

Kurs OMSI im WiSe 2013/14

Objektorientierte Simulation mit ODEMx

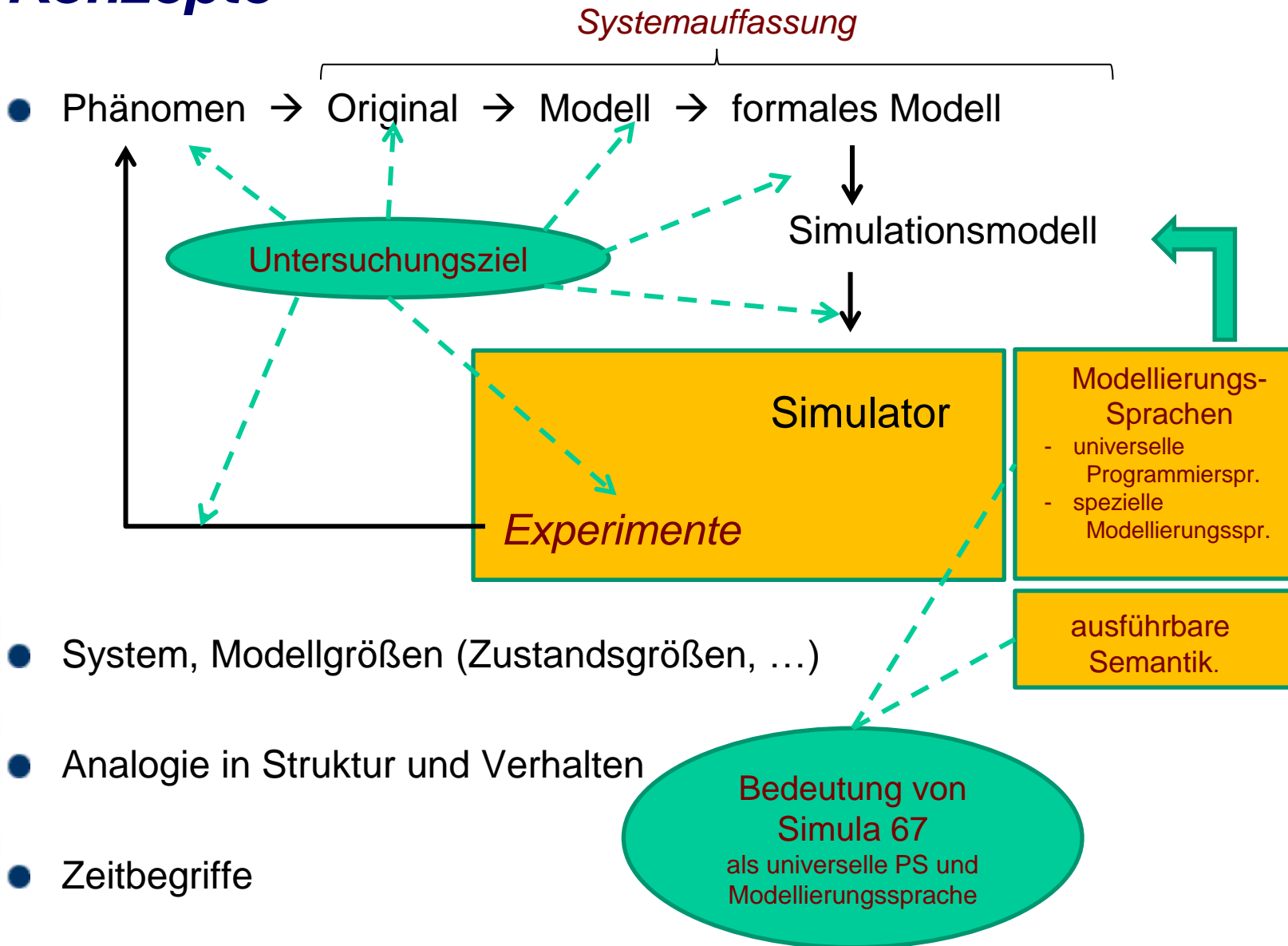
Prof. Dr. Joachim Fischer
Dr. Klaus Ahrens
Dipl.-Inf. Ingmar Eveslage

fischer|ahrens|eveslage@informatik.hu-berlin.de

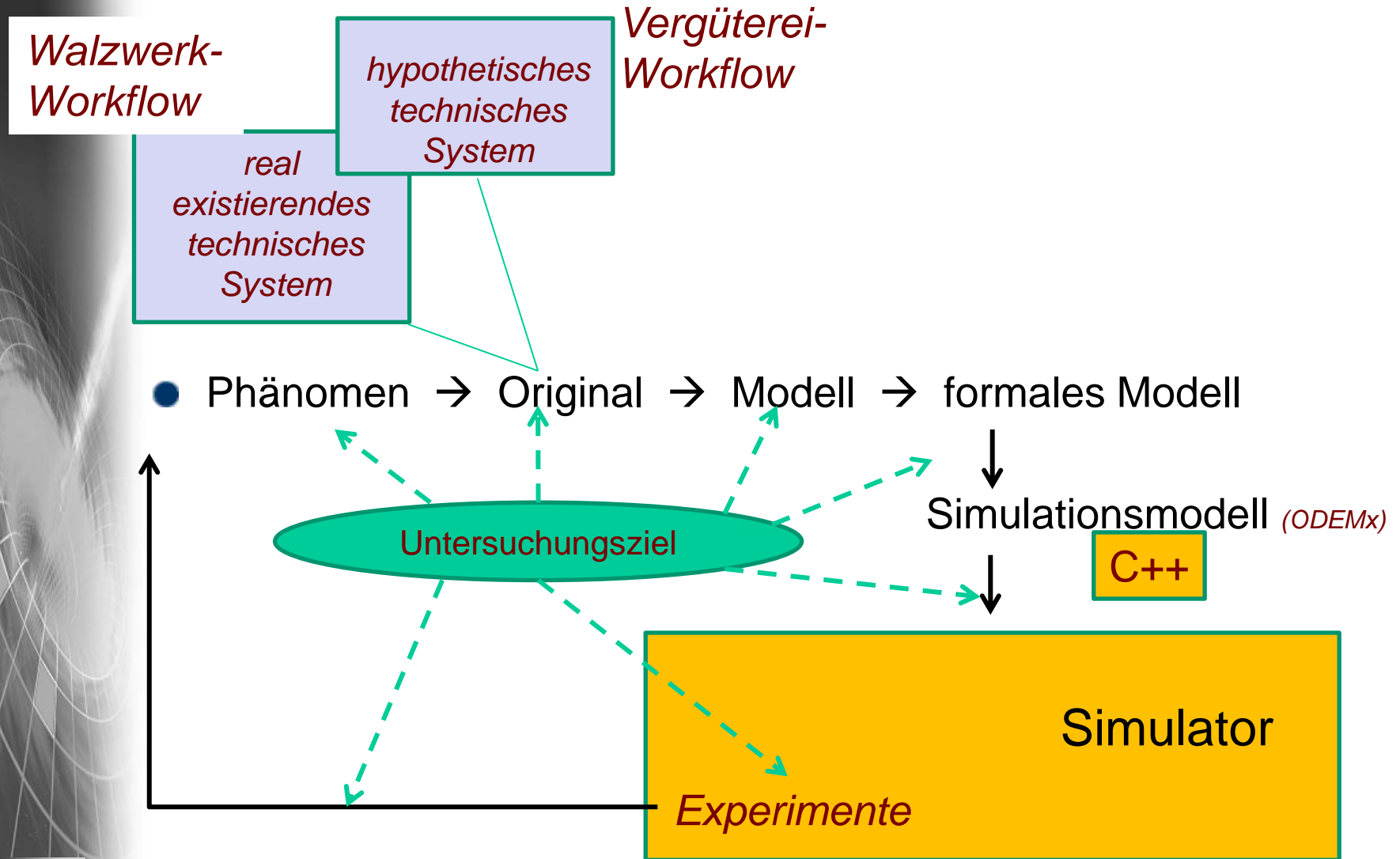
Letzte Vorlesung

1. Systemsimulation – was ist das?
2. Ein Blick zurück in die Anfänge
3. Modelle und Originale
4. Modellierungssprachen, Simulationsumgebungen
5. Beispiele aus der aktuellen Forschung
6. Paradigma der objektorientierten Modellierung
7. Klassifikation dynamischer Systeme
8. M&S eines Niedertemperaturofens

