Research Statement

Motivation. Innovation drivers in modern engineering, such as digital twins in the Industry 4.0 initiative or the Internet-of-Things, are becoming increasingly heterogeneous and data-driven, while their development process is getting more elaborate and integrates novel development methodologies. These systems are distributed and contain integrated simulators, AI and cyber-physical components, while relying heavily on data streams. Some of these systems, like digital twins and other software-defined systems, even break the dichotomy of system and model by integrating a physical system and its own model into an overall system that interoperates with live data.

To provide the much-needed conceptual framework and tool support in software engineering, formal methods and software technologies must (1) provide modularity mechanisms to encapsulate the heterogeneous components and (2) incorporate formalisms for handling large amounts of data. My research targets *modelling, specification, and analysis of distributed, cyber-physical, and data-heavy systems* through formal software engineering approaches to improve their reliability and resilience. I am working on three connected research directions: *Modular Verification of Distributed Systems*, the *Language-based Integration of Semantic Data with Formal Methods* and *Software Engineering for Digital Twins*.

Current and Completed Research. I am participating in `PeTWIN`, an cooperation with industry partners on digital twins in the energy industry. Prior to this, I worked an interdisciplinary cooperation with Deutsche Bahn AG [8] to investigate the formal modelling of railway operations, based on formalisms developed for modelling cloud systems. In my current research, I am exploring formal methods that connect software-defined, cyber-physical systems with AI in the form of semantic data and ontologies [24, 16, 20]. The following presents current and completed research in more detail.

Modular Deductive Verification. One main technique to achieve modularity is a specification principle that decomposes a problem into smaller parts. I summarized approaches to modularity in verification by contributing to a recent overview article [27]. as part of an ISoLA track on modular verification co-organized by me. For object-oriented software, the most dominant notion in a *sequential* setting is the one of method contracts. I have developed an adequate notion of *cooperative* method contracts for Active Objects [28, 25], a concurrency model for distributed system based on actors and futures. Active Objects have been applied successfully in modelling and analysis of cloud systems, in particular to analyze deployment.

The verification of cooperative method contracts uses logical proof calculus that directly interfaces with static analyses. An implementation is available and maintained in the open source `Crowbar` symbolic execution engine [4]. I also explored the other direction, integration of a proof calculus into a static analysis, in a deadlock checker tool [44]. Starting with method contracts for Active Objects, I extended the notion of method contracts for distributed systems in two directions.

I extended Session Types, a *top-down* specification approach, to the Active Objects concurrency model [29], which generates local types from a global specification. Session types for Active Objects can be combined with invariants and *bottom-up* method contracts. Both specification approaches are unified through the Behavioral Program Logic (BPL) [3], a formalism that combines heavyweight symbolic execution, behavioral/local session types, and dynamic logic. These types can also result in additional global proof obligations, which can be checked either statically [34], or ensured through runtime verification [26].

Secondly, I have extended the Active Object concurrency to *Hybrid Active Objects* [7], where each encapsulated actor has internal continuous dynamics described by differential equations. Hybrid Active Objects model *distributed cyber-physical* systems and are suited for specification, verification, and simulation. I developed a simulator, a modeling and verification approach for cyber-physical systems with controlers running in the cloud [12] using a contract mechanism [2].

Integrating Formal Methods with Knowledge Representation. The Semantic Web is a set of logic-based knowledge representation tools for data storage, domain formalization, and efficient reasoning over data. It is one of the success stories of formal approaches in computer science and allows a formal treatment of *data*.

I conduct research on its integration with formal methods on a software-level and have introduced *semantically lifted programs* [1]: object-oriented programs that use semantic web technologies for data-related actions, such as database connections, debugging, and reflection. Semantically lifted programming is based on object-oriented programming and is natural to users with mainstream programming experience, while giving safety properties through an extended type system [39] and a query-based connection between ontological and object-oriented class models [17].

Recently, I have used semantically lifted programs to develop a simulator for geological processes [**?**], that allows a natural programming of geological processes and yet dircetly connects to existing geo-phjysical ontologies for the contained data. In the same domain, I have investigated the use of knowledge graphs to use ontolgies in a runtime enforcement mechanism to ensure the correctness of such semantically lifted simulators with respect to their application domain [13].

Software Engineering for Digital Twins Semantically lifted programs are a key tool for applying formal methods to digital twins, and I recently organized a workshop on Applications of Formal Methods in Digital Twins [47]. They allow us to connect with the rich asset models and data streams that govern engineering systems while remaining in a fully formal framework.

Preliminary studies in this paradigm show that they are indeed suited to developing the coordinating core of the digital twin: Semantically lifted programs can interpret their states as part of an asset model and self-adapt their structure to match the physical system, i.e., ensure that the system is *twinned* correctly [16]. Currently, this is done through runtime verification [20], and I am currently investigating its static verification. Semantically lifted programming is implemented in the Semantic Micro Object Language, whose open source implementation I actively develop.

Beyond semantically lifted programming, I am currently collaborating with two external research groups. For one, I work with the developers of a Digital-Twin-as-a-Service platform [10], to use semantic lifting in a microservice platform. For another, I am working on the use of knowledge graphs to reuse experimental results throughout the whole lifecycle of a system [37]. Lastly, I am managing the Digital Twin lab of the ASR group at the university of Oslo, where I develop software architectures for Digital Twins using a greenhouse, with the aim to exploit synergies with other Digital Twin activities at the department.

