

Model-Driven Software Development in Technical Spaces (MOST) - Heterogeneous Software Factories -

Announcements

Prof. Dr. Uwe Aßmann

Technische Universität Dresden

Institut für Software- und
Multimediatechnik

Lehrstuhl Softwaretechnologie

[http://st.inf.tu-dresden.de/teaching/
most](http://st.inf.tu-dresden.de/teaching/most)

WS 21/22-0.2, 11/20/21



DRESDEN
concept
Exzellenz aus
Wissenschaft
und Kultur

Relation of the Course to Modules

The course can be used for the following modules:

- ▶ Diplom Informatik: INF-BAS3, INF-VERT3
- ▶ Master Informatik: INF-BAS3, INF-VERT3
- ▶ Bachelor Medieninformatik: INF-B-540
- ▶ Master Medieninformatik: INF-BI-4, INF-BI-5
- ▶ Diplom IST: IST-B-321
- ▶ Diplom Informatik (2004): INF-04-FG-SWT
- ▶ Diplom Informatik (2004): FG 4 SE
- ▶ Master Distributed Systems Engineering (DSE)
- ▶ Master Computational Modeling and Simulation (CMS)

For other programs, special rules may hold; consult the manuals of your "Nebenfach".

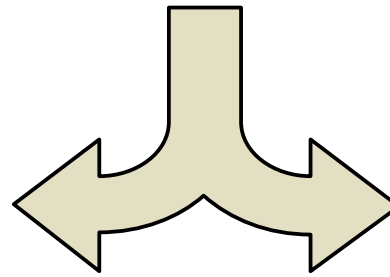
Master's Courses (Hauptstudium)

3

Model-Driven Software Develop

Softwaretechnologie II (Bachelor)

Modellierung, Entwurfsmethoden,
Produktlinien, Geschäftsmodelle (WS)



**Model-Driven Software
Development in Technical Spaces**
How to be productive in software
development (WS)

**Requirements Engineering
und Testen (Dr. Demuth)**
Wie man Qualität für Software
erzielt (WS)

**Ausgewählte Kapitel aus
der Softwaretechnik
(Dr. Götz)**
Softwarearchitektur (SS)

**Future-Proof Software
Systems (Prof. Furrer)**
Evolvable architectures (WS)

**Design Patterns and
Frameworks**
Architektur objektorientierter
Systeme (WS)

**Component-Based
Software Engineering**
Product lines with component
models (SS)

**Automotive Software
Engineering
(Dr. Conrad) (SS)**

**Academic Skills in Software
Engineering**
How to work scientifically (SS)
(Dr. Götz)

Software-Management
Wie man Projekte macht (SS)
(Dr. Demuth)

**Softwareentwicklung in der
industriellen Praxis**
Ringvorlesung mit
Industriedozenten

**Software as a Business
(WS)**
How to develop a business model
and a startup

Lecturing – How in Corona-Times?

- ▶ We will start with presence teaching and film a video

- ▶ If Aßmann catches a cold, there might be sudden changes
 - Then, lecture films from home
 - Lecture dates will morph to question hours
 - People may ask questions via a video conference room
 - <https://www.gotomeet.me/UweAssmann>

Central Topics of the Course

5

Model-Driven Software Development in Technical Spaces (MOST)

6. Advanced Topics in Modeling

**Melanie
Deep Modeling**

5. Integration of Tools and Macromodels in Multi-TS Software Factories

**Transformation Bridge
Adapter Bridge**

4. Technical Space GraphWare and its languages

**Datalog
Graph Transformation**

3. Macromodels as the core of homogeneous, One-TS Software Factories

MDA OSM ROSI

2. Technical Space TreeWare and its languages

**Tree Grammars
Ref. Attr. Grammars**

1. Classic Metamodeling
Metamodels structure languages, data, models and programs
Tool Construction and -Generation in a Technical Space

