Zorro: op naar nul downtime

Sturen op Veiligheid: Break-out sessie 2025.11.06

Thomas Nägele (TNO)

Vadim Zaytsev (UT)

Annemarie Jutte (Saxion)

This publication is part of the project ZORRO with project number KICH1.ST02.21.003 of the research programme Key Enabling Technologies (KIC) which is (partly) financed by the Dutch Research Council (NWO). This study is supported by the TNO-ESI joint innovation center part of the Netherlands Organization for Applied Scientific Research TNO.

Zero downtime in cyber physical systems

Powered by industry and academia

Zorro: introduction

Society greatly depends on high-tech systems

- Enormous economic dependencies: see Nexperia in the past weeks
- Safety: healthcare, transportation, defense, …

High-tech systems become more complex by the day …

… while there are few skilled people around to maintain them.

There is a need for techniques to reduce downtime of complex systems.

Zorro: consortium

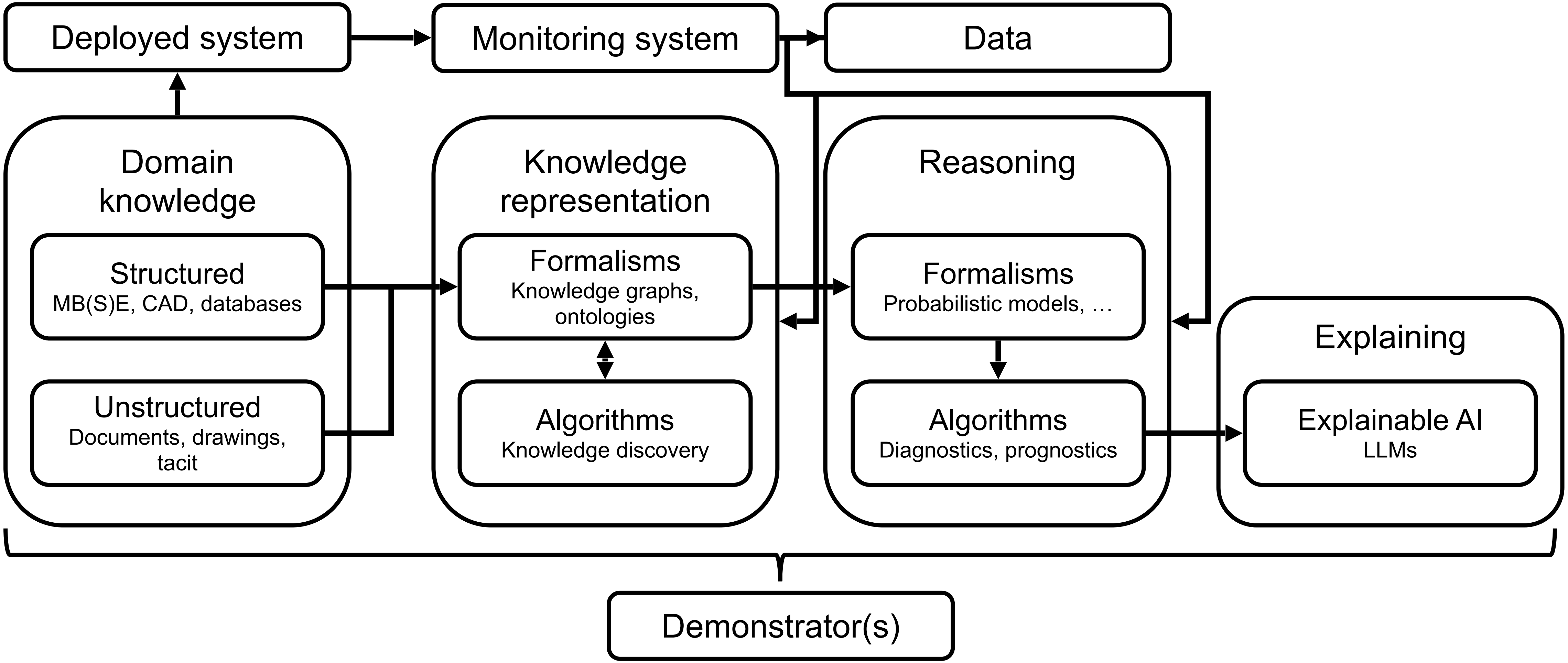
Action research: experiment in the field as early as possible, bringing ideas into practice.

Academic partners

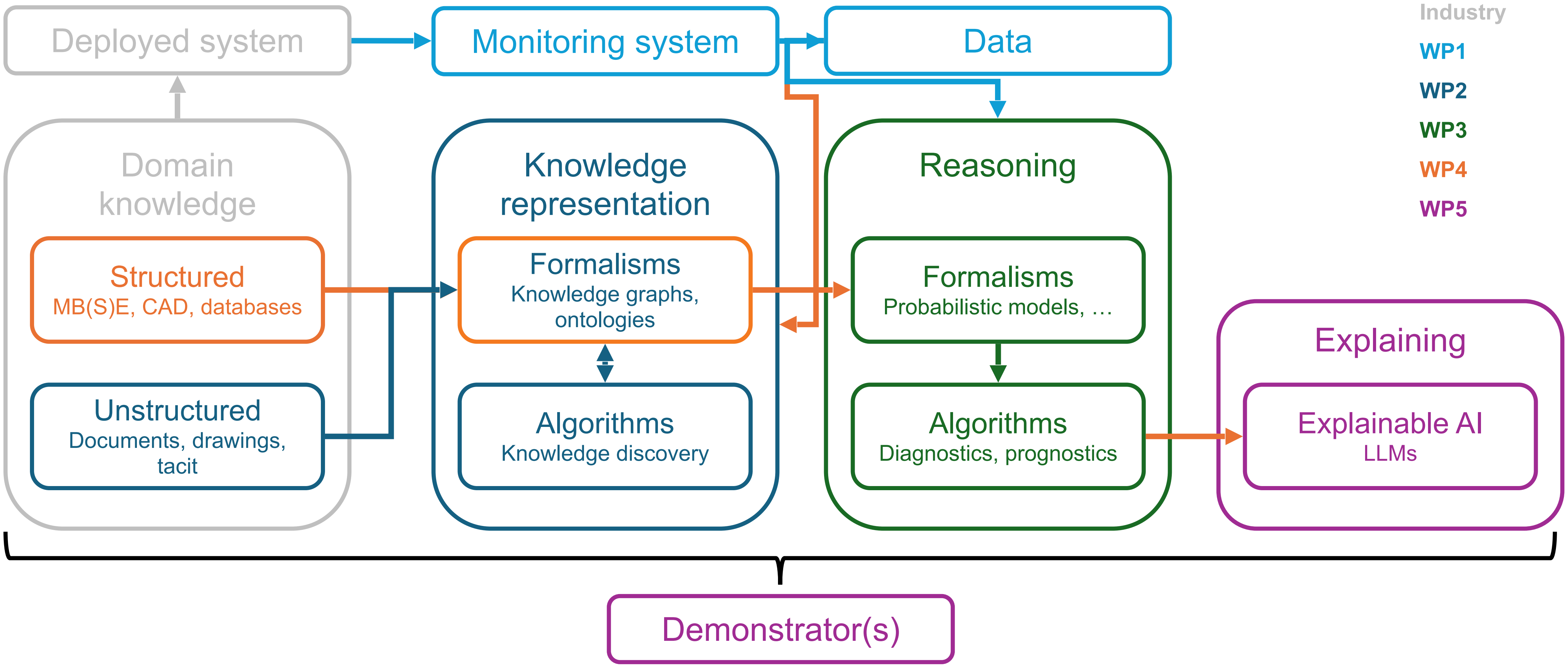
Knowledge institutes & funding body

Industry partners

Zorro project overview



Zorro project overview



Reasoning under uncertainty

High-tech system's diagnostics is about explaining observations made on the system, like at the doctor

What is the root cause, given what I now observe?

Every step in the diagnostic process should reduce the uncertainty about the root cause of an issue

Conduct tests to acquire additional information about your system

Continue until the root cause is identified

At TNO-ESI, we use probabilistic reasoning approaches for this

Great for dealing with uncertainty …

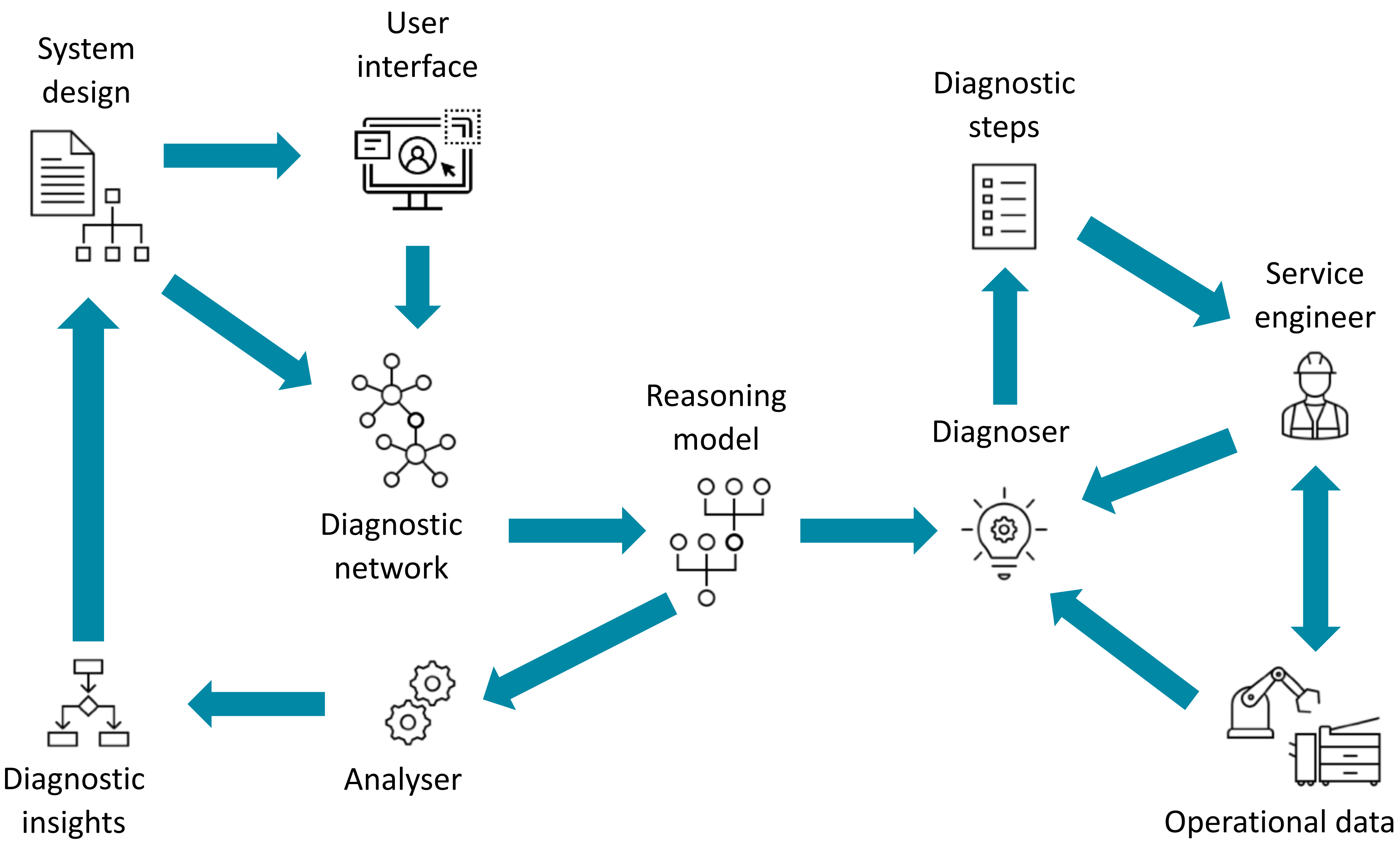
… difficult to create

How to come to a probabilistic model?