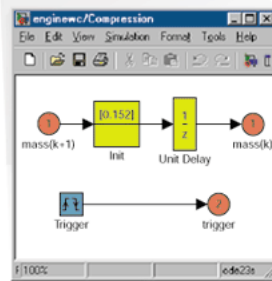
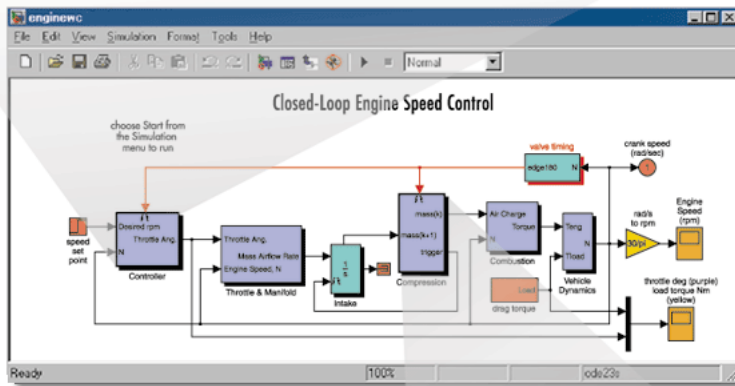
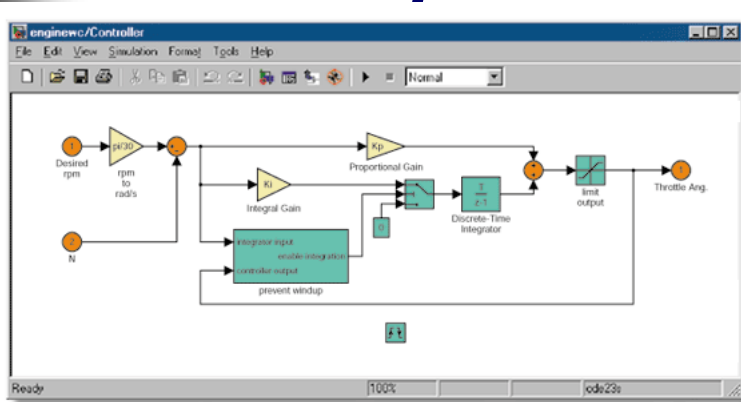
Konzepte



Modellierung: Universelle Programmiersprache

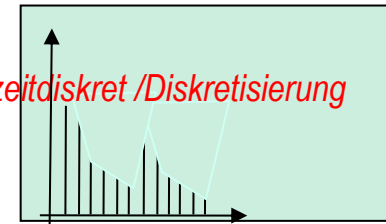
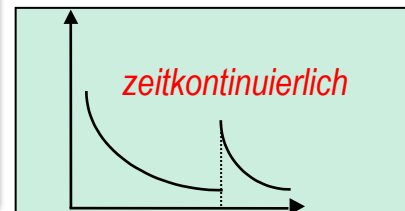


Beispiele domänenspezifische Modellierungssprachen



Simulink

- hierarchische graphische Modellierung
- kontinuierliche u. diskreter Schaltblöcke
- S-Functions: eigener Code u. MATLAB
- für einzelne Domänen (wie **mechanische**, **elektrische** oder **hydraulische** Systeme) stehen spezielle Zusätze zur Verfügung, welche die Modellierung von physikalischen Systemen zusätzlich vereinfachen

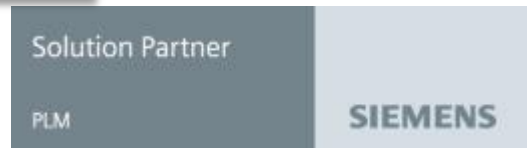


Beispiele domänenspezifische Modellierungssprachen



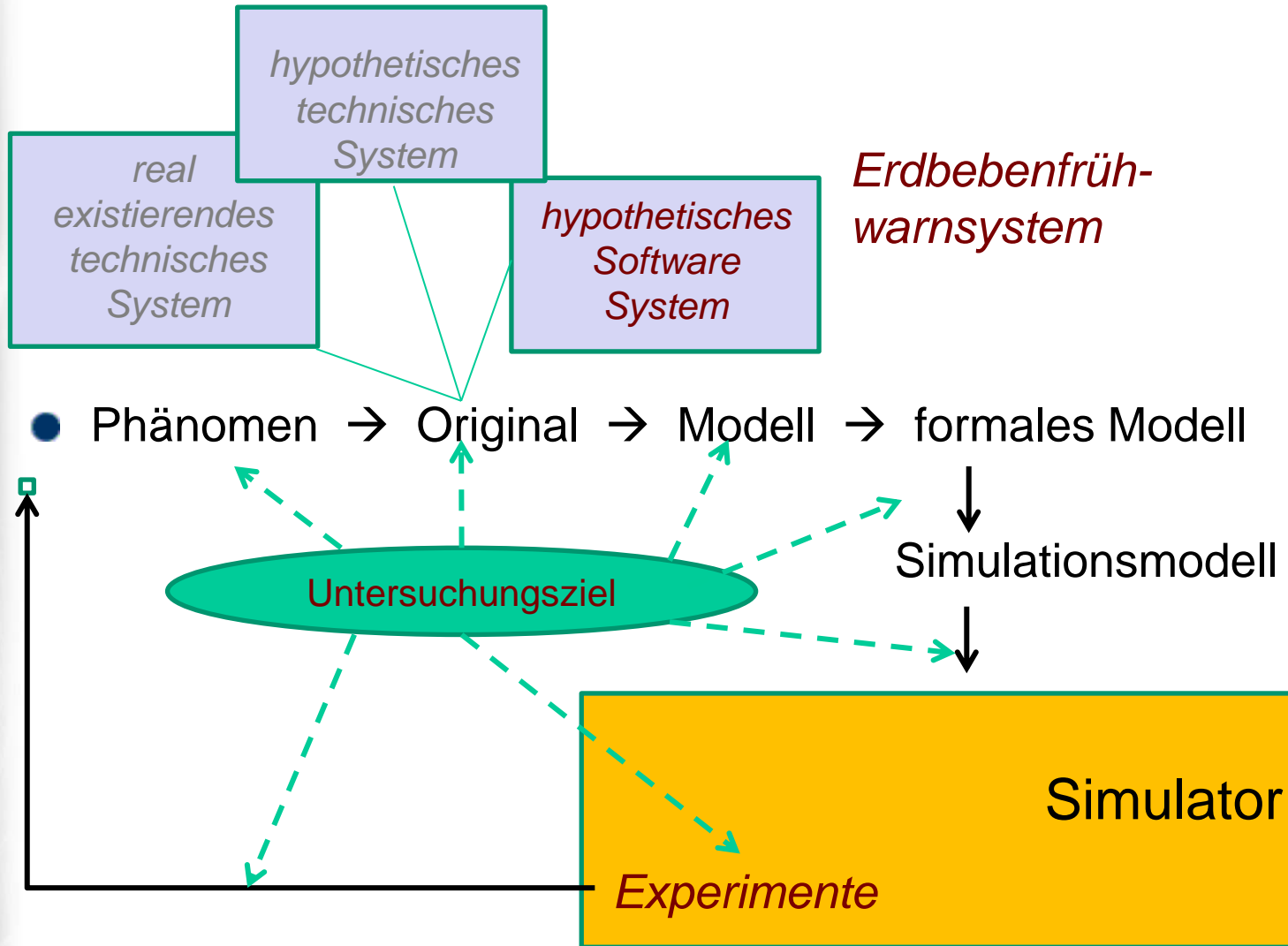
Plant Simulation

- graphische Modellierung, Simulation, Visualisierung
- Optimierung von Logistik- und Geschäftsprozessen

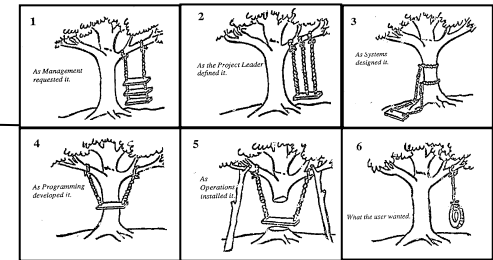


Allg. Bedeutung von Modellierung und Simulation unstrittig, aber:
Wie ist der Stand des Modelleinsatzes bei der Software-Produktion?