EMFText eMOFlon

0. Basics:
Applications of MDSD: Design tools of complex systems, Design tools for CPS

**EMOF MOF EMOF
CROM**

MDSD and Heterogeneous Software Factories

- ▶ MDSD is the engineering of applications with *several related models* (inclusively code), based on systematic engineering of metamodels (languages) in technical spaces.
- ▶ MDSD therefore treats *heterogeneous models*
 - Treating different aspects of a system
 - Specified in different modeling languages
 - Living in different *technical spaces*
- ▶ How are models related?
 - Model cover by matching rules
 - Model mappings
 - Model transformations
- ▶ How are the languages related?
 - By mapping
 - By role fattening
- ▶ How are the technical spaces related?
 - By macromodelling

Engineering of Technical Spaces, Macromodels, and Software Factories

- ▶ Def.: A **multimodel** is a set of correlated models.
- ▶ Def.: A **macromodel** is a set of *systematically* related, heterogeneous models.
 - Macromodelling treats heterogeneity
 - Requires model integration techniques
- ▶ Engineering of Technical Spaces and macromodels is one of the most important topics of the future of software and systems development
- ▶ Dresden has modern technologies and tools based on macromodels
 - Role-and context-based languages (CROM, SCROLL)
 - Invasive composition (SkAT)
 - Metacomposition tools (Reuseware, SkAT, Style Sheets)
 - Round-Trip Engineering and Role-based tools

Part 0: Basics

- ▶ 1. Modeling in MOST (MDSD)
- ▶ 2. MDSD Applications
- ▶ 3. Context and Roles in Modelling

Part I: Classic Metamodeling

- ▶ 10. Metamodeling in the (E)MOF technical space
 - The Technical Space House
- ▶ 11. Bridging the TS Grammarware and EMOF with EMFText
- ▶ 12. Technical Spaces and Software Factories
- ▶ 13. Structure of M2

Part II: Treating heterogeneity in TreeWare

- ▶ 20. Grammarware: Parser generators, Text algebrae
- ▶ 21. Treeware: Query
- ▶ 22. Deep Analysis in Treeware
 - Abstract interpretation
- ▶ 23. Link-Treeware
 - XML, JSON, Xcerpt

Part III: Macromodels in 1-TS Software Factories

- ▶ 30. Requirements, Documentation and Test Management in a macromodel
- ▶ 31. Model-Driven Architecture
- ▶ 32. Single Underlying Model (SUM) and Skeleton-SUMs
- ▶ 33. Stream-based integration

Outline Part IV-VII – Architecture of Software Factories

Part IV Treating heterogeneity in GraphWare

- ▶ 40 Flat Analysis in Graph- and Modelware
 - Semmler .QL, TGreQL
- ▶ 41 Deep Analysis in Graphware
 - Reachability
- ▶ 42 Graph and Model Transformations
- ▶ 43. MetaCASE tools (MetaEdit+)
- ▶ 44. MOFLON as example
- ▶ 45. Smart applications (SMAGs)
- ▶ 46. Fujaba

Part V: Architecture and Composition of Tools to Applications

Tool integration, Exchange formats, Bridges between technical spaces, Composition of stream-based tools

- ▶ 50. Tools-and-Materials Pattern Language
- ▶ 51. Role-based composition of languages for the composition of materials
- ▶ And tools
- ▶ 52. Role-based repository architecture
- ▶ 53. Material-exchange and Stream-Based integration
- ▶ 54. Synchronization and Round-Trip Engineering with Triple Graph Grammars