What information to use?

How to do this at scale?

A methodology for high-tech diagnostics



Reasoning about safety is not very different

- **Health assessment of national infrastructure**
- **Assessing mission readiness of military equipment**
- **Social safety due to societal disturbances**

All of these require you to reason with uncertainty!

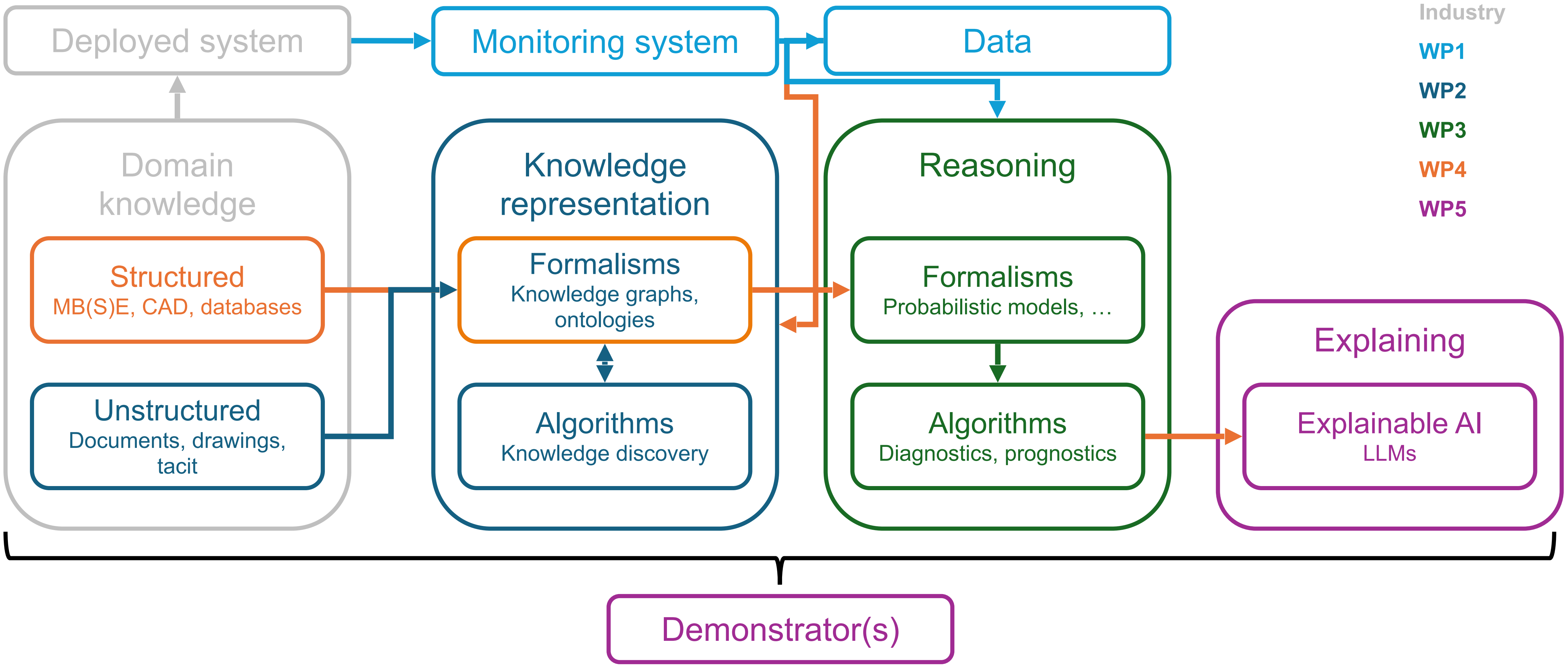
We're always open to discuss how to strengthen each other

Next up:

Model-based approaches and domain-specific languages: *why and how to formalize knowledge?*

Explainable AI: *how can we trust what the models tell us?*

Zorro overzicht



waarom ⇒ hoe

Moderne systemen zijn complex

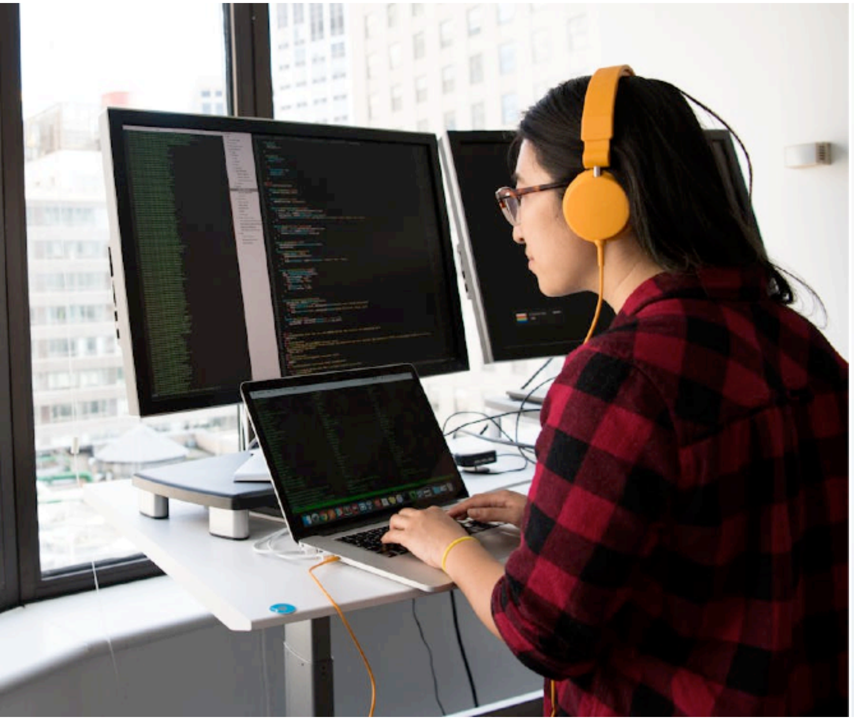
- programmatuur
- sensoren
- natuurkunde
- gegevens
- mensen
- onzekerheid
- . . .

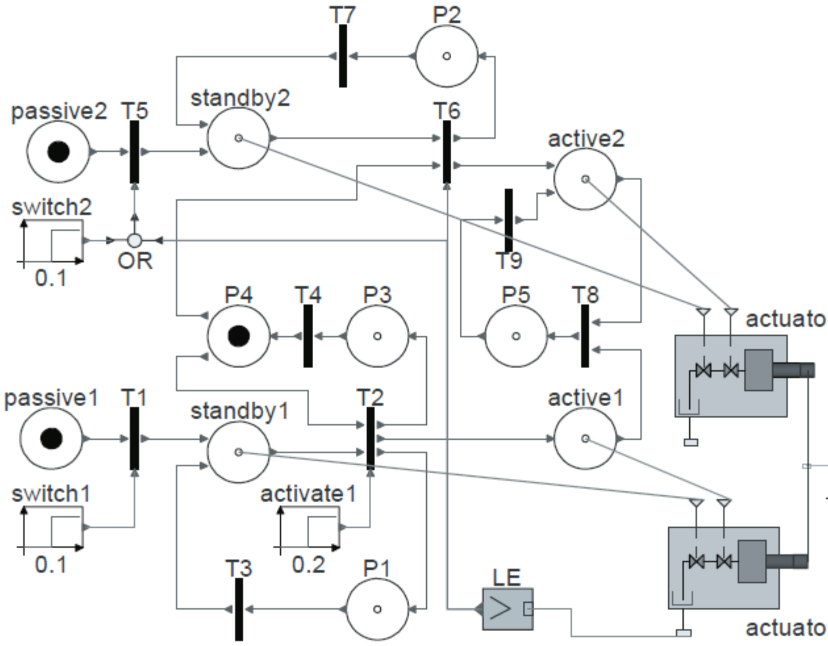
Zorro
Zero downtime in cyber physical systems