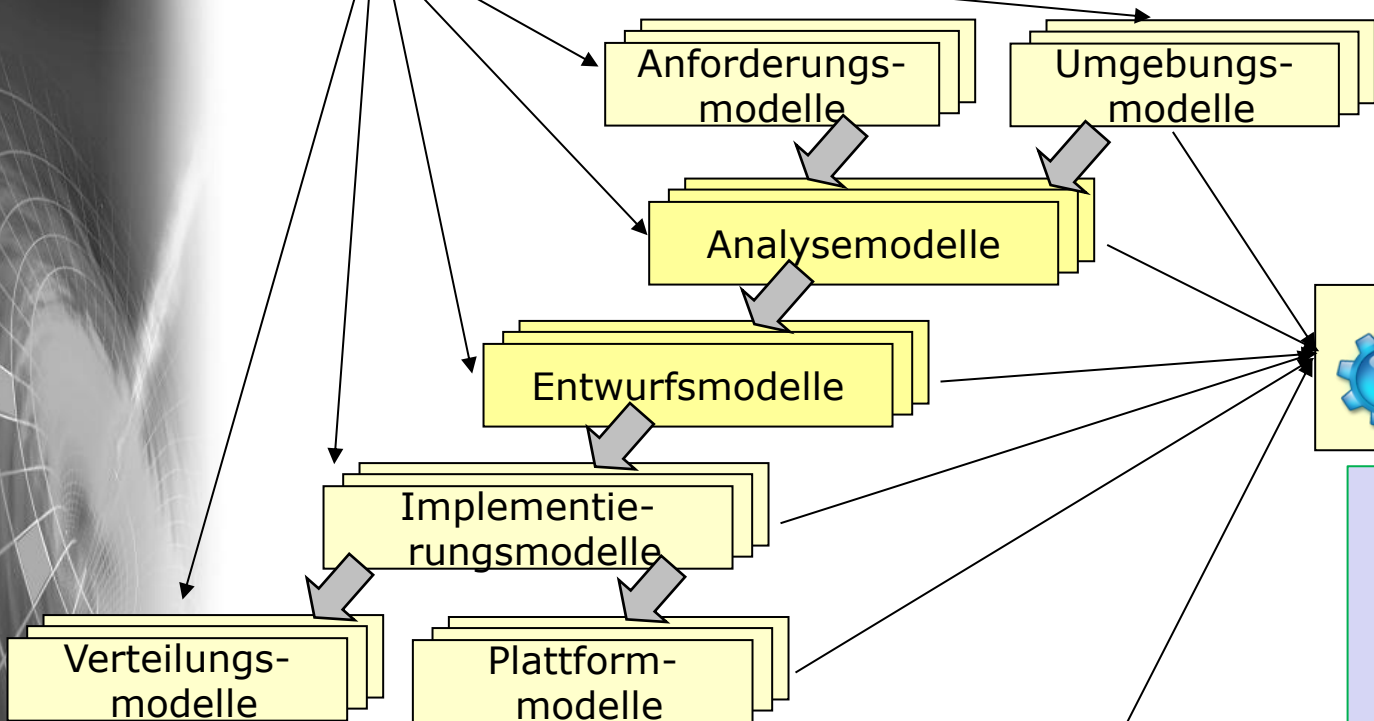
... am Beispiel



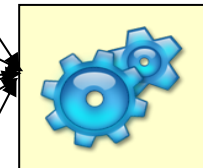
Alte Idee: Modellbasierte Software-Entwicklung verteilter Systeme (vereinfacht)



Anforderungen



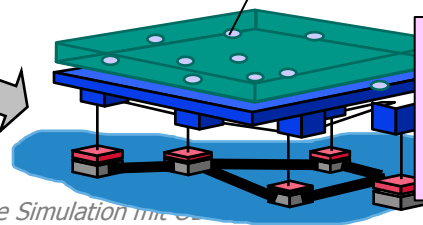
Modell-Checker
Simulator
Tester



ursprünglich:
Wasserfallmethode

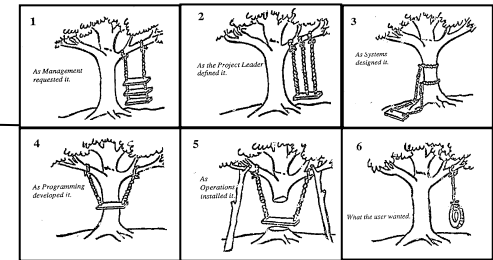
später verbessert:
iterativ, inkrementell

manuelle
Modelltransformation

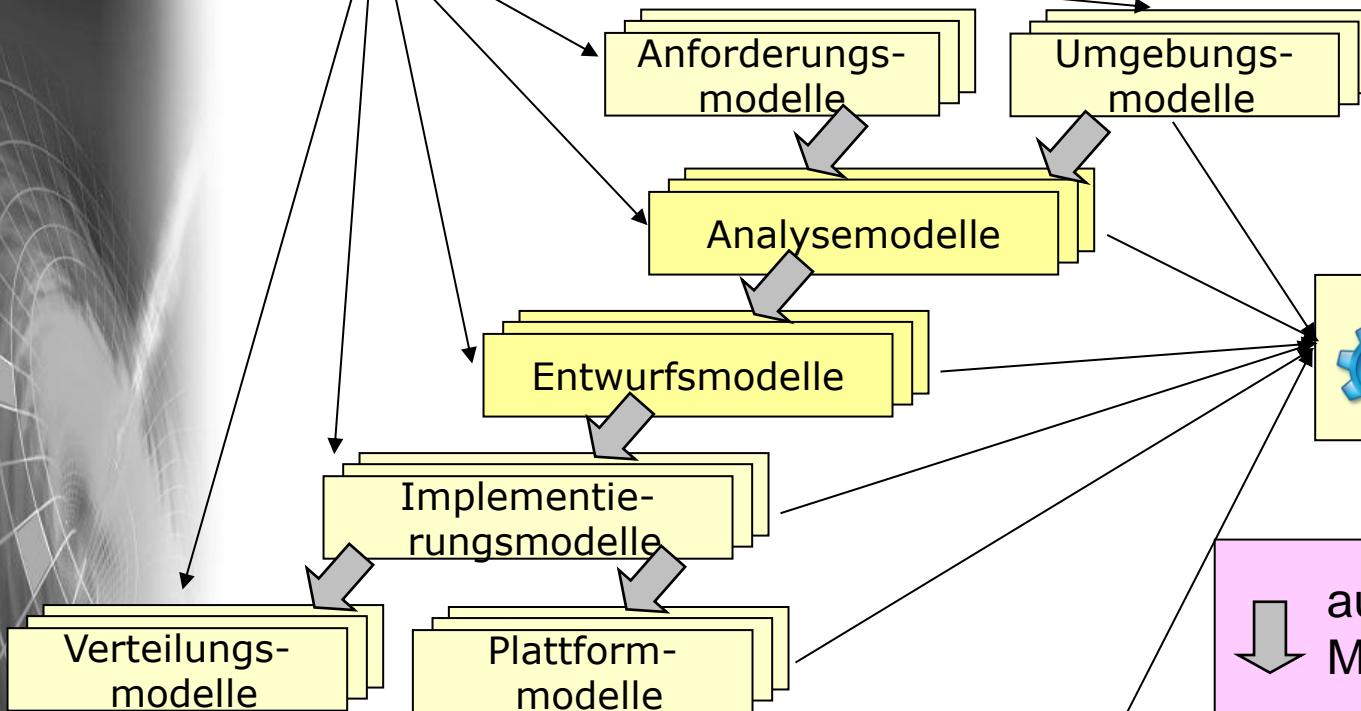


Objektorientierte Simulation mit...