Part VI: Advanced Modeling

- ▶ 60. Orthographic Software Modeling
- ▶ 61. 2-d-Software Modeling

Part VII: Multi-TS Software Factories

- ▶ 70. Large Software Factories

- ▶ [GPHS] Cesar Gonzalez-Perez and Brian Henderson-Sellers. Metamodelling for software engineering. Wiley, 2008.
- ▶ [Züll] Züllighoven, Heinz. Object-Oriented Construction Handbook; dpunkt.verlag 2005
- ▶ [B93] Balzert, H. (Hrg.) u.a.: CASE - Systeme und Werkzeuge; BI-Wissenschaftsverlag Mannheim, 5. vollst. überarb. Auflage 1993
- ▶ [BAL, BOO] Balzert, H. : Lehrbuch der Software-Technik (Bd. I): Software-Entwicklung; Spektrum Akademischer Verlag (3. Auflage) Heidelberg 2009
- ▶ [BAL-II B98] Balzert, H. : Lehrbuch der Software-Technik (Bd. II): Software-Management. Spektrum Akademischer Verlag Heidelberg 1998
- ▶ [HMF] Hesse, W., Merbeth, G., Fröhlich, R.: Software-Entwicklung / Vorgehensmodelle. Projektführung, Produktverwaltung; Oldenbourg Verlag München 1992
- ▶ [SN92] Schönthaler, F., Nemeth, T.: Software-Entwicklungswerkzeuge: Methodische Grundlagen; B.G. Teubner Verlag Stuttgart 1992
- ▶ [ES89] Engels, G., Schäfer, W.: Programmentwicklungsumgebungen - Konzepte und Realisierung; B.G. Teubner Verlag Stuttgart 1989
- ▶ [ZK04] Zeller, A., Krinke, J.: Open-Source-Programmierwerkzeuge (2. Auflage) dpunkt.verlag Heidelberg 2004

- ▶ [8 F91] Fisher, A.S.: CASE - Using Software Development Tools (Second Edition); John Wiley & Sons 1991
- ▶ [9 B92] Bauknecht, K. (Hrg.): Informatik-Anwendungsentwicklung - Praxiserfahrungen mit CASE; B.G.Teubner Verlag Stuttgart 1992
- ▶ [Raasch] Raasch. Systementwicklung mit strukturierten Methoden. Hanser. 1993
- ▶ Stachowiak, Herbert. Allgemeine Modelltheorie. Springer, Wien, 1973

- ▶ [TLS] Ivan Kurtev, Jean Bezivin, and Mehmet Aksit. Technological Spaces: An Initial Appraisal. In Proceedings of the Confederated International Conferences CoopIS, DOA, and ODBASE 2002, 2002.
- ▶ [TS] Jean Bezivin and Ivan Kurtev. Model-based Technology Integration with the Technical Space Concept. In Proceedings of the Metainformatics Symposium, Berlin; Heidelberg, 2005. Springer.
- ▶ [HesseMayr] Wolfgang Hesse and Heinrich C. Mayr. Modellierung in der Softwaretechnik: eine Bestandsaufnahme. Informatik Spektrum, 31(5):377-393, 2008.
- ▶ Ed Seidewitz. What models mean. IEEE Software, 20:26-32, September 2003.
 - http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1231147&tag=1
- ▶ Jean Bézivin. Model Driven Engineering: An Emerging Technical Space. In R. Lämmel, J. Saraiva, and J. Visser (Eds.): GTTSE 2005, LNCS 4143, pp. 36 – 64, 2006. Springer.
- ▶ Wolfgang Hesse. More matters on (meta-)modelling: remarks on Thomas Kühne's 'matters'. Software and System Modeling, 5(4):387-394, 2006.

ST Works on Metamodelling in the Last Years

Henrik Lochmann. HybridMDSD: Multi-Domain Engineering with Model-Driven Software Development using Ontological Foundations. PhD thesis, Technische Universität Dresden, Fakultät Informatik, 2009,

<http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-27380>

Mirko Seifert. Designing Round-Trip Systems by Model Partitioning and Change Propagation. PhD thesis, Technische Universität Dresden, Fakultät Informatik, June 2011, <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-71098>

Konrad Voigt. Structural Graph-based Metamodel Matching. PhD thesis, Technische Universität Dresden, Fakultät Informatik, November 2011,

<http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-81671>

Christian Wende. Language Family Engineering. PhD thesis, Technische Universität Dresden, Fakultät Informatik, March 2012,

<http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-88985>

Max Leuthäuser. A Pure Embedding of Roles - Exploring 4-dimensional Dispatch for Roles in Structured Contexts. PhD thesis, Technische Universität Dresden, August 2017. <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-227624>