UNIVERSITY OF TWENTE. | FmT Formal Methods & Tools

Zorro
Zero downtime in cyber physical systems

Soorten modellen





mentale modellen

natuurlijke modellen

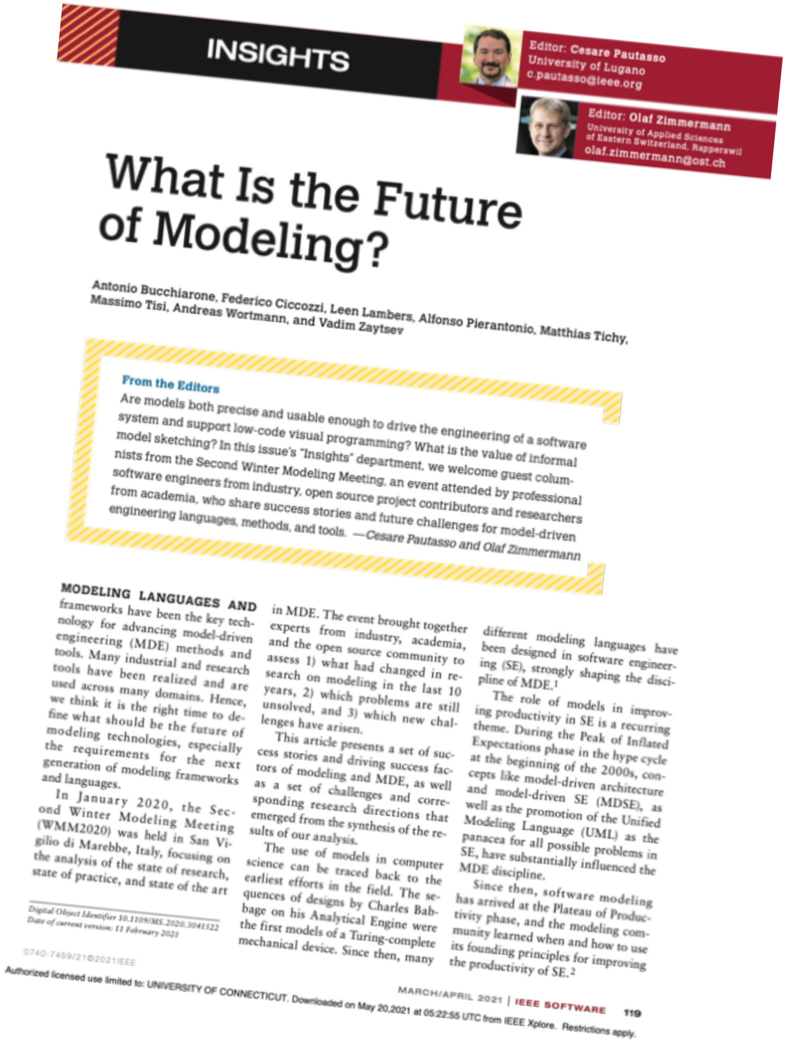
formele modellen

Elk vertrouwd systeem is gemodelleerd

UNIVERSITY OF TWENTE. | FMT Formal Methods & Tools

Wat is MBSE/MGST?

- modellen staan centraal
  - niet documenten, mensen, afspraken, code…
- past in alle fasen
  - vereisten
  - ontwerp
  - implementatie
  - onderhoud
  - testen



INSIGHTS

Editor: Cesare Pautasso
University of Lugano
c.pautasso@ieee.org

Editor: Olaf Zimmermann
University of Applied Sciences of Eastern Switzerland, Rapperswil
olaf.zimmermann@ost.ch

What Is the Future of Modeling?

Antonio Bucchiarone, Federico Ciccozzi, Leen Lambers, Alfonso Pierantonio, Matthias Tichy, Massimo Tisi, Andreas Wortmann, and Vadim Zaytsev

From the Editors

Are models both precise and usable enough to drive the engineering of a software system and support low-code visual programming? What is the value of informal model sketching? In this issue's "Insights" department, we welcome guest columnists from the Second Winter Modeling Meeting, an event attended by professional software engineers from industry, open source project contributors and researchers from academia, who share success stories and future challenges for model-driven engineering languages, methods, and tools. —*Cesare Pautasso and Olaf Zimmermann*

MODELING LANGUAGES AND frameworks have been the key technology for advancing model-driven engineering (MDE) methods and tools. Many industrial and research tools have been realized and are used across many domains. Hence, we think it is the right time to define what should be the future of modeling technologies, especially the requirements for the next generation of modeling frameworks and languages.

In January 2020, the Second Winter Modeling Meeting (WMM2020) was held in San Vigilio di Marebbe, Italy, focusing on the analysis of the state of research, state of practice, and state of the art in MDE. The event brought together experts from industry, academia, and the open source community to assess 1) what had changed in research on modeling in the last 10 years, 2) which problems are still unsolved, and 3) which new challenges have arisen.

This article presents a set of success stories and driving success factors of modeling and MDE, as well as a set of challenges and corresponding research directions that emerged from the synthesis of the results of our analysis.

The use of models in computer science can be traced back to the earliest efforts in the field. The sequences of designs by Charles Babbage on his Analytical Engine were the first models of a Turing-complete mechanical device. Since then, many different modeling languages have been designed in software engineering (SE), strongly shaping the discipline of MDE.[1]

The role of models in improving productivity in SE is a recurring theme. During the Peak of Inflated Expectations phase in the hype cycle at the beginning of the 2000s, concepts like model-driven architecture and model-driven SE (MDSE), as well as the promotion of the Unified Modeling Language (UML) as the panacea for all possible problems in SE, have substantially influenced the MDE discipline.

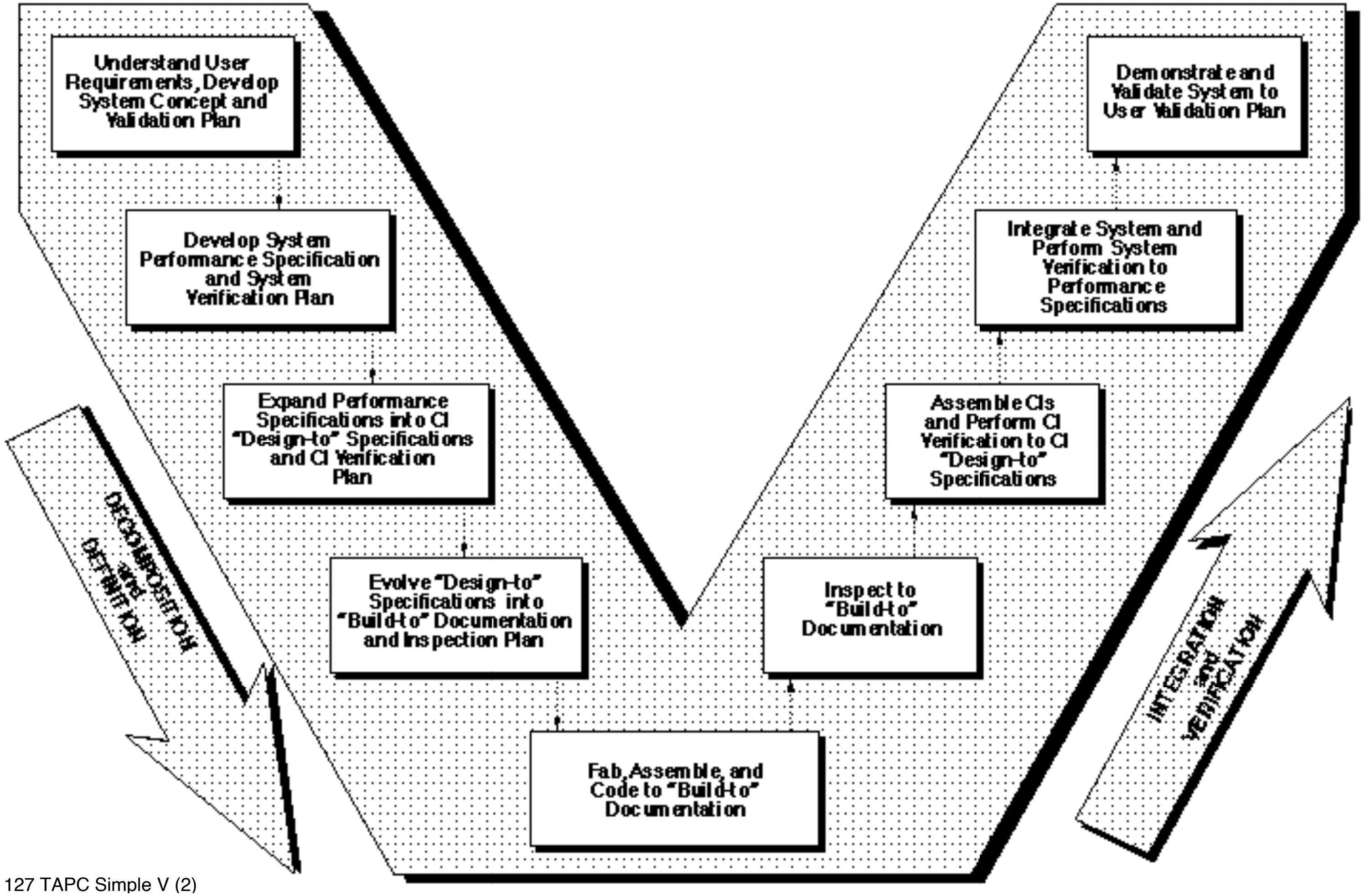
Since then, software modeling has arrived at the Plateau of Productivity phase, and the modeling community learned when and how to use its founding principles for improving the productivity of SE.[2]

Digital Object Identifier 10.1109/MS.2020.3041522
Date of current version: 11 February 2021