Neue Idee: Modellgetriebene Software-Entwicklung verteilter Systeme (vereinfacht)



Anforderungen

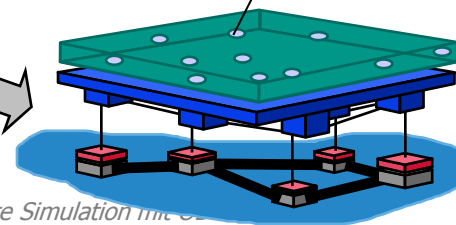


Modell-Checker
Simulator
Tester

↓ automatisierte
Modelltransformation

Zielcode

Binär-Komponenten



Objektorientierte Simulation mit...

Modellgetriebene Software-Entwicklung

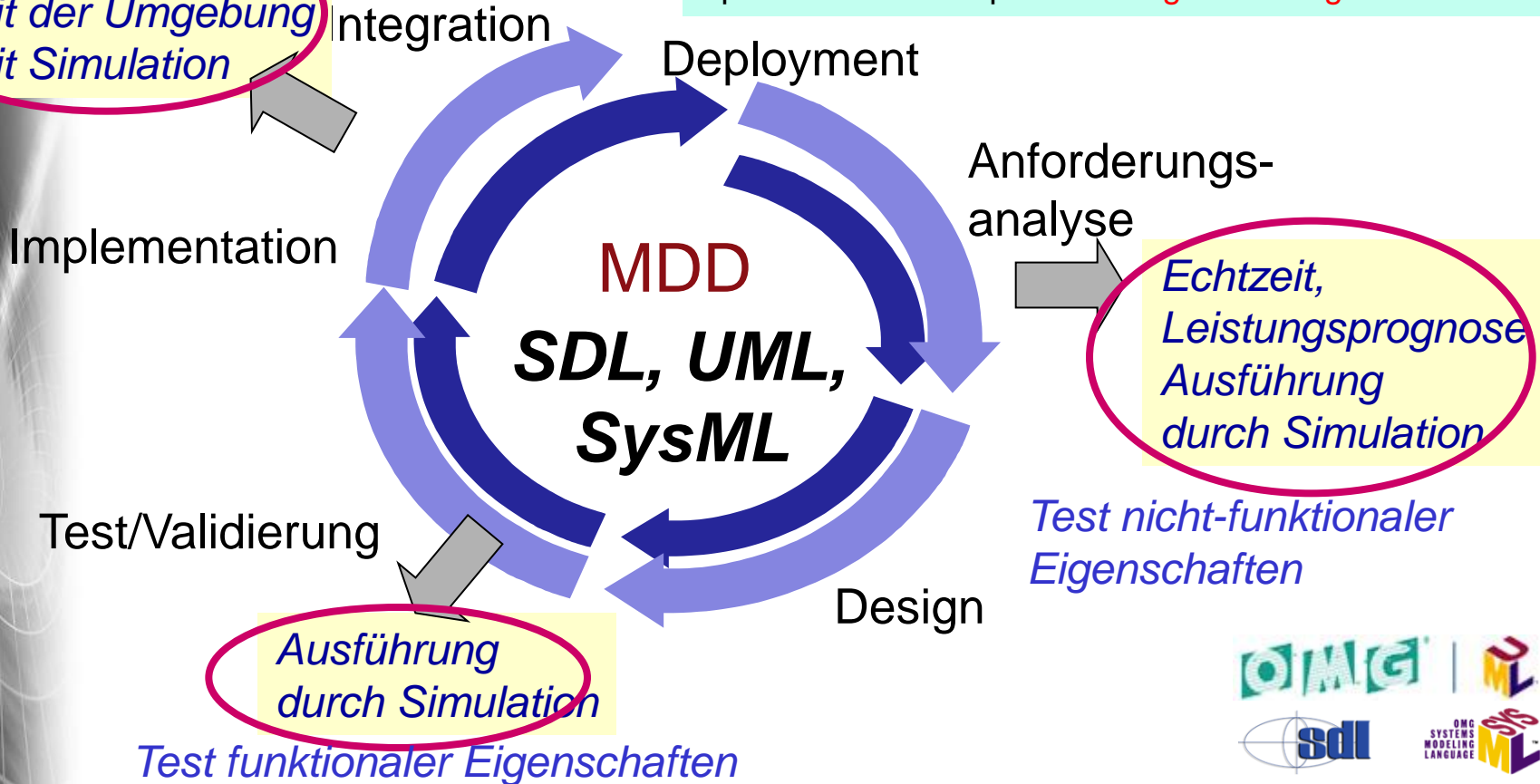
spiralförmig, inkrementell & iterativ

Test funktionaler und
nicht-funktionaler
Rückkopplungen

MDD:= Model Driven Development

- SW-Entwicklung ist **modellzentriert** (Modelle begleiten ges. SW-Lebenszyklus)
- automatische Transformationen für Modellübergänge
- spezifische Analysen (Checker, Simulatoren, ...)
- partielle oder komplette **Codegenerierung**

Wechselwirkung
mit der Umgebung
mit Simulation



Modellgetriebene Software-Entwicklung

spiralförmig, inkrementell & iterativ

Anwendungsobjekte in unterschiedlichen Sprachen: C++, Java, ...

PSM-1

CORBA

unterschiedliche Betriebssysteme

austauschbare Basiskommunikationstechnologie (Draht, Luft)

Implementation

! (OMG)

PSM-2

EJB

PIM

Modellierungssprachen

UML, ...

Integration

Deployment

Anforderungsanalyse

Test/Validierung

Design

automatisierte Modelltransformation

Objektorientierte Simulation mit ...

Model-Driven Architecture (Leitsätze)

- **Formalisierung** ist ein wichtiger Baustein für ein erfolgreiches **Qualitätsmanagement** in Softwareprojekten.
Speziell in den Bereichen der Anforderungs- und Systemanalyse besteht häufig noch ein hohes Optimierungspotential.
- Ein möglicher Weg, um den **Formalisierungsgrad** von Projektinformationen zu erhöhen, ist die Verwendung von **formal eindeutigen Modellen**.
Für den erfolgreichen Einsatz von Modellen ist es jedoch unabdingbar, die **Syntax und die Semantik der Modelle (über Metamodelle)** exakt festzulegen.
Ist dies einmal geschehen, ergibt sich meist eine deutliche Steigerung der Qualität wie auch der Effizienz in der Projektarbeit.
- Über den gezielten **Einsatz von Metamodellen** in der Softwareentwicklung können große Teile der Prozessaktivitäten automatisiert werden.
Dennoch muss berücksichtigt werden, dass die Formalisierung eines Softwareentwicklungsprozesses **nicht in einem Schritt** erfolgen kann.
Sie sollte vielmehr als ein **iterativer Prozess** verstanden werden, in dem die entstehenden Metamodelle von Projekt zu Projekt immer weiter verfeinert werden müssen.

Fazit

- Modellierung in allen Wissenschaftsdisziplinen das zentrale Paradigma zum Verständnis komplexer realer oder hypothetischer Systeme
(auch in bestimmten Bereichen der Informatik)
- In der SW-Entwicklung lange Zeit nicht hoffähig :
Alternative: von der Idee direkt zum gut dokumentierten Quellcode
aber: Komplexität der Systeme bereiten praktische Probleme
- **Achtung**: MDD verlangt nicht nur Konzepte,
sondern integrierte Werkzeugunterstützung

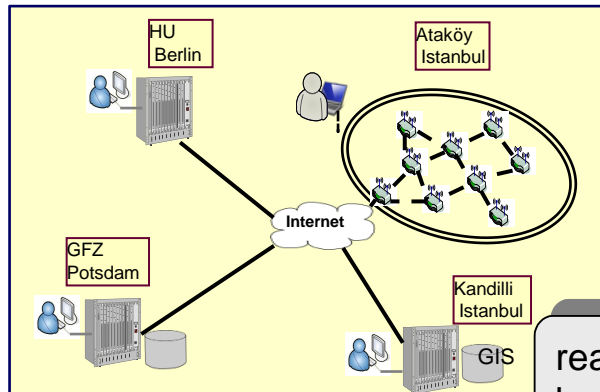
→OMSI: Technologien zum Bau effizienter Simulatoren dynamischer Systeme