Thomas Kühn. A Family of Role-Based Languages. PhD thesis, Technische Universität Dresden, March 2017. <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-228027>



TECHNISCHE
UNIVERSITÄT
DRESDEN

Fakultät Informatik - Institut Software- und Multimediatechnik - Softwaretechnologie

Model-Driven Software Development in Technical Spaces (MOST) - Heterogeneous Software Factories -

Announcements

Prof. Dr. Uwe Aßmann
Technische Universität Dresden
Institut für Software- und
Multimediatechnik
Lehrstuhl Softwaretechnologie
[http://st.inf.tu-dresden.de/teaching/
most](http://st.inf.tu-dresden.de/teaching/most)
WS 21/22-0.2, 11/20/21



DRESDEN
concept
Exzellenz aus
Wissenschaft
und Kultur

Prof. Aßmann - Model-Driven Software Development (MDS)

Welcome to the MOST course!

Relation of the Course to Modules

The course can be used for the following modules:

- ▶ Diplom Informatik: INF-BAS3, INF-VERT3
- ▶ Master Informatik: INF-BAS3, INF-VERT3
- ▶ Bachelor Medieninformatik: INF-B-540
- ▶ Master Medieninformatik: INF-BI-4, INF-BI-5
- ▶ Diplom IST: IST-B-321
- ▶ Diplom Informatik (2004): INF-04-FG-SWT
- ▶ Diplom Informatik (2004): FG 4 SE
- ▶ Master Distributed Systems Engineering (DSE)
- ▶ Master Computational Modeling and Simulation (CMS)

For other programs, special rules may hold; consult the manuals of your "Nebenfach".



Master's Courses (Hauptstudium)

3 Model-Driven Software Develop

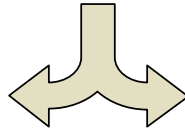
Softwaretechnologie II (Bachelor)
Modellierung, Entwurfsmethoden,
Produktlinien, Geschäftsmodelle (WS)

Model-Driven Software Development in Technical Spaces
How to be productive in software development (WS)

Requirements Engineering und Testen (Dr. Demuth)
Wie man Qualität für Software erzielt (WS)

Ausgewählte Kapitel aus der Softwaretechnik (Dr. Götz)
Softwarearchitektur (SS)

Future-Proof Software Systems (Prof. Furrer)
Evolvable architectures (WS)



Design Patterns and Frameworks
Architektur objektorientierter Systeme (WS)

Component-Based Software Engineering
Product lines with component models (SS)

Automotive Software Engineering (Dr. Conrad) (SS)

Academic Skills in Software Engineering
How to work scientifically (SS)
(Dr. Götz)

Software-Management
Wie man Projekte macht (SS)
(Dr. Demuth)

Softwareentwicklung in der industriellen Praxis
Ringvorlesung mit Industriezentren

Software as a Business (WS)
How to develop a business model and a startup

Lecturing – How in Corona-Times?

- ▶ We will start with presence teaching and film a video

- ▶ If Aßmann catches a cold, there might be sudden changes
 - Then, lecture films from home
 - Lecture dates will morph to question hours
 - People may ask questions via a video conference room
 - <https://www.gotomeet.me/UweAssmann>



Central Topics of the Course

6. Advanced Topics in Modeling	Melanie Deep Modeling
5. Integration of Tools and Macromodels in Multi-TS Software Factories	Transformation Bridge Adapter Bridge
4. Technical Space GraphWare and its languages	Datalog Graph Transformation
3. Macromodels as the core of homogeneous, One-TS Software Factories	MDA OSM ROSI
2. Technical Space TreeWare and its languages	Tree Grammars Ref. Attr. Grammars
1. Classic Metamodeling Metamodels structure languages, data, models and programs Tool Construction and -Generation in a Technical Space	EMFText eMOFlon
0. Basics: Applications of MDS: Design tools of complex systems, Design tools for CPS	EMOF MOF EMOF CROM

