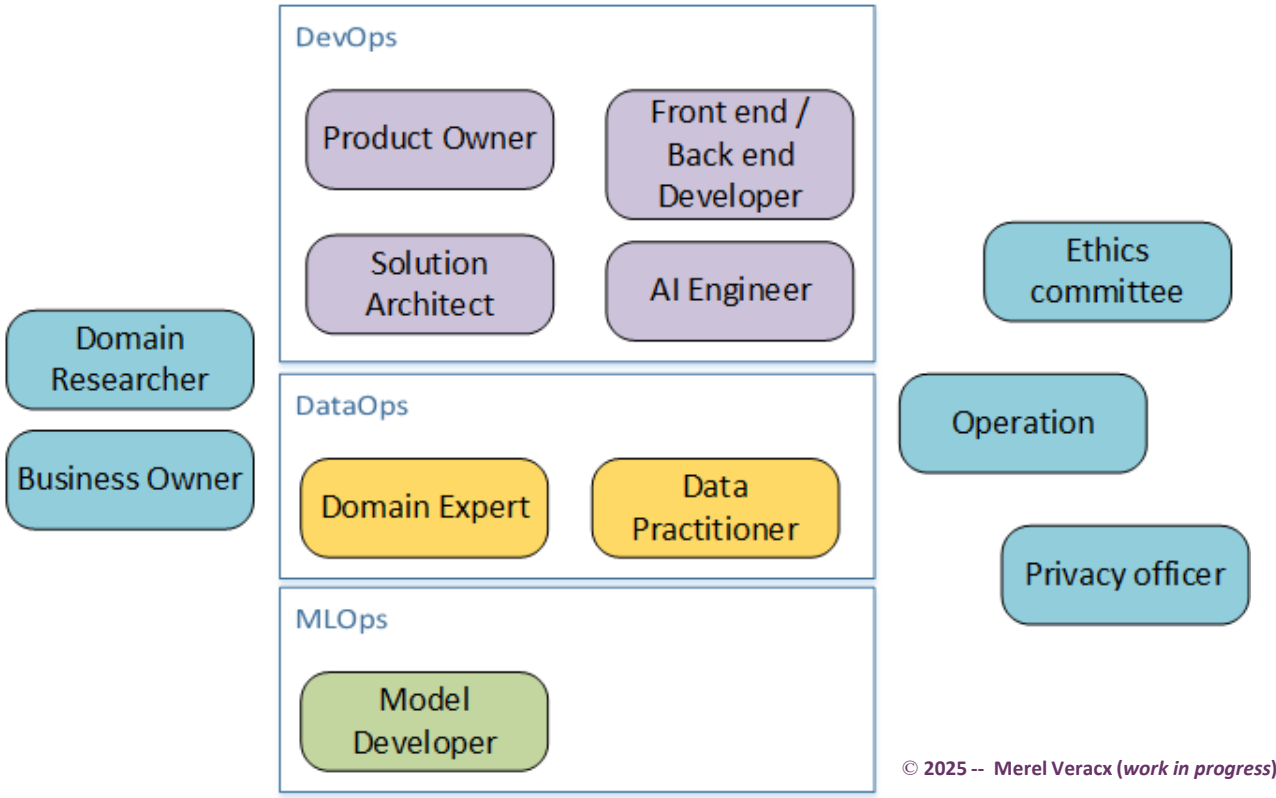


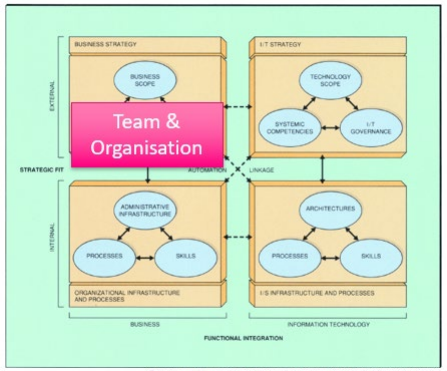
© 2025 -- Merel Veracx, Petra Heck



© 1992 -- John Henderson, N. Venkatraman – Strategic Alignment Model

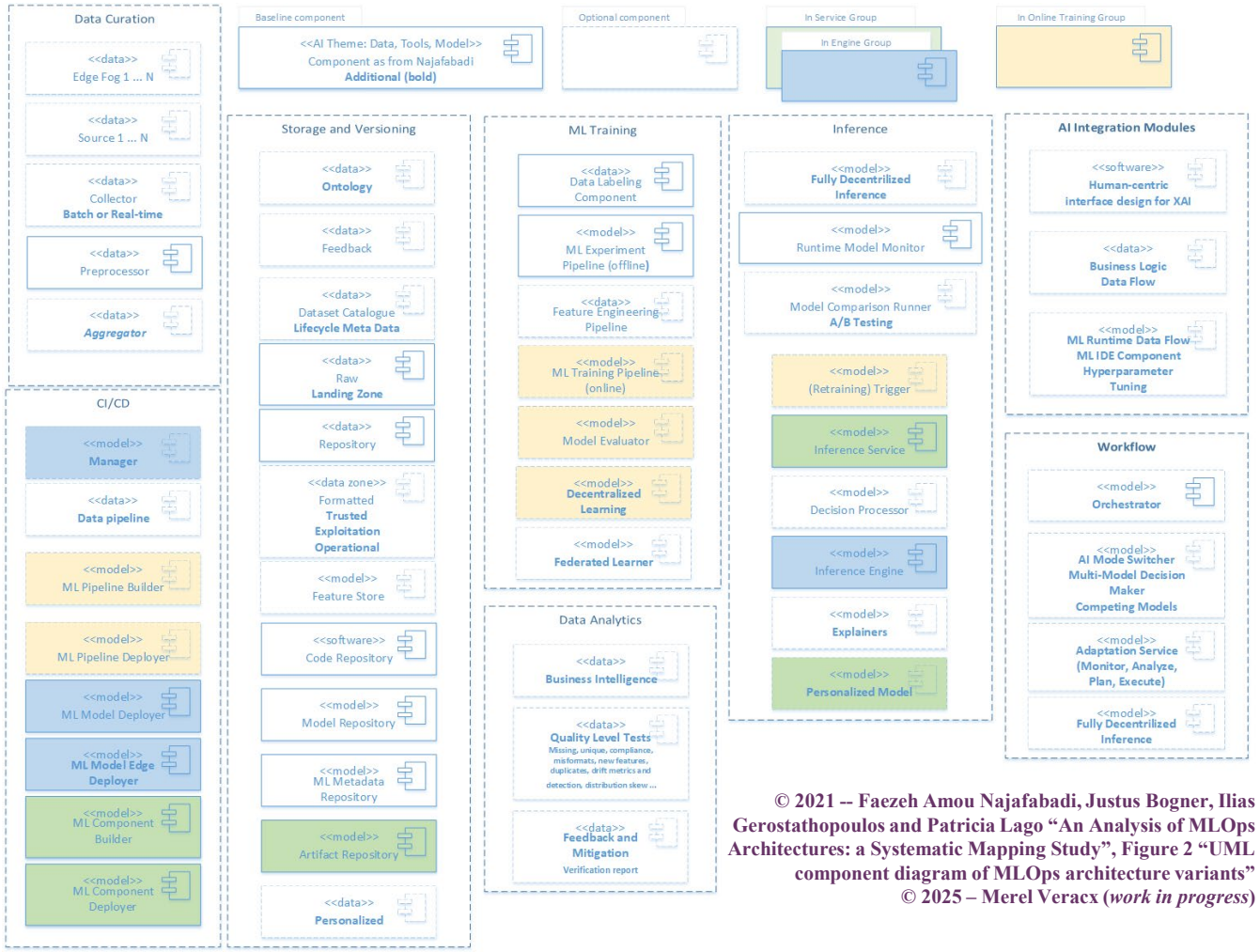
Work in progress: sneak preview
Architectural perspective: team & organisation



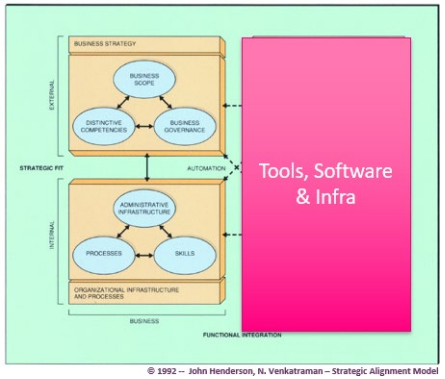


© 1992 -- John Henderson, N. Venkatraman – Strategic Alignment Model

Work in progress: sneak preview
Architectural perspective: tools, software and infra



© 2021 -- Faezeh Amou Najafabadi, Justus Bogner, Ilias Gerostathopoulos and Patricia Lago "An Analysis of MLOps Architectures: a Systematic Mapping Study", Figure 2 "UML component diagram of MLOps architecture variants"
© 2025 – Merel Veracx (*work in progress*)