1. *Einführung*

1. Systemsimulation – was ist das?
2. Ein Blick zurück in die Anfänge
3. Modelle und Originale
4. Modellierungssprachen, Simulationsumgebungen
5. Beispiel(e) aus der aktuellen Forschung
6. Paradigma der objektorientierten Modellierung
7. Klassifikation dynamischer Systeme
8. Scheduler für zeitdiskrete und zeitkontinuierliche Systemmodelle
9. M&S eines Niedertemperaturofens

1. Projekt: Erdbebenfrühwarnsystem

Sonderfall:
Modell und Original-
Teile
repräsentieren sich
als Software



reales
bzw. gedachtes
Phänomen

Ausführungs-
Modell im Speicher
Simulator

Experiment-
Management-
System

Ziel

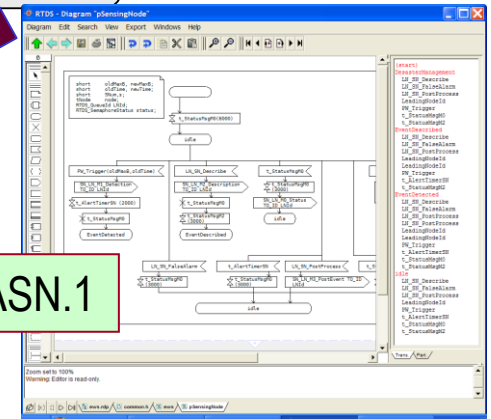
informales
/semiformales
Systemmodell

unvollständige
Darstellung:
UML/SDL

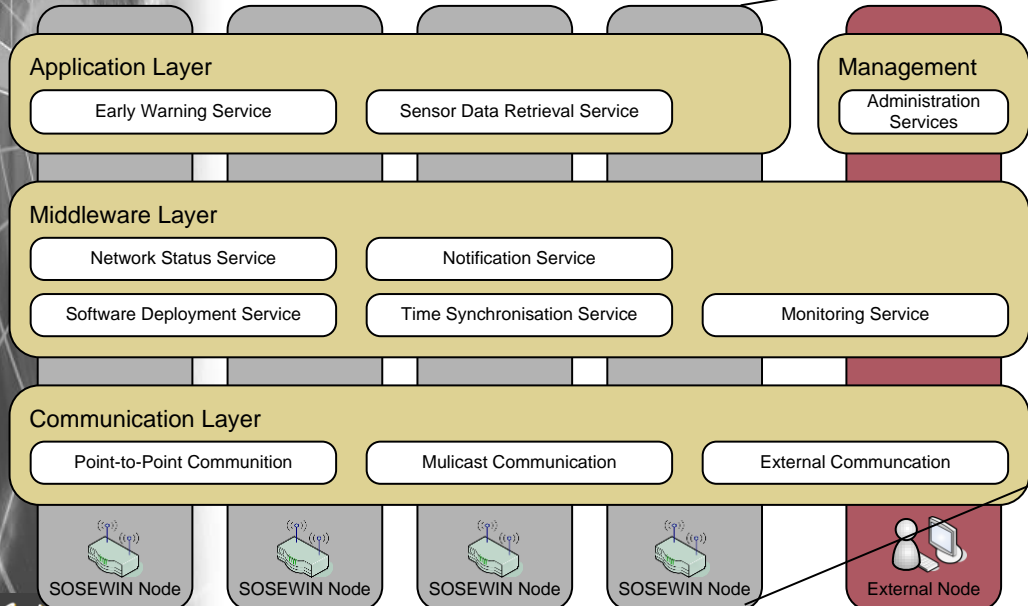
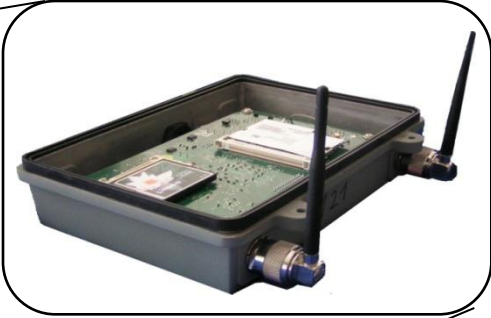
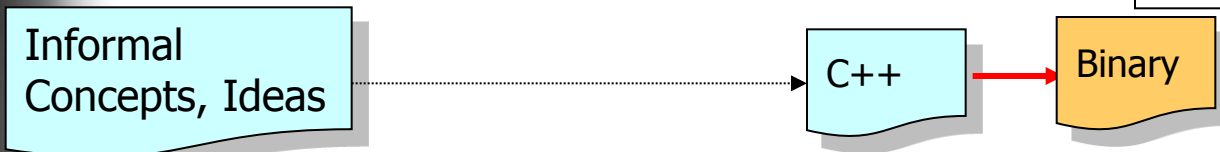
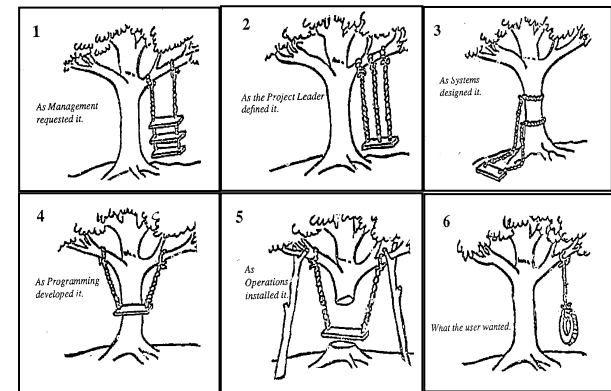
formales
mathematisches
Simulationsmodell