- ▶ MDSO is the engineering of applications with *several related models* (inclusively code), based on systematic engineering of metamodels (languages) in technical spaces.
- ▶ MDSO therefore treats *heterogeneous models*
 - Treating different aspects of a system
 - Specified in different modeling languages
 - Living in different *technical spaces*
- ▶ How are models related?
 - Model cover by matching rules
 - Model mappings
 - Model transformations
- ▶ How are the languages related?
 - By mapping
 - By role fattening
- ▶ How are the technical spaces related?
 - By macromodelling

Engineering of Technical Spaces, Macromodels, and Software Factories

- ▶ Def.: A **multimodel** is a set of correlated models.
- ▶ Def.: A **macromodel** is a set of *systematically* related, heterogeneous models.
 - Macromodelling treats heterogeneity
 - Requires model integration techniques
- ▶ Engineering of Technical Spaces and macromodels is one of the most important topics of the future of software and systems development
- ▶ Dresden has modern technologies and tools based on macromodels
 - Role-and context-based languages (CROM, SCROLL)
 - Invasive composition (SkAT)
 - Metacomposition tools (Reuseware, SkAT, Style Sheets)
 - Round-Trip Engineering and Role-based tools

Part 0: Basics

- ▶ 1. Modeling in MOST (MDSD)
- ▶ 2. MDSD Applications
- ▶ 3. Context and Roles in Modelling

Part I: Classic Metamodeling

- ▶ 10. Metamodeling in the (E)MOF technical space
 - The Technical Space House
- ▶ 11. Bridging the TS Grammarware and EMOF with EMFText
- ▶ 12. Technical Spaces and Software Factories
- ▶ 13. Structure of M2

Part II: Treating heterogeneity in TreeWare

- ▶ 20. Grammarware: Parser generators, Text algebrae
- ▶ 21. Treeware: Query
- ▶ 22. Deep Analysis in Treeware
 - Abstract interpretation
- ▶ 23. Link-Treeware
 - XML, JSON, Xcerpt

Part III: Macromodels in 1-TS Software Factories

- ▶ 30. Requirements, Documentation and Test Management in a macromodel
- ▶ 31. Model-Driven Architecture
- ▶ 32. Single Underlying Model (SUM) and Skeleton-SUMs
- ▶ 33. Stream-based integration

Outline Part IV-VII – Architecture of Software Factories

9

Model-Driven Software Development in Technical Spaces (MOST)

Part IV Treating heterogeneity in GraphWare

- ▶ 40 Flat Analysis in Graph- and Modelware
 - Semmlé .QL, TGreQL
- ▶ 41 Deep Analysis in Graphware
 - Reachability
- ▶ 42 Graph and Model Transformations
- ▶ 43. MetaCASE tools (MetaEdit+)
- ▶ 44. MOFLON as example
- ▶ 45. Smart applications (SMAGs)
- ▶ 46. Fujaba

Part V: Architecture and Composition of Tools to Applications

Tool integration, Exchange formats, Bridges between technical spaces, Composition of stream-based tools

- ▶ 50. Tools-and-Materials Pattern Language
- ▶ 51. Role-based composition of languages for the composition of materials
- ▶ And tools
- ▶ 52. Role-based repository architecture
- ▶ 53. Material-exchange and Stream-Based integration
- ▶ 54. Synchronization and Round-Trip Engineering with Triple Graph Grammars