Literature research

inventory

Taxonomy of changes

Change Scenario's

Teams and Organization			Tools, Infra and Software			Processes		
Data	SE	Model	Data	SE	Model	Data	SE	Model
	Impact			*Impact*				*Impact*
		Impact				*Impact*		

validation

Junior Software Engineers

Portfolio, interviews, intervisies

validation

Exploratory

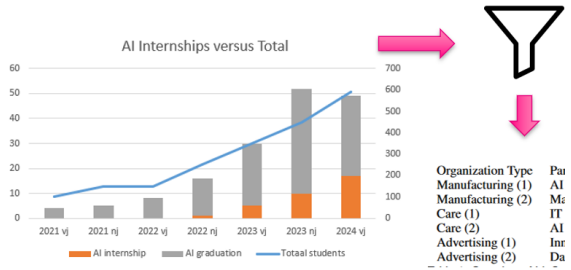
Explanatory

Confirmatory

(Runeson, 2012)

FONTYS ICT InnovationLab

AI Engineers @ Industry



Best-practices for Software Engineers

Business process → Best-practices → Maturity → Automatisation → Modifiability

Data Engineering, AI Engineering Software Engineering

→ IT Architecture

Design Decisions

Design Patterns

result

Dutch High-Tech Region

Interviews, surveys

Fontys UNIVERSITY OF APPLIED SCIENCES