Zorro
Zero downtime in cyber physical systems

UNIVERSITY OF TWENTE. | FMT Formal Methods & Tools

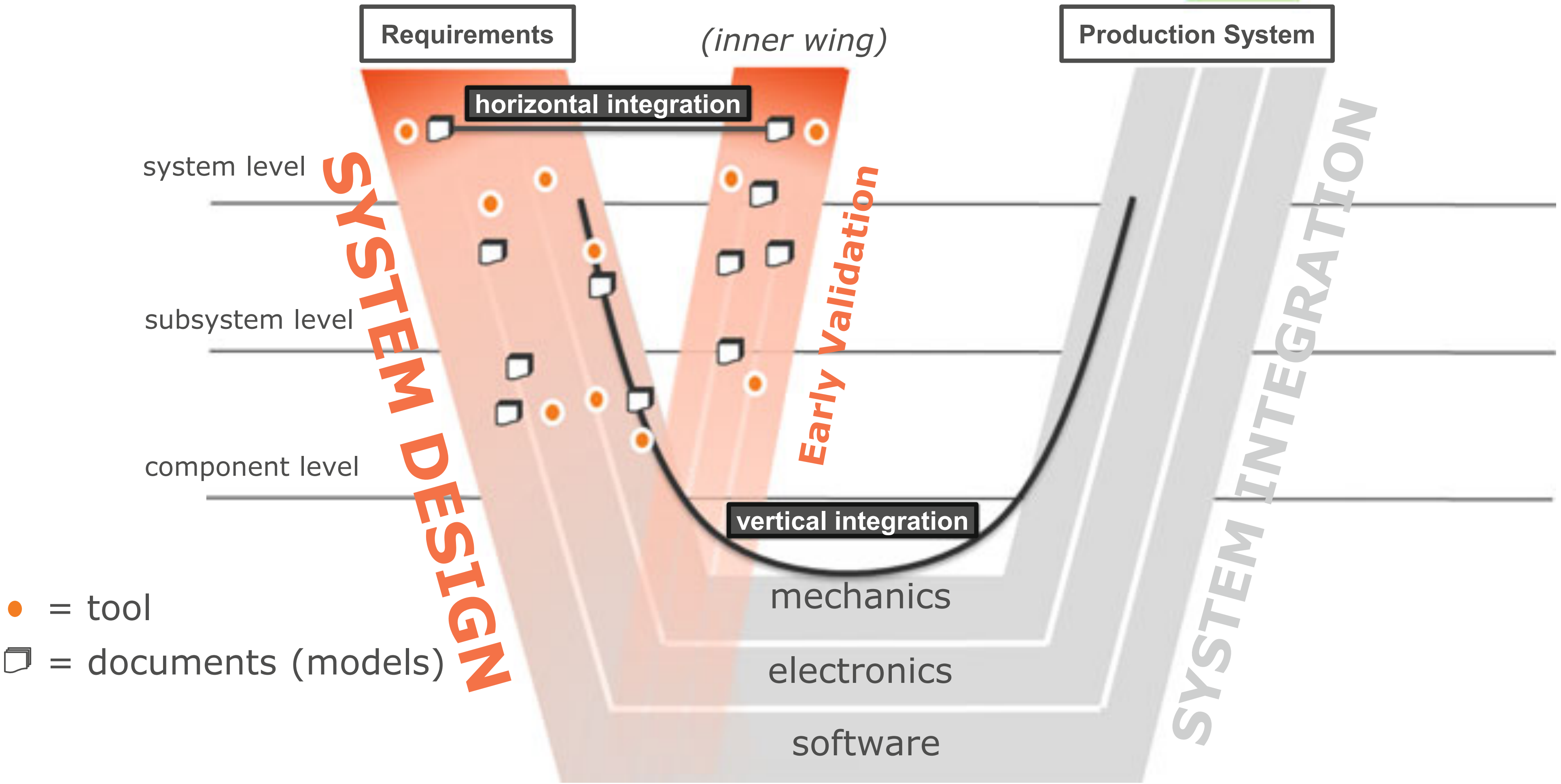
Modelgebaseerde systeemtechniek



127 TAPC Simple V (2)

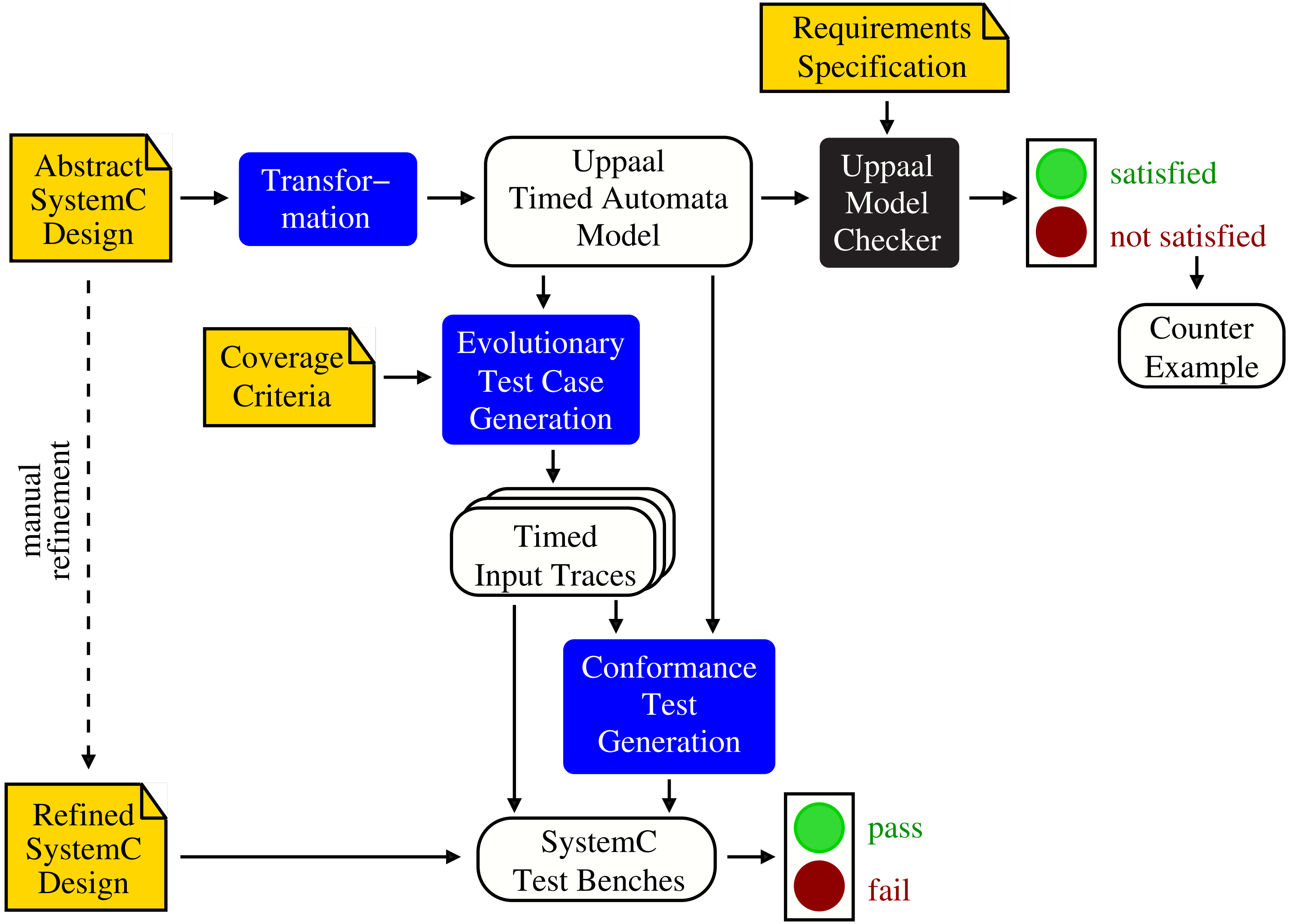
Forsberg, Mooz, The Relationship of System Engineering to the Project Cycle, 1991-1995

Modelgebaseerde systeemtechniek



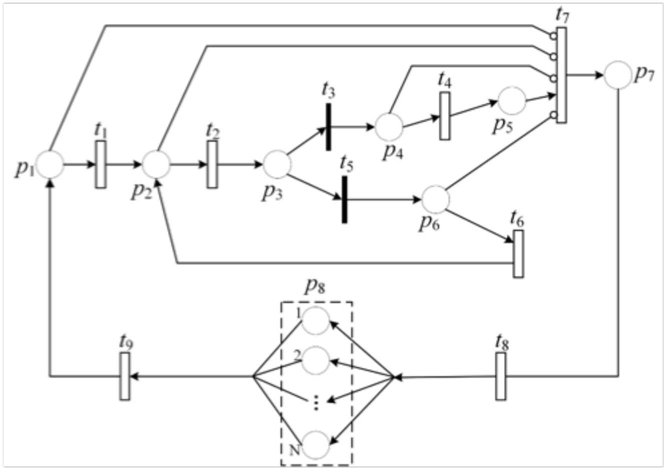
Berardinelli, Mazak, Alt, Wimmer, Kappel, DOI:10.1007/978-3-319-56345-9_11, 2017

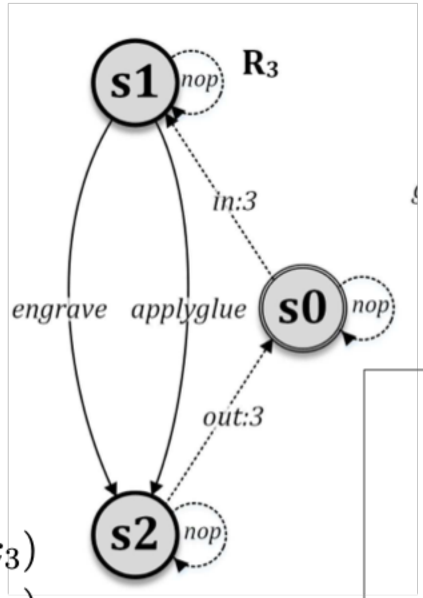
Modelgebaseerde systeemtechniek

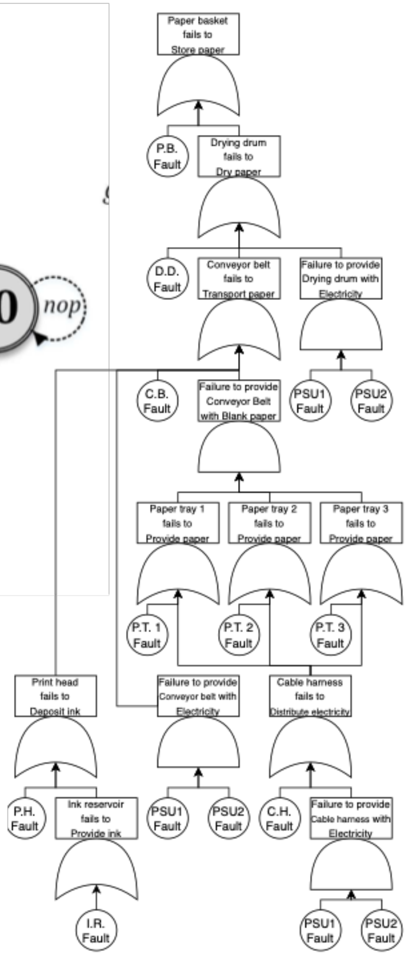


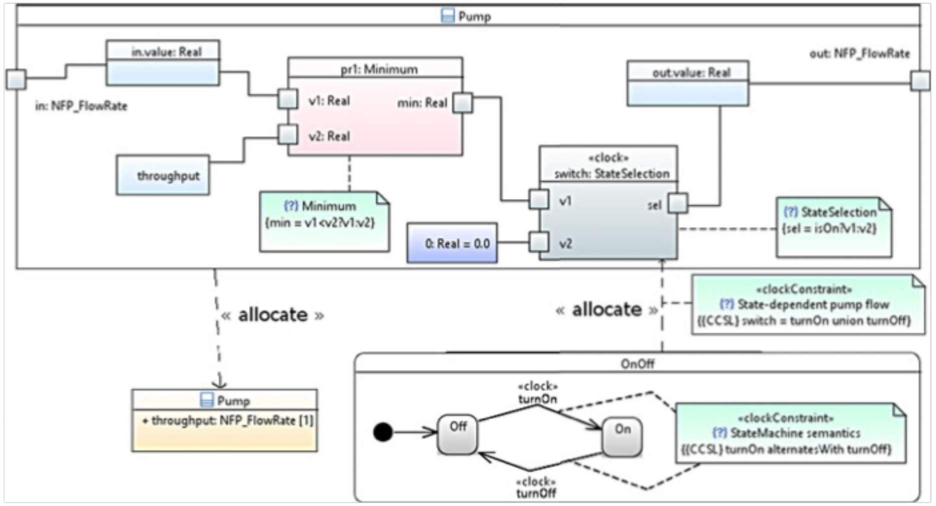
Herber, Glesner, DOI:10.1007/978-3-658-09994-7_1, 2015