Part VI: Advanced Modeling

- ▶ 60. Orthographic Software Modeling
- ▶ 61. 2-d-Software Modeling

Part VII: Multi-TS Software Factories

- ▶ 70. Large Software Factories

- ▶ [GPHS] Cesar Gonzalez-Perez and Brian Henderson-Sellers. Metamodelling for software engineering. Wiley, 2008.
- ▶ [Züll] Züllighoven, Heinz. Object-Oriented Construction Handbook; dpunkt.verlag 2005
- ▶ [B93] Balzert, H. (Hrg.) u.a.: CASE - Systeme und Werkzeuge; BI-Wissenschaftsverlag Mannheim, 5. vollst. überarb. Auflage 1993
- ▶ [BAL, BOO] Balzert, H. : Lehrbuch der Software-Technik (Bd. I): Software-Entwicklung; Spektrum Akademischer Verlag (3. Auflage) Heidelberg 2009
- ▶ [BAL-II B98] Balzert, H. : Lehrbuch der Software-Technik (Bd. II): Software-Management. Spektrum Akademischer Verlag Heidelberg 1998
- ▶ [HMF] Hesse, W., Merbeth, G., Fröhlich, R.: Software-Entwicklung / Vorgehensmodelle. Projektführung, Produktverwaltung; Oldenbourg Verlag München 1992
- ▶ [SN92] Schönthaler, F., Nemeth, T.: Software-Entwicklungswerkzeuge: Methodische Grundlagen; B.G. Teubner Verlag Stuttgart 1992
- ▶ [ES89] Engels, G., Schäfer, W.: Programmentwicklungsumgebungen - Konzepte und Realisierung; B.G. Teubner Verlag Stuttgart 1989
- ▶ [ZK04] Zeller, A., Krinke, J.: Open-Source-Programmierwerkzeuge (2. Auflage) dpunkt.verlag Heidelberg 2004

- ▶ [8 F91] Fisher, A.S.: CASE - Using Software Development Tools (Second Edition); John Wiley & Sons 1991
- ▶ [9 B92] Bauknecht, K. (Hrg.): Informatik-Anwendungsentwicklung - Praxiserfahrungen mit CASE; B.G.Teubner Verlag Stuttgart 1992
- ▶ [Raasch] Raasch. Systementwicklung mit strukturierten Methoden. Hanser. 1993
- ▶ Stachowiak, Herbert. Allgemeine Modelltheorie. Springer, Wien, 1973

Obligatory Papers

- ▶ [TLS] Ivan Kurtev, Jean Bezivin, and Mehmet Aksit. Technological Spaces: An Initial Appraisal. In Proceedings of the Confederated International Conferences CoopIS, DOA, and ODBASE 2002, 2002.
- ▶ [TS] Jean Bezivin and Ivan Kurtev. Model-based Technology Integration with the Technical Space Concept. In Proceedings of the Metainformatics Symposium, Berlin; Heidelberg, 2005. Springer.
- ▶ [HesseMayr] Wolfgang Hesse and Heinrich C. Mayr. Modellierung in der Softwaretechnik: eine Bestandsaufnahme. Informatik Spektrum, 31(5):377-393, 2008.
- ▶ Ed Seidewitz. What models mean. IEEE Software, 20:26-32, September 2003.
 - http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=1231147&tag=1
- ▶ Jean Bézivin. Model Driven Engineering: An Emerging Technical Space. In R. Lämmel, J. Saraiva, and J. Visser (Eds.): GTTSE 2005, LNCS 4143, pp. 36 – 64, 2006. Springer.
- ▶ Wolfgang Hesse. More matters on (meta-)modelling: remarks on Thomas Kühne's 'matters'. Software and System Modeling, 5(4):387-394, 2006.



Henrik Lochmann. HybridMDSD: Multi-Domain Engineering with Model-Driven Software Development using Ontological Foundations. PhD thesis, Technische Universität Dresden, Fakultät Informatik, 2009, <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-27380>

Mirko Seifert. Designing Round-Trip Systems by Model Partitioning and Change Propagation. PhD thesis, Technische Universität Dresden, Fakultät Informatik, June 2011, <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-71098>

Konrad Voigt. Structural Graph-based Metamodel Matching. PhD thesis, Technische Universität Dresden, Fakultät Informatik, November 2011, <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-81671>

Christian Wende. Language Family Engineering. PhD thesis, Technische Universität Dresden, Fakultät Informatik, March 2012, <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-88985>

Max Leuthäuser. A Pure Embedding of Roles - Exploring 4-dimensional Dispatch for Roles in Structured Contexts. PhD thesis, Technische Universität Dresden, August 2017. <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-227624>

Thomas Kühn. A Family of Role-Based Languages. PhD thesis, Technische Universität Dresden, March 2017. <http://nbn-resolving.de/urn:nbn:de:bsz:14-qucosa-228027>