PIM: SDL,UML,ASN.1

PSM (Zielcode): C++, OpenWRT, Boost



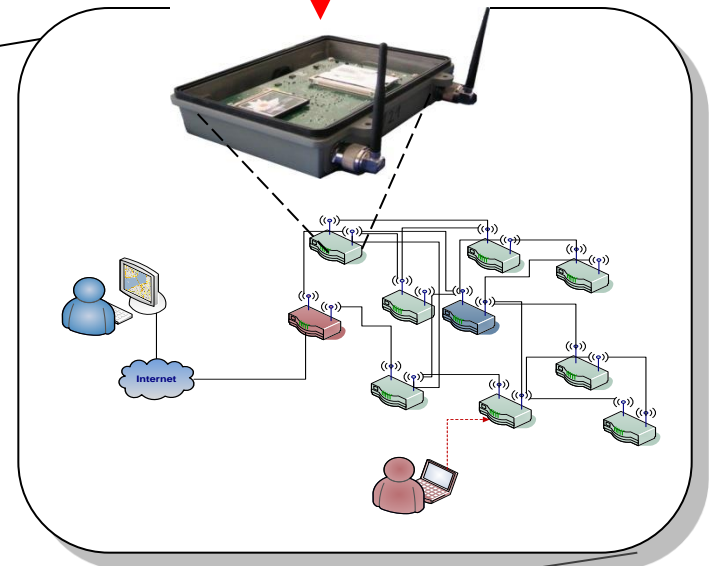
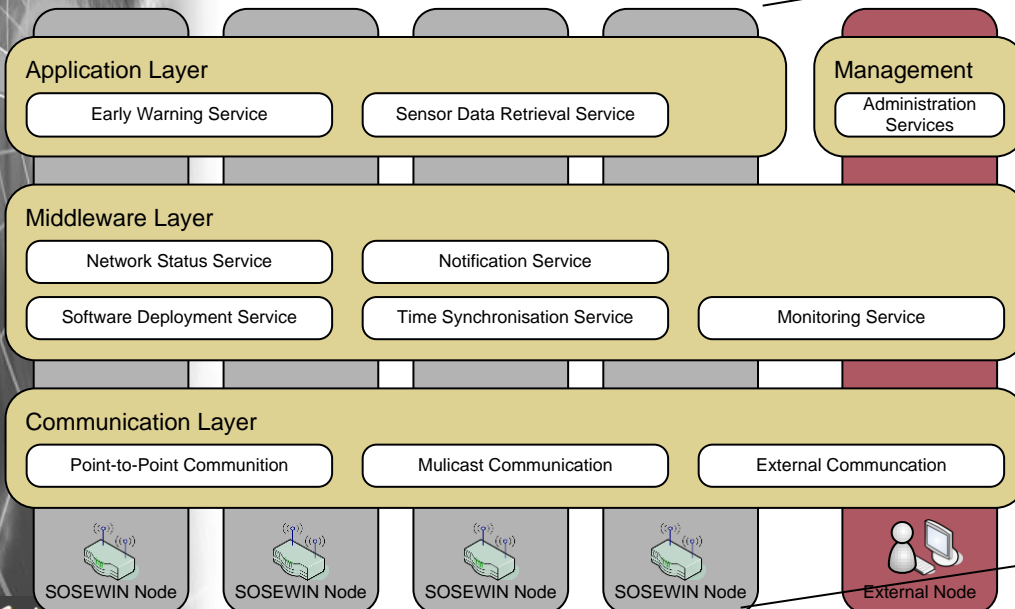
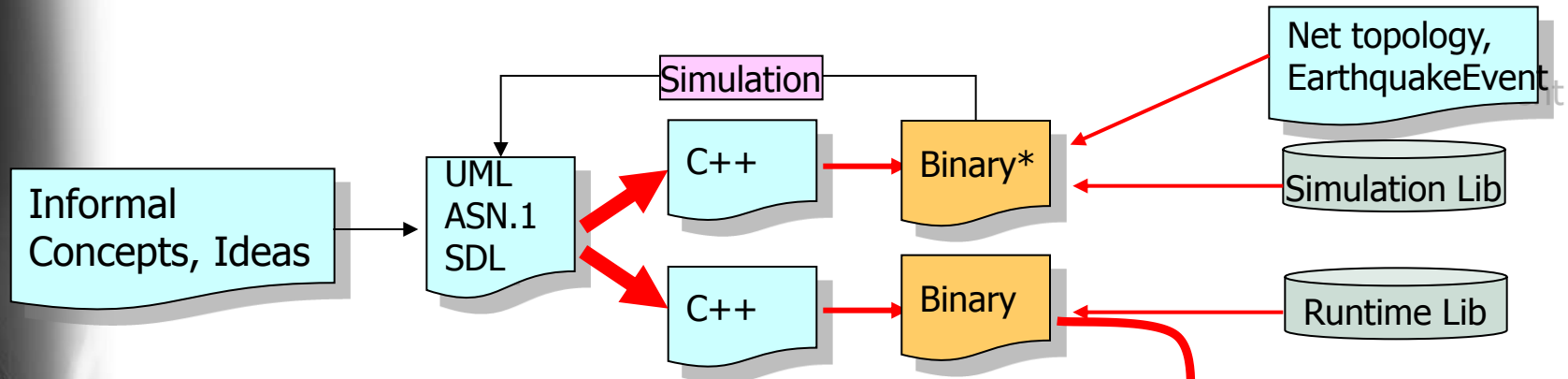
Traditionelle Software-Entwicklung



Test eines einzelnen Knotens reicht nicht

Test eines kompletten Systems ist problematisch

Ein modellgetriebener Ansatz



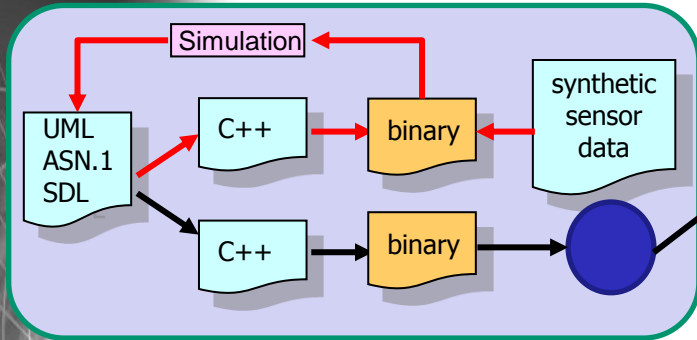
Simulation des kompletten Systems vereinfacht das Testen enorm

SOSEWIN-Überblick

Self-organized Seismic Early Warning Information Network

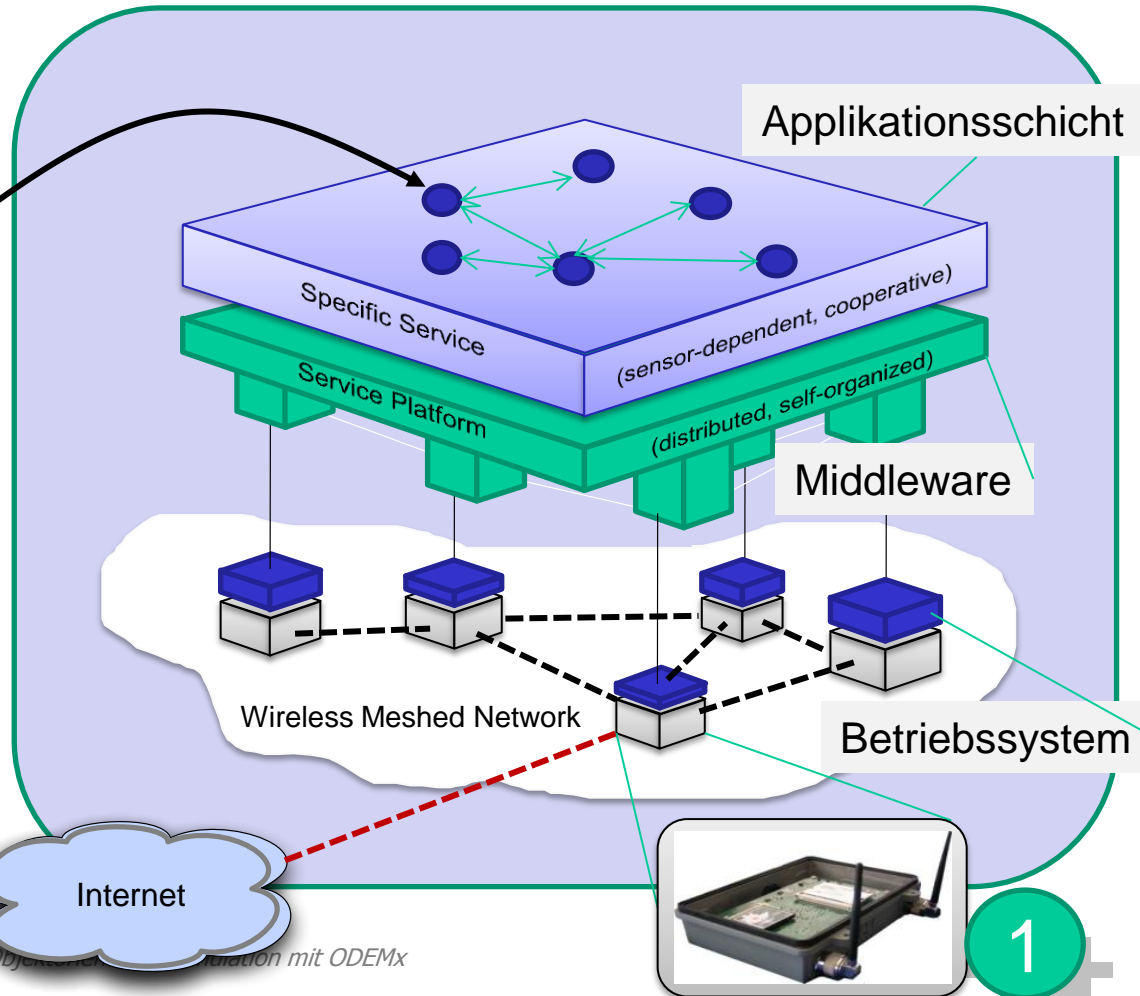
3

SW- Entwicklungstechnologie
(Modelleditor, Simulator, Code -Generator, ...)