Formele modellen voor CFS

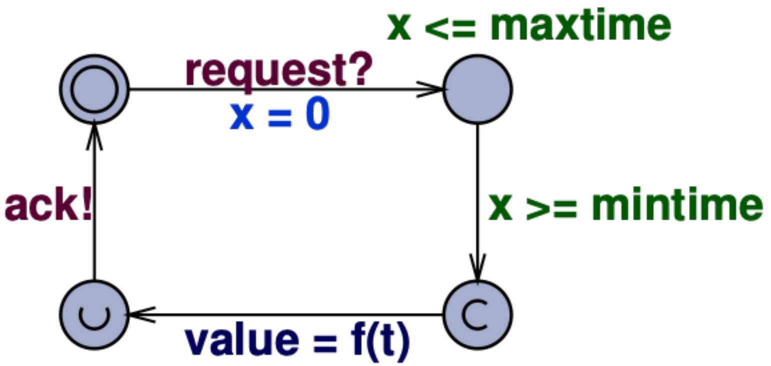








$$f_B(x,u) = \begin{cases} \dot{x}_1 = x_4 \cos(x_3) - x_5 \sin(x_3) \\ \dot{x}_2 = x_4 \sin(x_3) + x_5 \cos(x_3) \\ \dot{x}_3 = x_6 \\ \dot{x}_4 = u_1 + x_5 x_6 + w_1 \\ \dot{x}_5 = f_{y,f}(x,u,w) + f_{y,r}(x,w) - x_4 x_6 \\ \dot{x}_6 = a\frac{m}{J}\left(f_{y,f}(x,u,w)\right) - b\frac{m}{J}\left(f_{y,r}(x,w)\right) \end{cases}$$

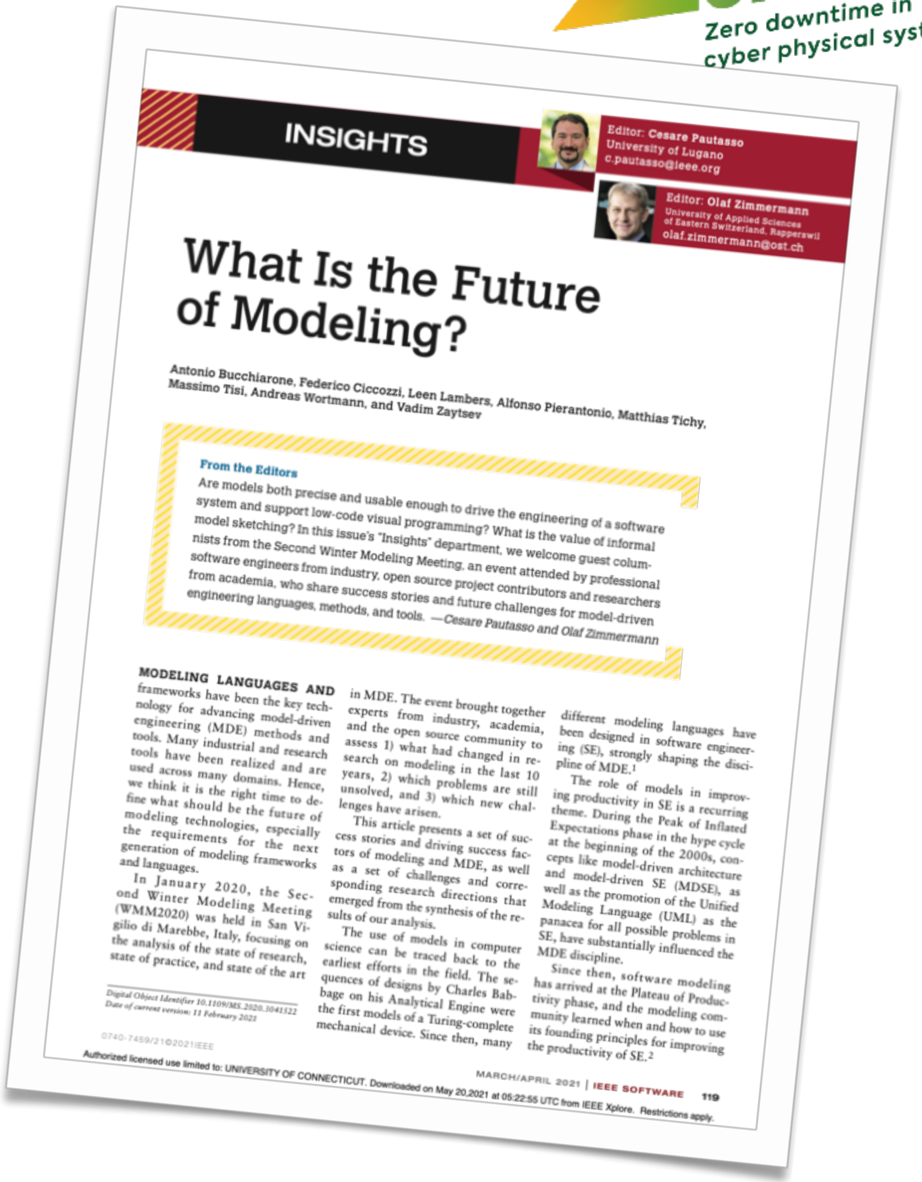


UNIVERSITY OF TWENTE. | Formal Methods & Tools

Waarom MBSE?

- gemeenschappelijk begrip van info
  - structuur en betekenis
- hergebruik van domeinkennis
- expliciete domeinaannames
  - redeneren, analyseren
- domein en ops zorgen apart
- afspraken over gebruik vastleggen
- consistente conversatie
  - verwarring voorkomen

DOI: [10.1109/MS.2020.3041522](https://doi.org/10.1109/MS.2020.3041522)



INSIGHTS

Editor: Cesare Pautasso
University of Lugano
c.pautasso@ieee.org

Editor: Olaf Zimmermann
University of Applied Sciences of Eastern Switzerland, Rapperswil
olaf.zimmermann@ost.ch

What Is the Future of Modeling?

Antonio Bucchiarone, Federico Ciccozzi, Leen Lambers, Alfonso Pierantonio, Matthias Tichy, Massimo Tisi, Andreas Wortmann, and Vadim Zaytsev

From the Editors

Are models both precise and usable enough to drive the engineering of a software system and support low-code visual programming? What is the value of informal model sketching? In this issue's "Insights" department, we welcome guest columnists from the Second Winter Modeling Meeting, an event attended by professional software engineers from industry, open source project contributors and researchers from academia, who share success stories and future challenges for model-driven engineering languages, methods, and tools. *—Cesare Pautasso and Olaf Zimmermann*

MODELING LANGUAGES AND frameworks have been the key technology for advancing model-driven engineering (MDE) methods and tools. Many industrial and research tools have been realized and are used across many domains. Hence, we think it is the right time to define what should be the future of modeling technologies, especially the requirements for the next generation of modeling frameworks and languages.

In January 2020, the Second Winter Modeling Meeting (WMM2020) was held in San Vigilio di Marebbe, Italy, focusing on the analysis of the state of research, state of practice, and state of the art in MDE. The event brought together experts from industry, academia, and the open source community to assess 1) what had changed in research on modeling in the last 10 years, 2) which problems are still unsolved, and 3) which new challenges have arisen.

This article presents a set of success stories and driving success factors of modeling and MDE, as well as a set of challenges and corresponding research directions that emerged from the synthesis of the results of our analysis.

The use of models in computer science can be traced back to the earliest efforts in the field. The sequences of designs by Charles Babbage on his Analytical Engine were the first models of a Turing-complete mechanical device. Since then, many different modeling languages have been designed in software engineering (SE), strongly shaping the discipline of MDE.[1]

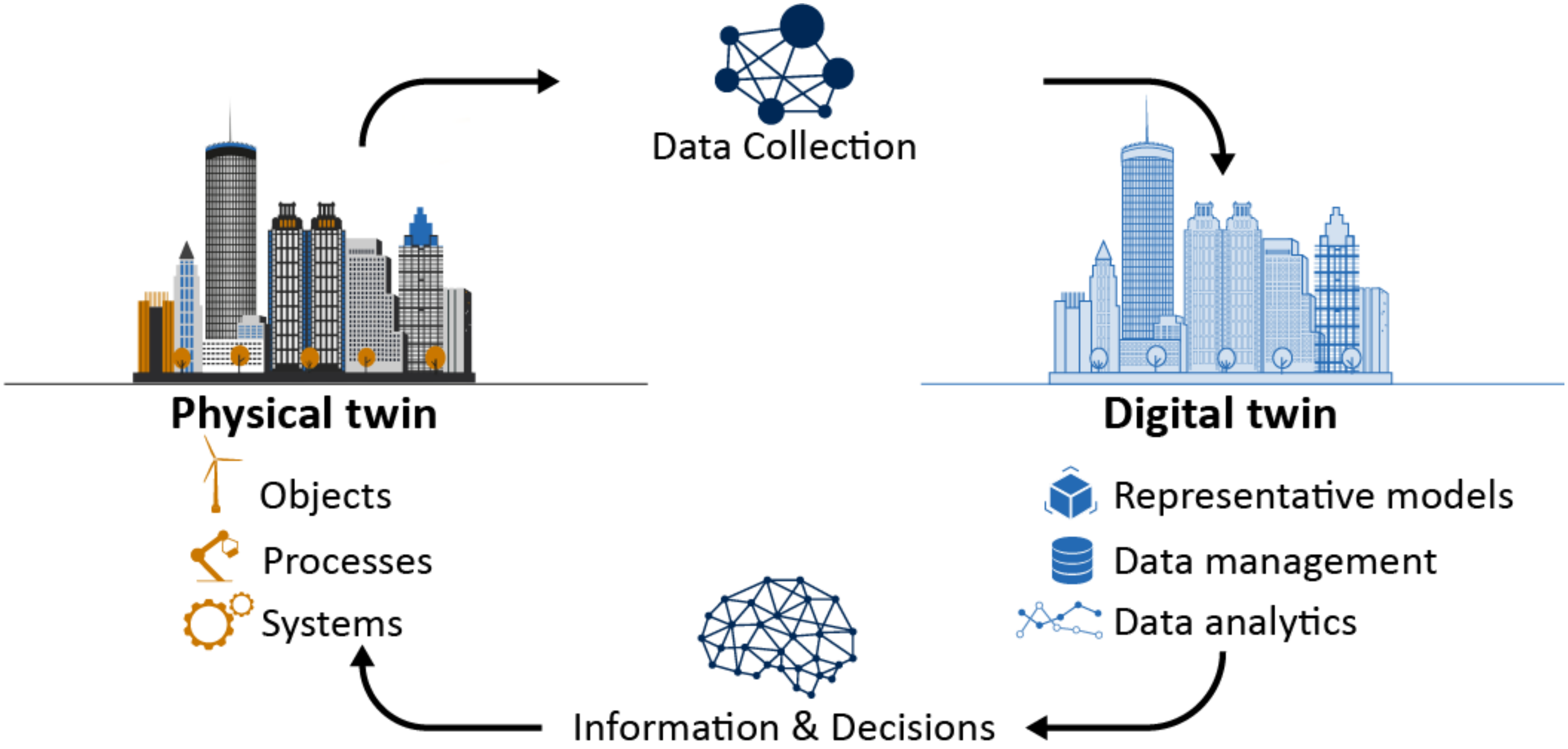
The role of models in improving productivity in SE is a recurring theme. During the Peak of Inflated Expectations phase in the hype cycle at the beginning of the 2000s, concepts like model-driven architecture and model-driven SE (MDSE), as well as the promotion of the Unified Modeling Language (UML) as the panacea for all possible problems in SE, have substantially influenced the MDE discipline.