Furthermore, I am investigating *interoperable* variability [31, 30], systems that contain several variants of one component managed by the same product line. These variants must be encapsulated, their relation to each other must be modeled and their interactions must be ensured to be safe. To this end, we developed variability modules [23] and implemented them in the ABS language, a long-running implementation project where I maintain the variability layer.

Planned Research I plan to focus my research activities towards *software architectures* and *tools support* for Digital Twins using the following two directions: (1) The development of verification and analysis approaches for software interacting with semantic data, and (2) the connection of hybrid distributed systems with industry standards for simulators.

(1) There is no notion of behavioral specification for software that interacts with AI tools for semantic technologies, such as knowledge graphs. The use of ontologies for domain knowledge during program execution is similarly unexplored. I plan to develop notions of invariants and method contracts for systems consisting of software and knowledge graphs. This will enable specifying interactions, such as loading/writing data in presence of reasoning.

Semantically lifted objects will form the basis for contracts and further integration. For example, we have preliminary results on the optimization of semantically lifted objects that shows that the semantic web offers notions of modularity that can be exploited to localize data access [41]. I plan to leverage the class generation based on SPARQL queries [17], from semantically lifted objects to Java and investigate the generation of libraries with static guarantees for data loading in mainstream languages.
Finally, I am currently investigation the possibilities of using such contracts for non-functional properties, especially data flow and privacy/security properties where knowledge graphs and programs must be analyzed, and specified, as a unit.

(2) Digital twins are software-defined, cyber-phyiscal, and distributed systems. Modeling and verification mechanisms, for example Hybrid Active Objects, need a connection to established engineering practices to be practical: They must be connected to industrial standards and available tools, in particular simulators and data stream monitors. We have described the principal connection with asset models, and the next step will be to connect with an industrial project for asset modelling, possibly the Asset Information Modelling Framework (IMF) of the READi project, with which I started the first preliminary studies. I plan to use the IMF to *automatically* generate correct-by-construction digital twins from a given asset.

To connect with simulators, I plan to simulate the continuous behavior of hybrid objects, which are natural to use with systems using the cloud, with off-the-shelf simulators through the Functional Mock-up Interface, an industrial standard for co-simulation. We can load simulators described by the Functional Mock-up Interface and semantically reflect them [18], and the next step will be to establish their connection with hybrid systems verification.

In the mid term, I plan to establish Semantic Web technologies as formalisms in formal methods for data-heavy systems, use a language-based approach for their integration into software and subsequent analyses, and integrate formal methods with industrial standards to improve their adaptation.

Teaching Statement

Personal Approach. Universities are places that provide opportunities and chances for our students – I see a teacher's main task as enabling the student to make the decisions fitting their future plans and in enabling them to understand current research trends, which will influence the future of the society our students will form. Thus, the goal of my teaching activities is to offer the students an education that both prepares them for the job market and engages them actively in research activities. I firmly believe that involving students in research is not an elitist activity for the very best – every student should have the choice and chance to participate in research at some point and level.
In this endeavor, formal methods are crucial: Situated between the engineering and theory-oriented subfields of computer science, it is here where students learn how theoretical concepts offer new perspectives in practice and it is here where they learn how industrial applications motivate current research.
I am a native German speaker and can teach courses in German. I could imagine contributing to undergraduate teaching in, for example, the course "Objektorientierte Programmierung" or "Softwaretechnologie", and to graduate level teaching by giving, for example, a course on analysis of hybrid systems or verification of object-oriented software. In the mid term, I would like to develop a lab project/seminar where students can develop a Digital Twin using AI and simulation components with state-of-the-art frameworks.

Experience. I have taught two courses I have designed and taught a course on the Analysis of Hybrid Systems at the Technical University of Darmstadt and was involved in further teaching activities where I supervised students in different contexts. Due to the constraints from the funding on my position, I was not allowed to participate in teaching at the University of Oslo prior to the autumn term'22. The following describes my teaching experience in more detail.

Lecturing. I designed and taught the *Analysis of Hybrid Systems* graduate-level course at the Technical University of Darmstadt in the summer term 2020. The course used a contrasting approach to introduce hybrid systems: A logic-based and an automata-based formalism were introduced and used to model the same phenomena. The course focused on modeling decisions and choosing the most suited formalism. Additionally, the course has a lab where the students applied state-of-the-art tools: SpaceEx and KeYmaera X. I based the logic-based part of the course on the book "Logical Foundations of Cyber-Physical Systems", and created the material for the automata-based part from scratch.

In the autumn terms 2022, I was lectured and modernized the *Models of Concurrency* (IN5170) graduate-level course at the University of Oslo, where I developed and taught the lectures for static analyses, which introduced linear and behavioral type systems and the Rust ownership system. These lectures were accompanied by exercises with Rust and its Session Types library. I modernized the course further for the autumn term 2023 by (1) introducing Go to teach asynchronous methods, and (2) integrating Java as a language for new lab sessions on shared memory.

Teaching Assistance. I was a teaching assistant for the *Automated Theorem Proving* course at the Technical University of Darmstadt in the summer term 2018. I conducted the exercises and discussed the solutions with the students. Based on my experience, we developed a special teaching tool (`kbar.app`) specifically to increase student engagement, based on studies in *interactive algorithm visualization* and *serious games*. The concept behind it is published [40].