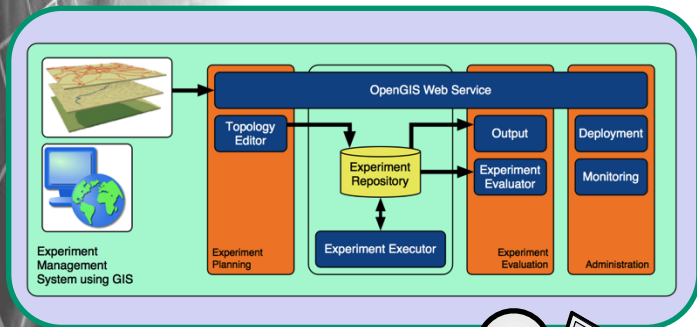
2

SOSEWIN-HW/SW Architektur



4

GIS-basierte Netzmanagement-
und Experiment-Unterstützung



Netz-Prototyp

Self-organized Seismic Early Warning Information Network



HU
Berlin



5

Ataköy
Istanbul

GIS



Internet



Kandilli
Istanbul

GFZ
Potsdam



GIS



Istanbul-
Infrastruktur-
daten



Objektorientierte Simulation mit ODEMx

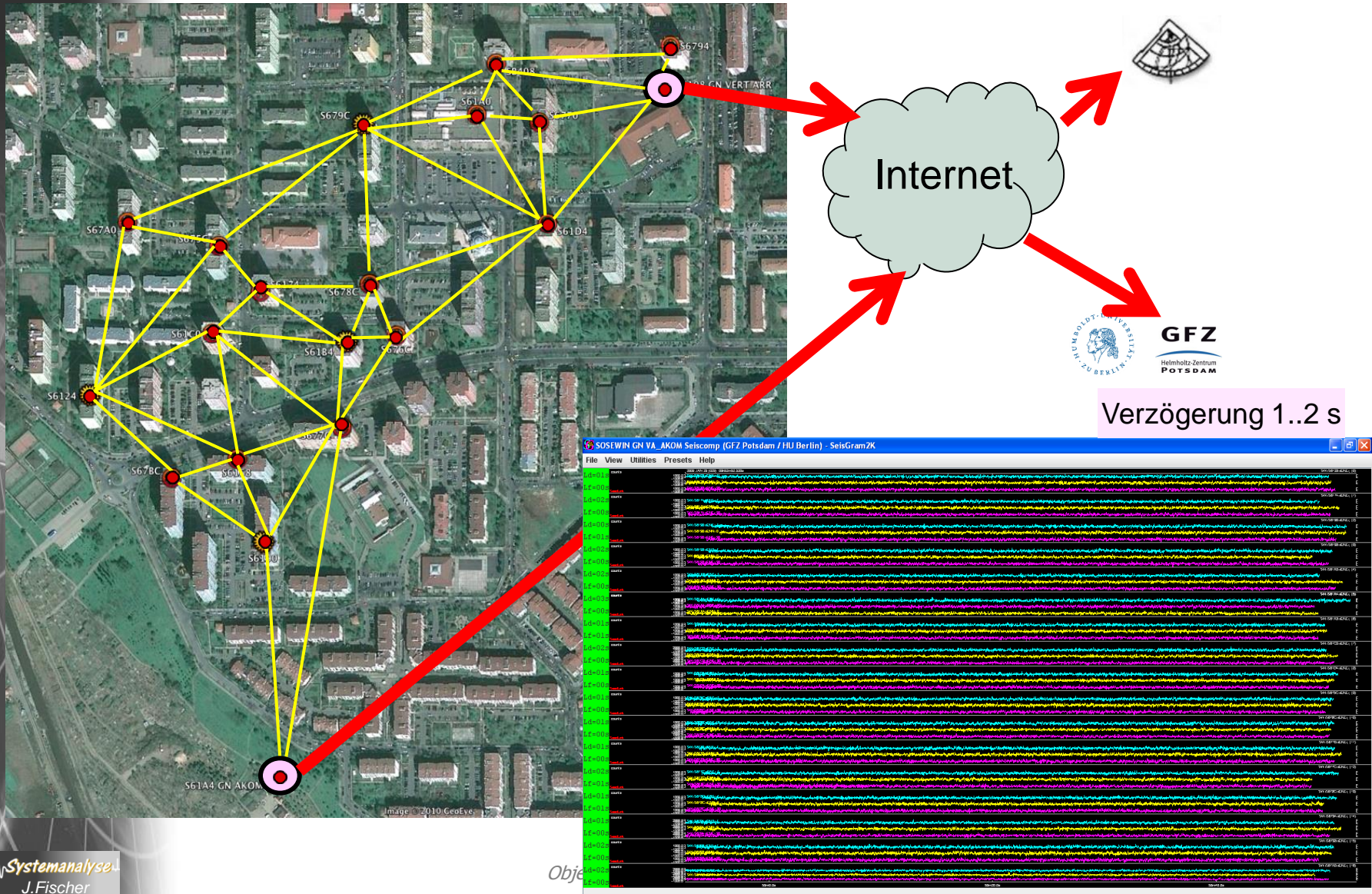
Erstinstallation von SOSEWIN



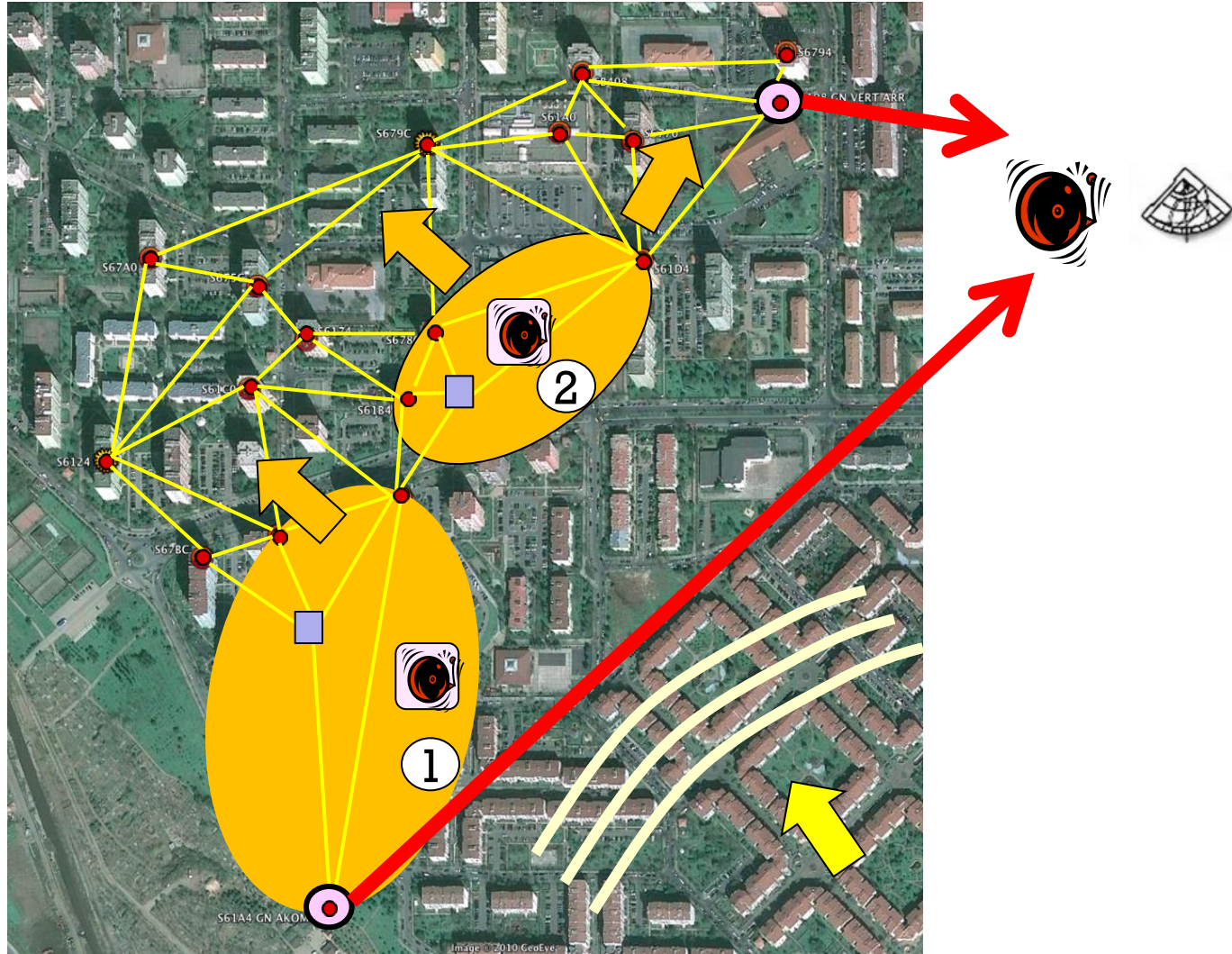