Since then, software modeling has arrived at the Plateau of Productivity phase, and the modeling community learned when and how to use its founding principles for improving the productivity of SE.[2]

Digital Object Identifier 10.1109/MS.2020.3041522
Date of current version: 11 February 2021

0740-7459/21©2021IEEE

Digitale tweeling



https://www.linkedin.com/pulse/power-limitations-digital-twins-look-future-dalia-a-ibrahim/

Op naar nul uitvaltijd

- modelgebaseerde systeemtechniek
- domeinspecifieke talen
- modeltransformaties
- digitale tweeling

Reflections on the Lack of Adoption of Domain Specific Languages*

Federico Tomassetti[1] and Vadim Zaytsev[2,3]

[1] federico@tomassetti.me, https://tomassetti.me Strumenta, Torino, Italy
[2] vadim@grammarware.net, http://grammarware.net Raincode Labs, Brussels, Belgium
[3] Universiteit Twente, Enschede, The Netherlands

Abstract. Given all the different benefits that domain specific languages are reported to produce, why are they not a widely adopted practice? The essence of the question has roots in industrial experience of the two authors of this report, but it was put out as a discussion starter at OOPSLE 2020. During the discussion session itself, there were some possible reasons voiced and possibly related concerns expressed. In this report, we try to condense those in a short coherent text. Not claiming any generality and fully acknowledging the anecdotal nature of our evidence, we still think that such a conversation is useful to have within the SLE community.

1 Introduction

While this paper reflects the opinions of the authors, it has been strongly influenced by the helpful discussions at OOPSLE over the business and organisational problems causing domain specific languages (DSLs) not to be adopted. By the lack (or perhaps just dearth) of adoption here we mean that whenever software developers have a problem to solve, among the possible solutions to it, modelling the problem domain by means of creating a new domain-specific language or adopting an existing one, is rarely, if ever, the first option. Quite often it is being left out of this list altogether.

2 Advantages of DSLs

From research and practice, domain-specific languages are known to bring the following advantages:

– **domain-specific abstractions** [3, 7, 13, 17] to express commonly needed concepts even by non-developers, beyond what a library would allow [16];

* Copyright © 2020 for this paper by its authors. Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).