Thesis Supervision. I currently co-supervise 2 Ph.D. students and 6 master students at the University of Oslo. I have supervised 5 master's theses, 8 bachelor's theses, and 1 study thesis, and was an internal examiner for 2 further master's theses. Three theses resulted in peer-reviewed publications, where the student actively co-authored the publication.

Further Supervision. I have supervised further master and bachelor students in the following contexts: (1) I have supervised 4 *bachelor projects*, a mandatory module at TU Darmstadt where students implement a project under non-technical supervision, similar to the "Praktikum zur Softwaretechnologie" at BUW. (2) I supervised 3 undergraduate student assistants (German: *Studentische Hilfskraft*) who implemented extensions of our research tools. (3) I supervised students in 3 seminars *Software Failures*, *Symbolic Execution* and *Actor Languages*. (4) I supervised two Erasmus+ trainees in 2022-2023, who investigated tool support for Java debugging and programming with ontologies.

Eduard Kamburjan

Personal Data

Name	Eduard Kamburjan
Date of Birth	19.02.1990, in Alma-Ata, Kazakhstan
Citizenship	German
Home Contact	*Phone (mobile):* (+49) 160 1154535 *Mail:* eduard.kamburjan@gmx.de
Work Contact	University of Oslo Postboks 1080 Blindern, 0373 Oslo, Norway *Mail:* eduard@ifi.uio.no
Website	`edkamb.github.io`

Education

2016 – 2020 **PhD/Dr. rer. nat.**, *Technical University of Darmstadt*.
Passed with distinction (summa cum laude), advisor: Prof. Reiner Hähnle, dissertation: *Modular Verification of a Modular Specification: Behavioral Types as Program Logics*

2014 – 2016 **M.Sc.**, *in Computer Science, Technical University of Darmstadt*, Total grade: 1.36.

Employment

Since 2023 **Senior Lecturer**, ***University of Oslo***, *Reliable Systems Group*.
- Digital twins and asset modeling in the energy industry in cooperation with Equinor
- Integration of techniques from the semantic web with programming languages and approaches to co-simulation

2020 – 2023 **Postdoctoral Fellow**, ***University of Oslo***, *Analytical Solutions and Reasoning Group*.
- Digital twins in the petroleum industry in cooperation with Equinor, Petrobras and Shell
- Integration of techniques from the semantic web with programming languages and approaches to co-simulation

2016 – 2020 **Research Assistant**, ***Technical University of Darmstadt***, *Software Engineering Group*.
- Modeling and verification of railway operations in cooperation with the Institute for Railway Engineering and DB Netz AG
- Development of new specification and verification approaches for Active Objects

2013 – 2016 **Student Assistant**, ***Technical University of Darmstadt***, *Software Engineering Group*.
Implementing loop invariant inference in the `KeY` system.

Community Service

PC Chair	*Workshop on Applications of Formal Methods and Digital Twins* (co-located with FM'23)
Track Chair	FASE'22 (*Artifact Evaluation* track) ISoLA'20 (*Modularity and (De-)composition in Verification* track)
PC Member	SAC'24 (*Software Verification and Testing* track) SemIIM@ISWC'23 FTSCS'23 MPM4CPS'23 ICE'23 PhD@iFM'23 (*PhD Symposium*) SAC'23 (*Software Verification and Testing* track) ANNSIM'23 (*Cyber-Physical Systems* track) ECOOP'23 (*Artifact Evaluation* track & *Extended Review Committee*) FASE'23 (*Artifact Evaluation* track) SPLC'22 (*Journal First* track) SPLC'21 (*Journal First* track) ECOOP'21 (*Artifact Evaluation* track) OOPSLA'21 (*Artifact Evaluation* track) SPLC'18 (*Challenges* track)
Guest Editor	Science of Computer Programming, Special Issue on FASE'22 Artifacts LNCS Volume on the State of the Art in Active Objects
External Reviewer (Conferences)	FASE'23,'22,'21,'20,'19,'18, FM'23,'21,'19, NFM'23, iFM'23,'22,'20,'19,'17, TAP'22,'21,'19, COORDINATION'22,'21, ISoLA'22, SEFM'22, FORTE'22, CoSim Workshop'21, FTfJP'21, DaLi'20,'19, ICTAC'20, FOSSACS'20, TABLEAUX'19, CADE'19, IJCAR'18, CPP'18 *Total: 37 conference and workshop articles for 33 venues as an external reviewer.*
(Journals)	IEEE Transactions on Software Engineering, IEEE Transactions on Automation Science and Engineering, Journal of Functional Programming, Software and System Modeling, Science of Computer Programming, Computing, Journal of Systems and Software, Journal of Logical and Algebraic Methods in Programming *Total: 12 journal articles for 8 journals as an external reviewer.*
(Grants)	Dutch Research Council (NWO)
Other	ETAPS community blog (Editorial Board, since 2023)
Organizer	ABS Workshop'23 ABS Workshop'21 ABS Workshop'18

Invitations

03.2024 (forthcoming)	Lorentz Seminar on *Contract Languages: Expressiveness, Abstraction, Interoperability, and Applications*
02.2024 (forthcoming)	Dagstuhl Seminar 24061 on *Are Knowledge Graphs Ready for the Real World?*
01.2024 (forthcoming)	Dagstuhl Seminar 24051 on *Next Generation Protocols for Heterogeneous Systems*

01.2024 (forthcoming) WAKERS workshop on *Digital Engineering and Knowledge Representation*

03.2023 18th *CAMPaM workshop on Twinning For and By Systems Engineering* at the Institut d'Études Scientifiques de Cargèse

11.2022 Dagstuhl Seminar 22451 on *Principles of Contract Languages*

09.2021 Dagstuhl Seminar 21372 on *Behavioural Types: From theory to practice*

Awards and Stipends

Scholarship Kristine Bonnevie travel stipend 2023, from the Faculty of Mathematics and Natural Sciences of UiO for young excellent researchers, worth *24 500 NOK*

Award Best Research Paper ESWC 2022

Research Visits and Summer Schools

03.2023 One month at the Aarhus University/Denmark, hosted by Prof. Peter Gorm Larsen

02./05.2022 Two 1 week stays at the University of Bergen/Norway, hosted by Prof. Crystal Din

11.2019 One week at the University of Turin/Italy, hosted by Prof. Ferruccio Damiani

2017 Marktoberdorf Summer School

Industrial Collaborations

2016 – 2020 With DB Netz AG, as project member of the `FormbaR` project, funded by DB AG.
Modelling railway operations books to increase maintainability.

since 2020 With Equinor, as project member of the `PeTWIN` project, funded by the Norwegian Research Council.
Combining knowledge graphs and programs to handle asset models in digital twins of petroleum industry facilities.

Professional Membership

- The Society for Modeling & Simulation International (SCS)
- Association for Computing Machinery (ACM)
- Formal Methods Europe (FME)

External Collaborations

- With Prof. Ferruccio Damiani (University of Turin/Italy) and Dr. Michael Lienhardt (ONERA/France) [15, 12, 23, 30, 31, 33].
  On variability modeling and type systems. Since 2017. On-going.
- With Prof. Crystal Din (University of Bergen/Norway) [13, 25, 28, 34].
  On semantic constraints of programs and verification. Since 2016. On-going.
- With Prof. Dilian Gurov (KTH/Sweden) [27].
  On modularity in deductive verification. Since 2020. On-going.
- With Prof. Peter Gorm Larsen and Prof. Claudio Gomes (Aarhus University/Denmark) [10].
  On digital twin engineering. Since 2023. On-going.

- With Prof. Hans Vangheluwe (University of Antwerp/Belgium) and Prof. Loek Cleophas (TU Eindhoven/Netherlands) [37].
  On digital twin engineering. Since 2023. On-going.
- With Dr. Sandro Rama Fiorini (IBM Research/Brazil) [19].
  On cognitive effects in modeling. 2021-2022
- With Dr. Stefan Mitsch (CMU/USA) [7].
  On hybrid active objects. 2017-2019.

Presentations at International Venues with Peer-Review

1. *"Deltas for Functional Programs with Algebraic Data Types"*
   SPLC'23, Tokyo, Japan [15]
2. *"Digital Twin Reconguration Using Asset Models"*
   ISoLA'22, Rhodos, Greece [16]
3. *"Twinning-by-Construction: Ensuring Correctness for Self-Adaptive Digital Twins"*
   ISoLA'22, Rhodos, Greece [20]
4. *"Knowledge Structures over Simulation Units"*
   ANNSIM'22, virtual [18]
5. *"Never Mind the Semantic Gap: Modular, Lazy and Safe Loading of RDF Data"*
   ESWC'22, Hersonissos, Greece [17]
6. *"Designing Distributed Control with Hybrid Active Objects"*
   ISoLA'21, Rhodos, Greece [24]
7. *"Optimizing Semantically Lifted Programs through Ontology Modularity"*
   NWPT'21, Reykjavik, Iceland [41]
8. *"Variability Modules for Java-like Languages"*
   SPLC'21, online [23]
9. *"From Post-Conditions to Post-Region Invariants"*
   HSCC'21, online [2]
10. *"Programming and Debugging with Semantically Lifted States"*
    ESWC'21, online [1]
11. *"Increasing Engagement with Interactive Visualization: Formal Methods as Serious Games"*
    FMTea@FM'21, online [40]
12. *"Type Checking Semantically Lifted Programs via Query Containment under Entailment Regimes"*,
    DL Workshop'21, online [39]
13. *"Asynchronous Cooperative Contracts for Cooperative Scheduling"*
    SEFM'19, Oslo, Norway [28]
14. *"Behavioral Program Logic"*
    TABLEAUX'19, London, UK [3]

15. *"Tool Support for Validation of Formal System Models"*
    F-IDE'19, Porto, Portugal [42]

16. *"Interoperability of software product line variants"*
    SPLC'18, Gothenborg, Sweden [31]

17. *"Stateful Behavioral Types for Active Objects"*
    iFM'18, Maynooth, Ireland [29]

18. *"Prototyping Formal System Models with Active Object"*
    ICE'18, Madrid, Spain [43]

19. *"Detecting Deadlocks in Formal System Models with Condition Synchronization"*
    AVoCS'18, Oxford, UK [44]

20. *"Asynchronous Cooperative Contracts for Cooperative Scheduling"*
    NWPT'18, Oslo, Norway (abstract of [28])

21. *"Deductive Verification of Railway Operations"*
    RSSRail'17, Pistoia, Italy [32]

22. *"Uniform Modeling of Railway Operations"*
    FTCSC'16, Tokyo, Japan [45]

23. *"Session-Based Compositional Analysis for Actor-Based Languages Using Futures"*
    ICFEM'16, Tokyo, Japan [34]

24. *"Session Types for ABS"*
    NWPT'15, Reykjavik, Iceland (abstract of [34])