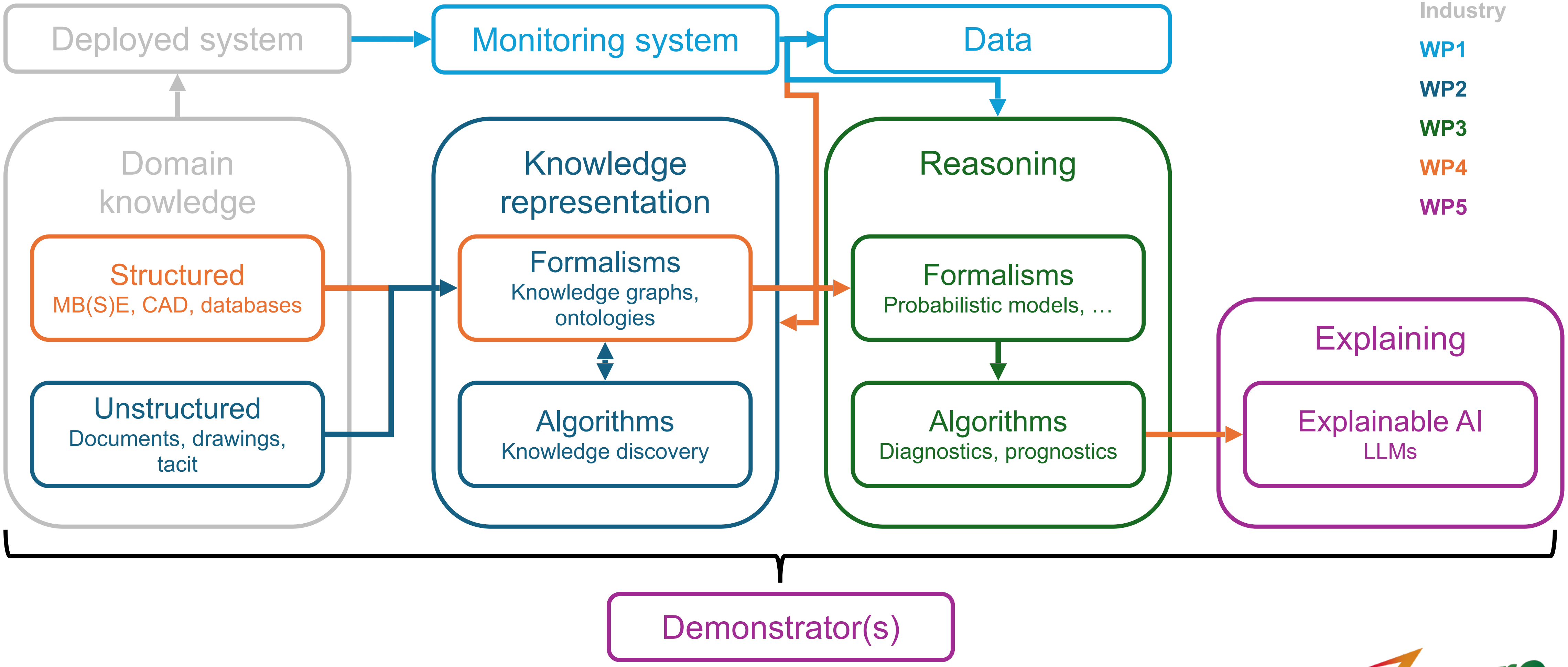
Doelen

- diagnosticeerbaarheid
- aanpasbaarheid
- betrouwbaarheid

Onzekerheid

- elimineren kan niet
- modelleren wel

ZORRO overview

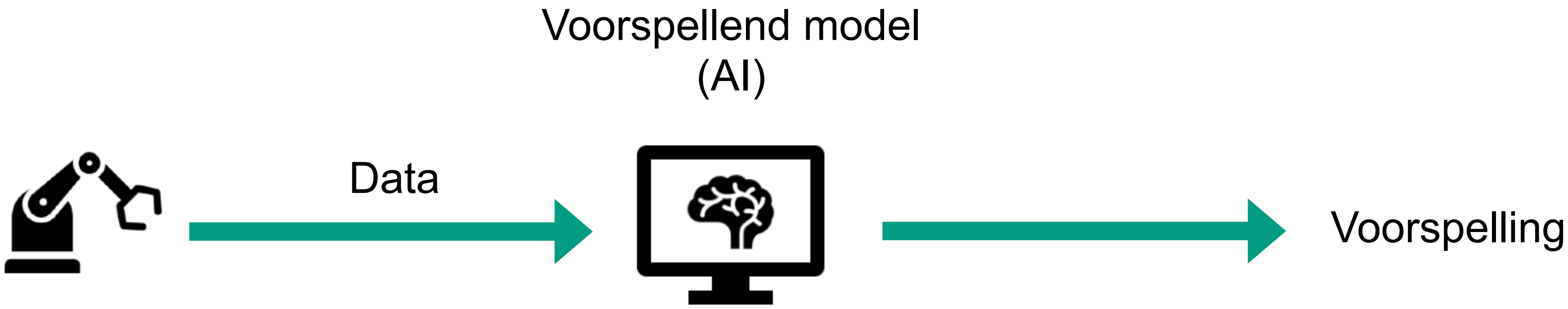


SAXION HOGESCHOOL

Zorro Zero downtime in cyber physical systems

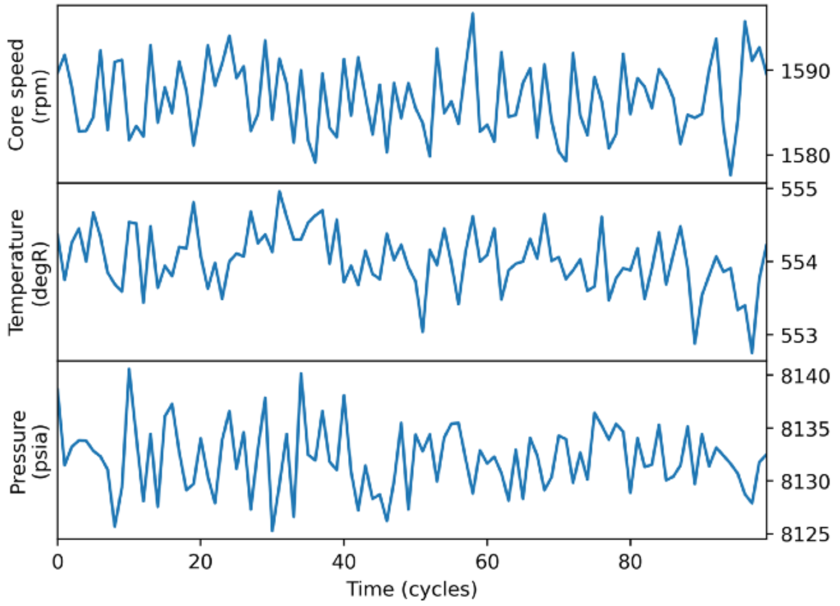
Industry 4.0

- Cyber-physical systems en Internet of Things
- Predictive maintenance

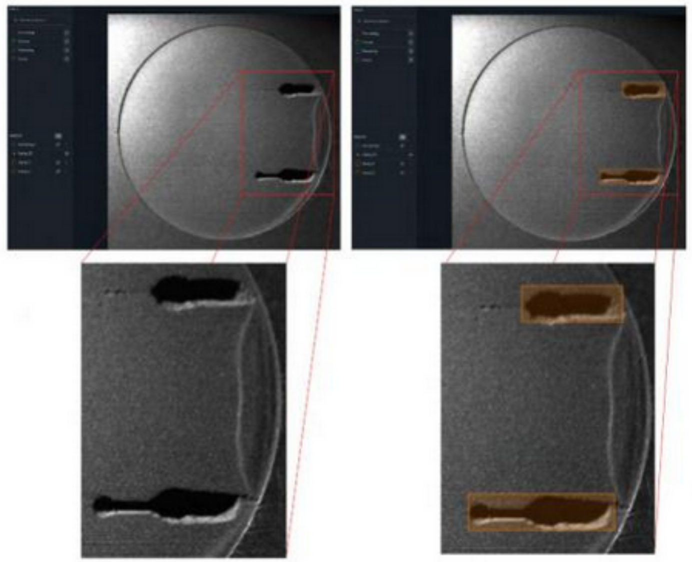


Industry 4.0

- Cyber-physical systems en Internet of Things
- Predictive maintenance



Sensor metingen

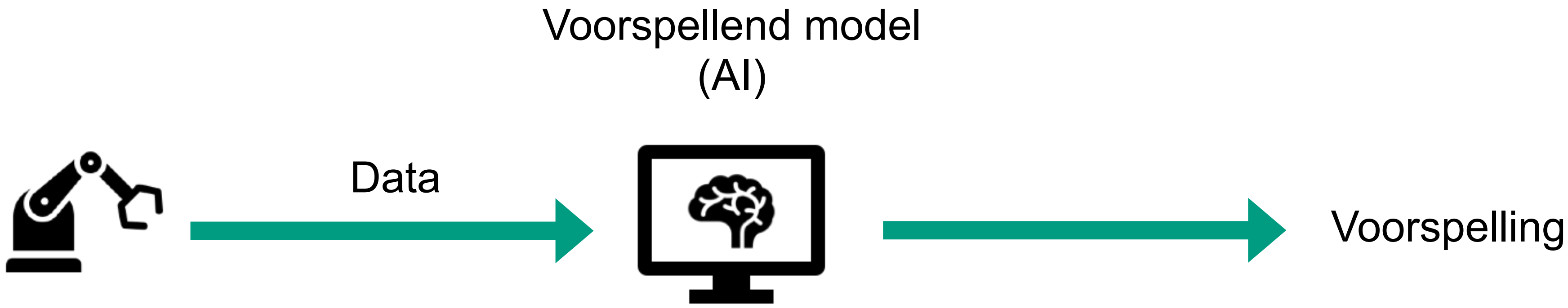


Afbeeldingen [1]

[1] Waheed, F., Höfig, K., & Riß, F. (2025, September). AI-Driven Data Management Framework for Quality Assurance in Additive Manufacturing: A Case Study on the TRUMPF TruPrint 1000. In *International Symposium on Model-Based Safety and Assessment* (pp. 3-17). Cham: Springer Nature Switzerland.

Industry 4.0

Hoe weten we dat de gemaakte voorspellingen betrouwbaar zijn?



Hoe weten we of we AI kunnen vertrouwen?

SAXION HOGESCHOOL

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).

Zorro
Zero downtime in cyber physical systems

Hoe weten we of we AI kunnen vertrouwen?

SAXION HOGESCHOOL

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).

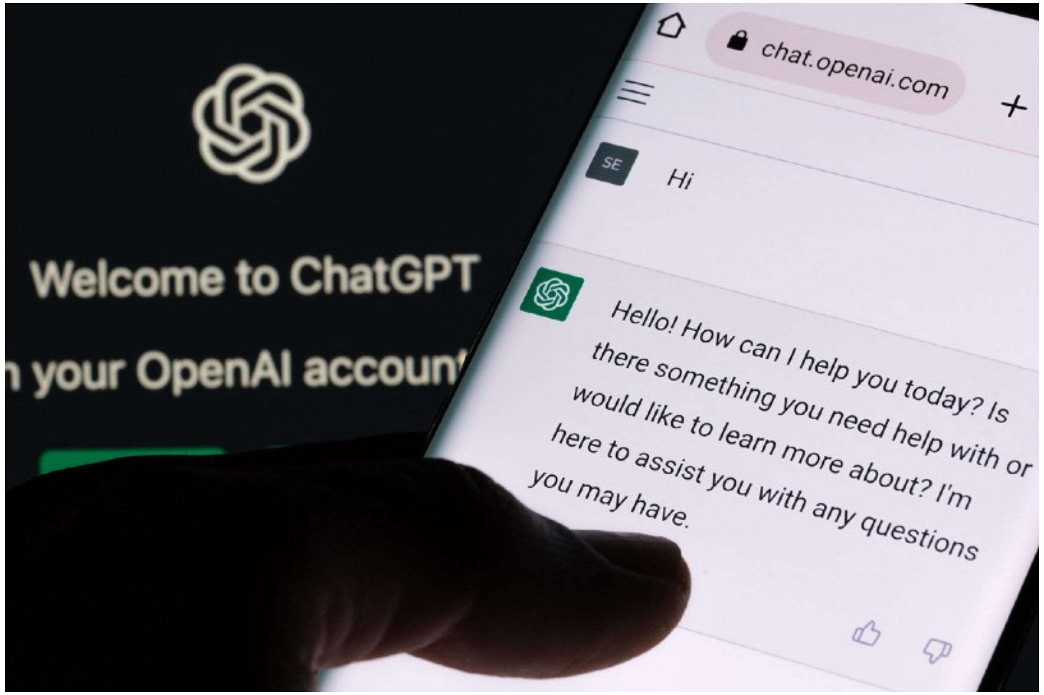
Zorro
Zero downtime in cyber physical systems

Hoe weten we of we AI kunnen vertrouwen?

Ribeiro, M. T., Singh, S., & Guestrin, C. (2016, August). " Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* (pp. 1135-1144).

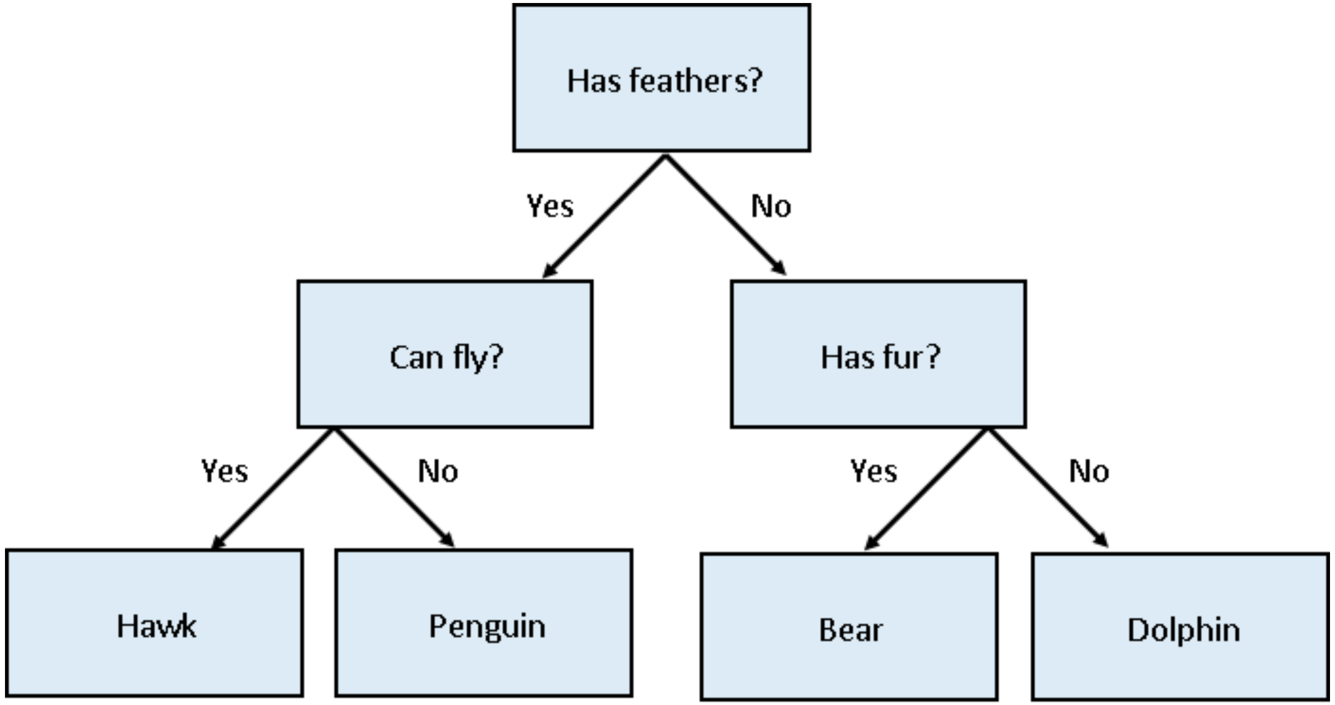
Hoe weten we of we AI kunnen vertrouwen?