Invited Presentations

1. *"The Semantically Reflected Digital Twin"*
   Tutorial, CAMPaM'23, with Einar Broch Johnsen, 23.03.23

2. *"Semantically Lifted Digital Twins"*
   Formal Methods Research Seminar, KIT, Karlsruhe, 09.01.23

3. *"Towards Contracts for Semantically Lifted Programs"*
   Dagstuhl Seminar 22451, 07.11.22

4. *"The Semantically Reflected Digital Twin"*
   Tutorial, ICTAC Summer School, with Einar Broch Johnsen, 26.10.22

5. *"Semantically Lifted Programming"*
   TCS Seminar, KTH Stockholm, 16.09.22

6. *"Semantic Programming"*
   BLDL Group Seminar, University of Bergen, 19.05.22

7. *"Session Types as Program Logics"*
   Dagstuhl Seminar 21372, 14.09.21

Other Presentations

1. *"Digital Twins for Ecological Systems"*
   Green Data Lab Conference'23, Ås, Norway
2. *"Monitoring of Self-Adaptive Digital Twins"*
   COEMS Forsterk Seminar'22, Tromsø, Norway
3. *"25 Years of FASE/ETAPS"*
   ETAPS'22, Munich, Germany, with Gabriele Taentzer
4. *"Hybrid Active Objects"*
   Workshop on Distributed Hybrid Systems'18, Amsterdam, Netherlands
5. *"Prototyping Formal System Models with Active Objects"*
   Workshop on Actors and Active Objects'17, Turin, Italy
6. *"The future Use Cases of Formal Methods in Railways"*
   Scientific Railway Signalling Symposium'18, Darmstadt, Germany
7. *"Formalisierung von betrieblichen und anderen Regelwerken – Das FormbaR-Projekt"*
   Scientific Railway Signalling Symposium'17, Darmstadt, Germany
8. *"Context-aware Trace Contracts"*
   International KeY Symposium'23, Bergen, Norway
9. *"KeY-Style Verification for ABS and Hybrid ABS"*
   International KeY Symposium'21 (online)
10. *"Behavioral Program Logic"*
    International KeY Symposium'19, Manigod, France
11. *"Hybrid Active Objects"*
    International KeY Symposium'19, Manigod, France
12. *"Update on KeY-ABS"*
    International KeY Symposium'17, Rastatt, Germany
13. *"Session Types for ABS"*
    International KeY Symposium'16, Manigod, France
14. *"Modeling Railways with ABS and KeY-ABS"*
    International KeY Symposium'16, Manigod, France
15. *"Abstract Object Creation for an Explicit Heap Representation"*
    International KeY Symposium'14, Bühl, Germany
16. *"Context-aware Trace Contracts"*
    ABS Workshop'23, Lyon, France
17. *"ypes and Verification for Delegated Control of Hybrid Objects"*
    ABS Workshop'23, Lyon, France
18. *"Crowbar and Hybrid ABS"*
    ABS Workshop'21 (online)

19. *"Behavioral Program Logic"*
ABS Workshop'19, Amsterdam, Netherlands

20. *"Hybrid Active Objects"*
ABS Workshop'19, Amsterdam, Netherlands

21. *"Asynchronous Method Contracts for ABS"*
ABS Workshop'18, Darmstadt, Germany

22. *"Experiences with await on Fields"*
ABS Workshop'17, Oslo, Norway

23. *"Session Types for ABS"*
ABS Workshop'16, Oslo, Norway

Teaching Experience

Course	*IN5170 Models of Concurrency*, Autumn Term'23	*Lecturer*
Course	*IN5170 Models of Concurrency*, Autumn Term'22	*Lecturer*
Course	*IN3040 Programming Languages*, Autumn Term'23	*Guest Lecturer*
Course	*IN3040 Programming Languages*, Autumn Term'22	*Guest Lecturer*
Course	*Analysis of Hybrid Systems*, Summer Term'20	*Lecturer*
Course	*Automatic Theorem Proving*, Summer Term'18	*Teaching Assistant*
Seminar	*Actor Languages*, Winter Term'19	*Teaching Assistant*
Seminar	*Symbolic Execution*, Summer Term'19	*Teaching Assistant*
Seminar	*Software Failures*, Summer Term'17	*Teaching Assistant*
Grading	*INF113 Operational Systems*, Autumn'22 at the University of Bergen.	*External Examiner*
Grading	*INF113 Operational Systems*, Autumn'21 at the University of Bergen.	*External Examiner*
Project	*KalkulierbaR*, Winter Term'20	*As supervisor*
Project	*KollaborierbaR*, Winter Term'19	*As supervisor*
Project	*VisualisierbaR II*, Winter Term'18	*As supervisor*
Project	*VisualisierbaR I*, Winter Term'17	*As supervisor*

Professional Training: Seminar "Researching Teaching" (German: *Forschende Lehre*) on integrating research questions into teaching.

PhD Student Supervision

Co-supervisor *Trace-Based Symbolic Execution*, Åsmund Kløvstad
On-going since 08.2022, main examiner: Prof. Einar Broch Johnsen

Co-supervisor *Composition of Multi-Scale Models for Digital Twins*, Riccardo Sieve
On-going since 09.2023, main examiner: Prof. Einar Broch Johnsen

Thesis and Student Supervision

Master *A backend for semantic digital twins*
by Alexander Wennevold Silva, UiO, on-going

Master	*Developing a semantic digital twin framework with live and historical sensor data* by André Finstad, Sander Lygren Sigmundstad and Janaaththan Manokaran, UiO, on-going
Master	*A Climate Barometer for the Oslo Fjord Using a Digital Twin Architecture* by Ingvild Emilie Øvsthus, UiO, on-going
Master	*Semantic framework for reconfiguration of digital twins* by Mariann Løtvedt, UiO, on-going
Master	*Mobile Assets in Semantic Digital Twins* by Oscar Lund Ramstad, UiO, 2023
Master	*Semantic Debugging for the JVM* by Anton Wolf Haubner, TUD, 2022
Master	*Implementing Variability-aware Modules* by Melissa Mendoza, TUD, 2020
Master	*Evaluation of ABS in Modeling Real World Safety-Critical Systems* by Chunyuan Yu, TUD, 2018
Master	*A Formal Model of a Railway Operating Procedure with Moving Blocks and Dynamic Speed Profile*, by Stefan Dillmann, TUD, 2017
Bachelor	*Introducing and Exploiting Extended Types for ABS* by Daniel Drodt, TUD, 2020
Bachelor	*Counterexample Generation for Formal Verification of ABS* by Nils Rollshausen, TUD, 2020, published in [4]
Bachelor	*Semi-Dynamic Session Types for ABS* by Anton Wolf Haubner, TUD, 2019, published in [26]
Bachelor	*Makroskopisches Editieren von prototypischen Eisenbahnbetriebsverfahren* by Jonas Stromberg, TUD, 2019, published in [42]
Bachelor	*Formalizing the Concurrency Model of AOs in a Linearization Framework* by Markus Bommer, TUD, 2019
Bachelor	*Concept Formation in Computer Science: Modeling and Programming* by Markus Kaltenpoth, TUD, 2019
Bachelor	*Hybrid Active Objects mit ABS* by Martina Kettenbach, TUD, 2018
Bachelor	*Active Object Languages for Railway Modeling* by Fabian Wagner, TUD, 2018
Study Thesis	*Practical Counterexample Generation and Lightweight Session Types for ABS* by Nils Rollshausen, TUD, 2021
Master (Examiner)	*Exploring Automatic Text Simplification of Requirements* by Eivind Grønli Guren, TUD, 2022
Master (Examiner)	*Commutativity Analysis in ABS* by Sondre Skaflem Lunde, TUD, 2021
Mentor	Erasmus+ trainee, Anton Wolf Haubner, 03.2022 – 05.2022
Mentor	Erasmus+ trainee, Nils Rolshausen, 01.2023 – 03.2023

Student Assistant	Robin Ferrari, 01.2019 – 10.2019 *Contributed to the data flow analysis in the ABS compiler*
Student Assistant	Jonas Stromberg, 05.2018 – 06.2019 *Contributed to visualization in the FormbaR project*
Student Assistant	Björn Petersen, 05.2017 – 11.2018 *Contributed to the functional layer in the ABS compiler*

Software

Main Author	SMOL `smolang.org`, actively maintained since 2021 *A language and interpreter combining knowledge graphs and object-orientation*
Main Author	Crowbar `github.com/edkamb/crowbar-tool`, actively maintained since 2020 *A deductive verification system for active objects*
Main Author	Hybrid ABS Compiler `formbar.raillab.de/habs`, actively maintained since 2018 *An extension of active objects with differential equations for dynamic behavior*
Contributor	ABS Compiler `abs-models.org`, since 2017 *Responsible for the variability layer*
Contributor	KeY Verification System `key-project.org`, 2014-2016 *Implemented an approach for loop invariant inference*

Language Skills

German(Native Speaker), **English**(C2), **Russian**(B2), **Norwegian**(A2)

References

- Prof. Reiner Hähnle, haehnle@cs.tu-darmstadt.de, (+49) 6151 16 21361, PhD Advisor
- Prof. Einar Broch Johnsen, einarj@ifi.uio.no, (+47) 2285 2509, Group Leader
- Prof. Ferruccio Damiani, damiani@unito.it, (+39) 011 6706719, Co-Author

Publications

In total, I have published 45 peer-reviewed articles: 40 in the proceedings of international conferences and workshops, and 5 journal articles.
Total number of citations in Google Scholar: 328. h-index: 9, i10-index: 9 (according to `https://scholar.google.de/citations?user=-GBTulYAAAAJ`, 13.10.2023).

Selected Conference Publications

[1] Eduard Kamburjan, Vidar Norstein Klungre, Rudolf Schlatte, Einar Broch Johnsen, and Martin Giese. *Programming and Debugging with Semantically Lifted States*. In *ESWC*, *LNCS* 12731, 2021.

[2] Eduard Kamburjan. *From Post-Conditions to Post-Region Invariants: Deductive Verification of Hybrid Objects*. In *HSCC*. ACM, 2021.

[3] Eduard Kamburjan. *Behavioral Program Logic*. In *TABLEAUX*, *LNCS* 11714, 2019.

Journal Articles

[4] Eduard Kamburjan, Marco Scaletta, and Nils Rollshausen. *Deductive Verification of Active Objects with Crowbar*. *Sci. Comput. Program.*, 226, 2023.

[5] Ferruccio Damiani, Reiner Hähnle, Eduard Kamburjan, Michael Lienhardt, and Luca Paolini. *Variability Modules*. *Journal of Systems and Software*, 195, 2023.