AI blijft voorspellingen maken



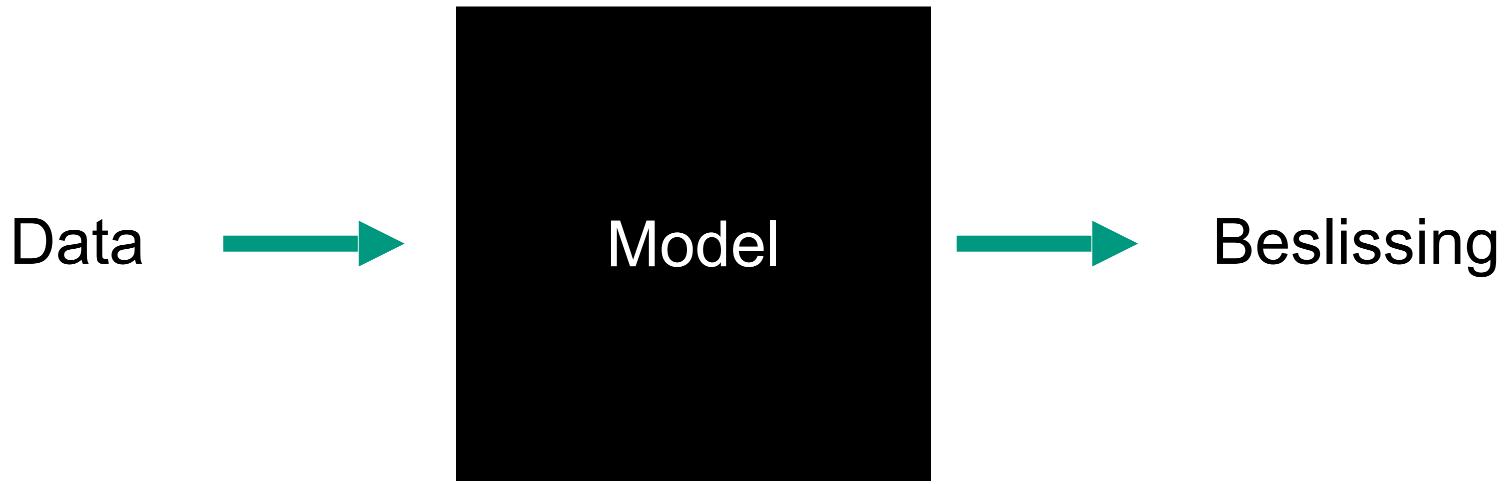
Hallucinaties in LLMs

Wat gebeurt er achter de schermen?



Beslisboom

Wat gebeurt er achter de schermen?



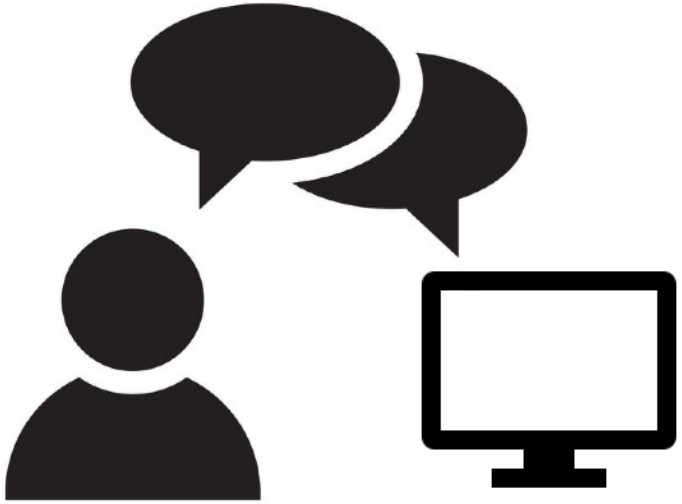
AI herkent patronen, maar we weten niet of dit de gewenste patronen zijn

Wat gebeurt er achter de schermen?

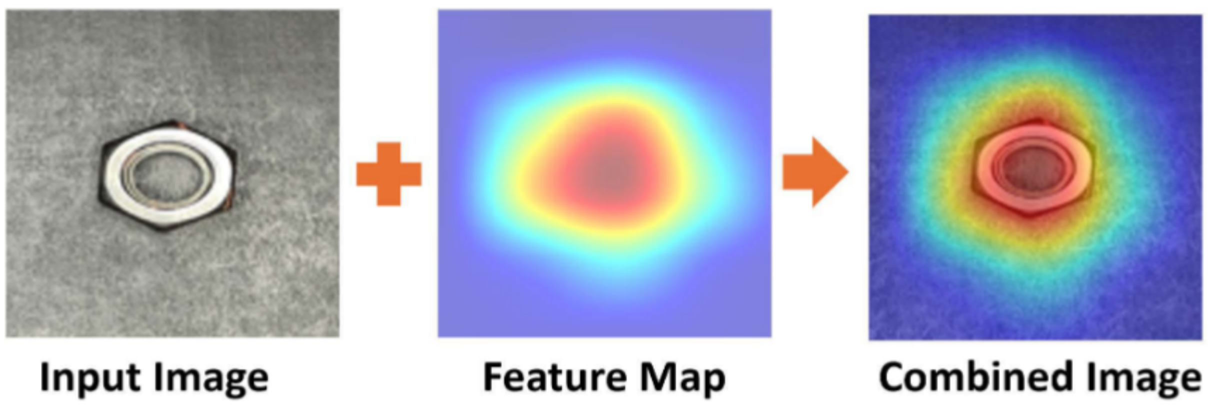
GPT-4 heeft 170.000.000.000.000.000.000 (170 triljoen!) parameters

Explainable AI

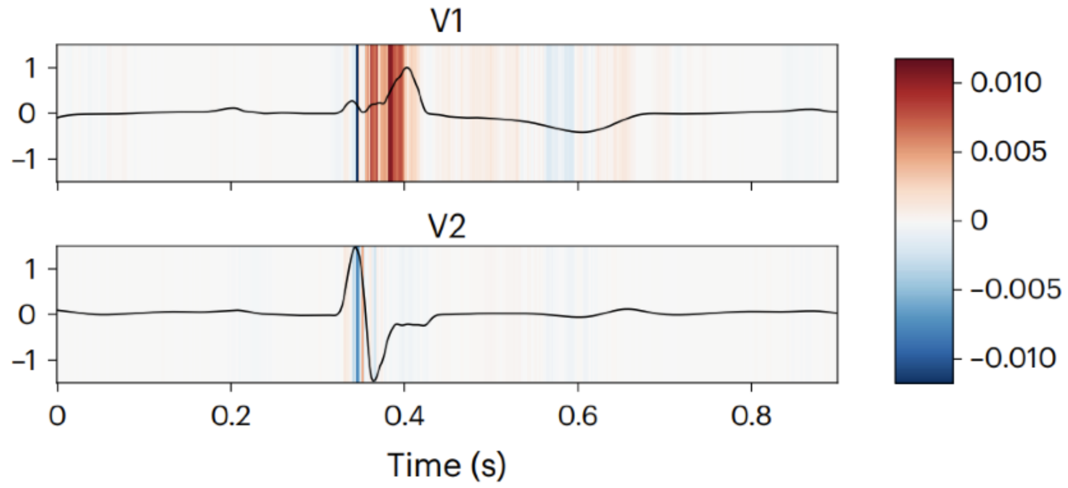
Explainable AI (XAI) bestaat uit methodes en tools die gebruikt kunnen worden om machine learning modellen begrijpelijk te maken voor mensen.



XAI in de praktijk



Onderdeelherkenning [2]

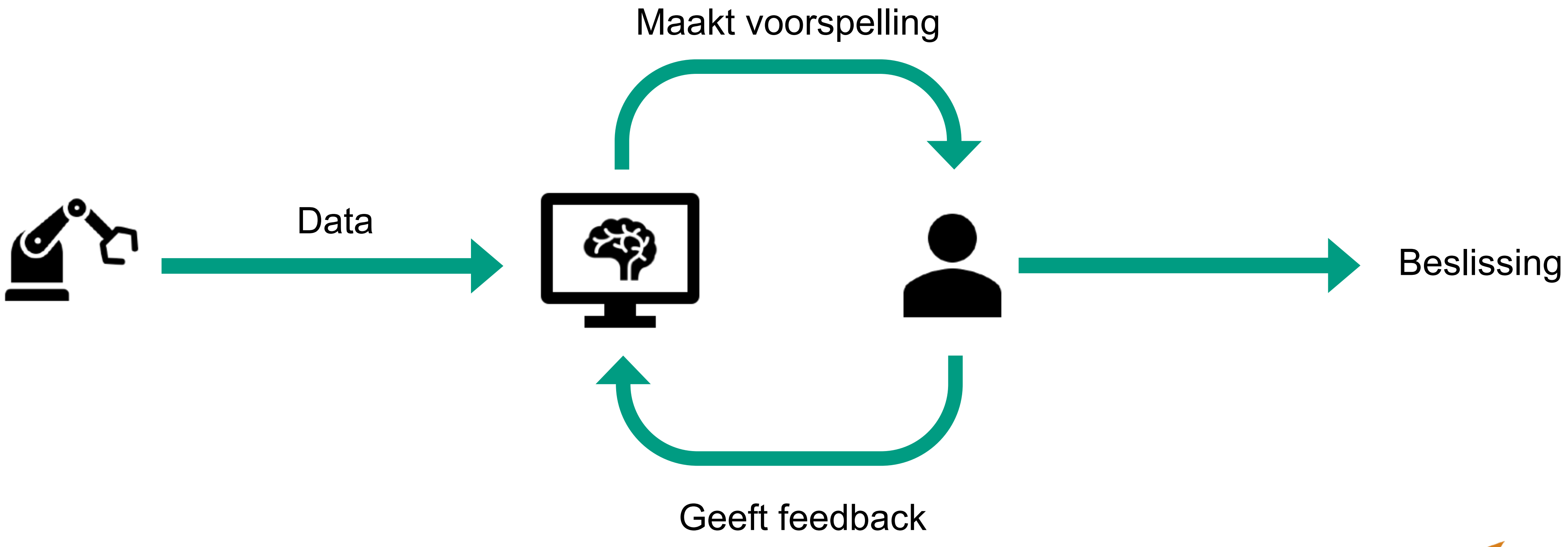


Diagnoses vanuit ECGs [3]

[2] Gill, A. S., Agarwal, A., Tummala, V., Lee, S. J., & Mears, L. (2024, November). Comparison of Explainable AI for Image Classification to Human Perception: A Case Study of Threaded Fasteners. In *ASME International Mechanical Engineering Congress and Exposition* (Vol. 88605, p. V002T03A094). American Society of Mechanical Engineers.
[3] Turbé, H., Bjelogrlic, M., Lovis, C., & Mengaldo, G. (2023). Evaluation of post-hoc interpretability methods in time-series classification. Nature Machine Intelligence, 5(3), 250-260.

Industry 5.0

"Industry 5.0 is emerging as an 'Age of Augmentation' when the human and machine reconcile and work in perfect symbiosis with one another" [5]



[5] Longo, F., Padovano, A., & Umbrello, S. (2020). Value-oriented and ethical technology engineering in industry 5.0: A human-centric perspective for the design of the factory of the future. *Applied Sciences*, *10*(12), 4182.