[6] Rudolf Schlatte, Einar Broch Johnsen, Eduard Kamburjan, and Silvia Lizeth Tapia Tarifa. *The ABS Simulator Toolchain*. *Sci. Comput. Program.*, 223, 2022.

[7] Eduard Kamburjan, Stefan Mitsch, and Reiner Hähnle. *A Hybrid Programming Language for Formal Modeling and Verification of Hybrid Systems*. *Leibniz Trans. Embed. Syst.*, 18(2), 2022.

[8] Eduard Kamburjan, Reiner Hähnle, and Sebastian Schön. *Formal modeling and analysis of railway operations with active objects*. *Sci. Comput. Program.*, 166, 2018.

Conference Publications

[9] Åsmund Aqissiaq Arild Kløvstad, Eduard Kamburjan, and Einar Broch Johnsen. *Compositional Correctness and Completeness for Symbolic Partial Order Reduction*. In *CONCUR*, *LIPIcs* 279, 2023.

[10] Prasad Talasila, Claudio Gomes, Peter Høgh Mikkelsen, Santiago Gil Arboleda, Eduard Kamburjan, and Peter Gorm Larsen. *Digital Twin as a Service (DTaaS): A Platform for Digital Twin Developers and Users*. In *IEEE Digital Twins*, 2023.

[11] Yuanwei Qu, Eduard Kamburjan, and Marting Giese. *A Geological Case Study on Semantically Triggered Processes*. In *ESWC*, *LNCS* 13998, 2023.

[12] Eduard Kamburjan and Michael Lienhardt. *Types and Verification for Delegated Control of Hybrid Objects*. In *Active Object Languages: Current Research Trends*, *LNCS* 14360, 2023.

[13] Eduard Kamburjan and Crystal Chang Din. *Runtime Enforcement Using Knowledge Bases*. In *FASE*, *LNCS* 13991, 2023.

[14] Reiner Hähnle, Eduard Kamburjan, and Marco Scaletta. *Context-aware Trace Contracts*. In *Active Object Languages: Current Research Trends*, *LNCS* 14360, 2023.

[15] Ferruccio Damiani, Eduard Kamburjan, Michael Lienhardt, and Luca Paolini. *Deltas for Functional Programs with Algebraic Data Types*. In *SPLC*. ACM, 2023.

[16] Eduard Kamburjan, Vidar Norstein Klungre, Rudolf Schlatte, S. Lizeth Tarifa Tapia, David Cameron, and Einar Broch Johnsen. *Digital Twin Reconguration Using Asset Models*. In *ISoLA*, *LNCS* 13704. Springer, 2022.

[17] Eduard Kamburjan, Vidar Norstein Klungre, and Martin Giese. *Never Mind the Semantic Gap: Modular, Lazy and Safe Loading of RDF Data*. In *ESWC*, *LNCS* 13261, 2022. **Best Paper Award**.

[18] Eduard Kamburjan and Einar Broch Johnsen. *Knowledge Structures over Simulation Units*. In *ANNSIM*. IEEE, 2022.

[19] Eduard Kamburjan and Sandro Rama Fiorini. *On the notion of naturalness in formal modeling*. In *The Logic of Software. A Tasting Menu of Formal Methods*, *LNCS* 13360, 2022.

[20] Eduard Kamburjan, Crystal Chang Din, Rudolf Schlatte, S. Lizeth Tarifa Tapia, and Einar Broch Johnsen. *Twinning-by-Construction: Ensuring Correctness for Self-Adaptive Digital Twins*. In *ISoLA*, *LNCS* 13701. Springer, 2022.

[21] Ole Jørgen Abusdal, Eduard Kamburjan, Violat Ka I Pun, and Volker Stolz. *A Notion of Equivalence for Refactorings with Abstract Execution*. In *ISoLA*, *LNCS* 13702. Springer, 2022.

[22] Rudolf Schlatte, Einar Broch Johnsen, Eduard Kamburjan, and S. Lizeth Tapia Tarifa. *Modeling and analyzing resource-sensitive actors: A tutorial introduction*. In *COORDINATION*, *LNCS* 12717, 2021.

[23] Ferruccio Damiani, Reiner Hähnle, Eduard Kamburjan, Michael Lienhardt, and Luca Paolini. *Variability Modules for Java-like Languages*. In *SPLC*. ACM, 2021.

[24] Eduard Kamburjan, Rudolf Schlatte, Einar Broch Johnsen, and S. Lizeth Tapia Tarifa. *Designing Distributed Control with Hybrid Active Objects*. In *ISoLA*, *LNCS* 12479, 2020.

[25] Eduard Kamburjan, Crystal Chang Din, Reiner Hähnle, and Einar Broch Johnsen. *Behavioral Contracts for Cooperative Scheduling*. In *20 Years of KeY*, *LNCS* 12345. 2020.

[26] Reiner Hähnle, Anton Haubner, and Eduard Kamburjan. *Locally Static, Globally Dynamic Session Types for Active Objects*. In *Recent Developments in the Design and Implementation of Programming Languages*, *OASIcs* 86. Schloss Dagstuhl, 2020.

[27] Dilian Gurov, Reiner Hähnle, and Eduard Kamburjan. *Who Carries the Burden of Modularity? - Introduction to ISoLA 2020 Track on Modularity and (De-)composition in Verification*. In *ISoLA*, *LNCS* 12476, 2020.

[28] Eduard Kamburjan, Crystal Chang Din, Reiner Hähnle, and Einar Broch Johnsen. *Asynchronous Cooperative Contracts for Cooperative Scheduling*. In *SEFM*, *LNCS* 11724, 2019.

[29] Eduard Kamburjan and Tzu-Chun Chen. *Stateful Behavioral Types for Active Objects*. In *IFM*, *LNCS* 11023, 2018.

[30] Ferruccio Damiani, Reiner Hähnle, Eduard Kamburjan, and Michael Lienhardt. *Same Same But Different: Interoperability of Software Product Line Variants*. In *Principled Software Development*. Springer, 2018.

[31] Ferruccio Damiani, Reiner Hähnle, Eduard Kamburjan, and Michael Lienhardt. *Interoperability of software product line variants*. In *SPLC*. ACM, 2018.

[32] Eduard Kamburjan and Reiner Hähnle. *Deductive Verification of Railway Operations*. In *RSSRail*, *LNCS* 10598, 2017.

[33] Ferruccio Damiani, Reiner Hähnle, Eduard Kamburjan, and Michael Lienhardt. *A Unified and Formal Programming Model for Deltas and Traits*. In *FASE*, *LNCS* 10202, 2017.

[34] Eduard Kamburjan, Crystal Chang Din, and Tzu-Chun Chen. *Session-Based Compositional Analysis for Actor-Based Languages Using Futures*. In *ICFEM*, *LNCS* 10009, 2016.

[35] Quoc Huy Do, Eduard Kamburjan, and Nathan Wasser. *Towards Fully Automatic Logic-Based Information Flow Analysis: An Electronic-Voting Case Study*. In *POST*, *LNCS* 9635, 2016.

Workshop Publications

[36] Eduard Kamburjan, Vidar Norstein Klungre, S. Lizeth Tapia Tarifa, Rudolf Schlatte, Martin Giese, David Cameron, and Einar Broch Johnsen. *Emerging Challenges in Compositionality and Correctness for Digital Twins*. In *FMDT*, 2023.

[37] Johan Cederbladh, Loek Cleophas, Eduard Kamburjan, Lucas Lima, and Hans Vangheluwe. *Symbolic Reasoning for Early Decision-Making in Model-Based Systems Engineering*. In *MBSE@Models*. IEEE, 2023. accepted for publication.

[38] Eduard Kamburjan and Nathan Wasser. *The Right Kind of Non-Determinism: Using Concurrency to Verify C Programs with Underspecified Semantics*. In *ICE*, 365. EPTCS, 2022.

[39] Eduard Kamburjan and Egor V. Kostylev. *Type Checking Semantically Lifted Programs via Query Containment under Entailment Regimes*. In *Description Logic Workshop*, *CEUR Workshop Proceedings* 2954. CEUR-WS.org, 2021.

[40] Eduard Kamburjan and Lukas Grätz. *Increasing Engagement with Interactive Visualization: Formal Methods as Serious Games*. In *Formal Methods Teaching*, *LNCS* 13122, 2021.

[41] Eduard Kamburjan and Jieying Chen. *Optimizing Semantically Lifted Programs through Ontology Modularity*. In *NWPT*. Reykjavik University Technical Reports, 2021.

[42] Eduard Kamburjan and Jonas Stromberg. *Tool Support for Validation of Formal System Models: Interactive Visualization and Requirements Traceability*. In *F-IDE@FM*, *EPTCS* 310, 2019.

[43] Eduard Kamburjan and Reiner Hähnle. *Prototyping Formal System Models with Active Objects*. In *ICE*, *EPTCS* 279, 2018.

[44] Eduard Kamburjan. *Detecting Deadlocks in Formal System Models with Condition Synchronization*. *AVoCS, Electron. Commun. Eur. Assoc. Softw. Sci. Technol.*, 76, 2018.

[45] Eduard Kamburjan and Reiner Hähnle. *Uniform Modeling of Railway Operations*. In *FTSCS*, *CCIS* 694, 2016.

PhD Thesis

[46] Eduard Kamburjan. *Modular Verification of a Modular Specification: Behavioral Types as Program Logics*. PhD thesis, Technical University of Darmstadt, 2020.

Editorship

[47] Stefan Hallerstede and Eduard Kamburjan, editors. *Proceedings of the Workshop on Applications of Formal Methods and Digital Twins, co-located with 25th International Symposium on Formal Methods (FM 2023), Lübeck, Germany, March 06, 2023.*, *CEUR Workshop Proceedings* 3507. CEUR-WS.org, 2023.

[48] Frank de Boer, Ferruccio Damiani, Reiner Hähnle, Einar Broch Johnsen, and Eduard Kamburjan, editors. *Active Object Languages: Current Research Trends*, *LNCS* 14360. Springer, 2023. in production.

[49] Marie-Christine Jakobs and Eduard Kamburjan. *Special issue on Fundamental Approaches to Software Engineering (FASE 2022) . Sci. Comput. Program.*, 228, 2023. in production.

Preprints

[50] Eduard Kamburjan. Modular analysis of distributed hybrid systems using post-regions (full version), 2023. https://doi.org/10.48550/arXiv.2309.10470.

Other Publications

[51] Reiner Hähnle, Eduard Kamburjan, and Marco Scaletta. *Herding CATs (invited paper)*. In *SEFM'23*, *LNCS* 14323, 2023.

[52] Sebastian Schön, Eduard Kamburjan, and Reiner Hähnle. *The Future Use Cases of Formal Methods in Railways*. In *Scientific Railway Signalling Symposium*. TU Darmstadt, 2018.