Istanbul, Mai 2008

mit Sensorik, aber zunächst noch ohne Alarmierungssoftware

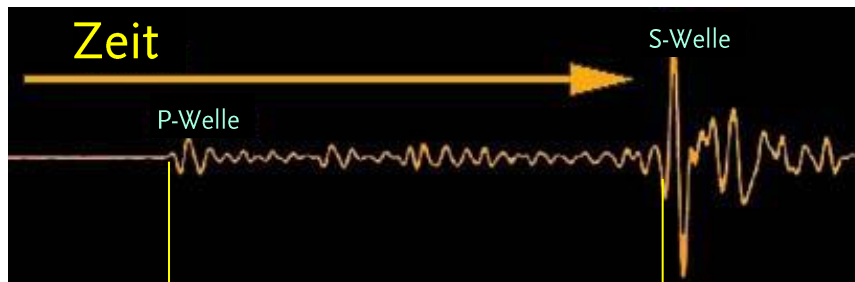
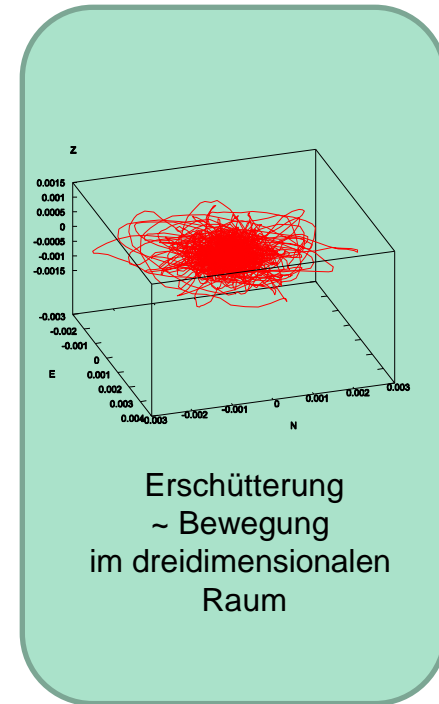
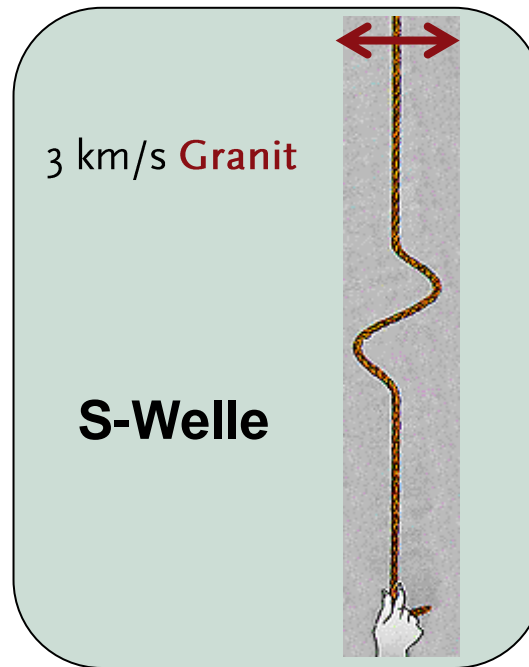
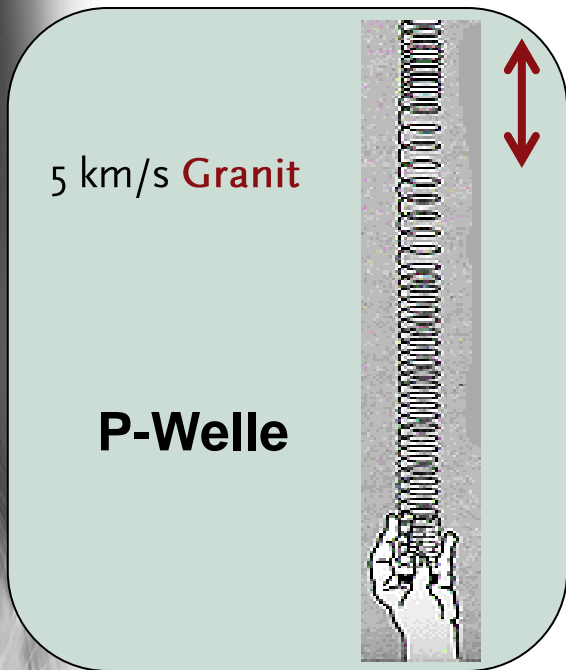
Alle Seismometer arbeiten zusammen



Alarm bei Eintreffen der P-Welle



Wellenarten

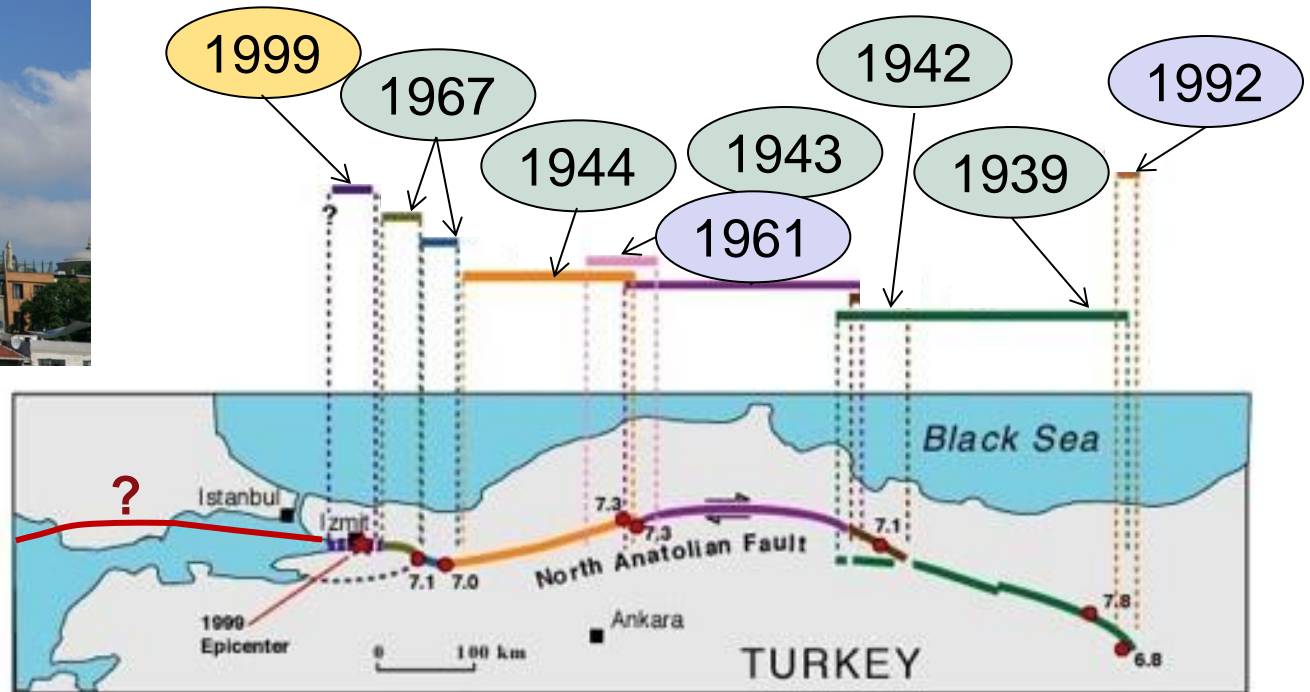


Vorwarnzeit



Objektorientierte Simulation mit ODEIMX

Ernsteste Bedrohung von Istanbul

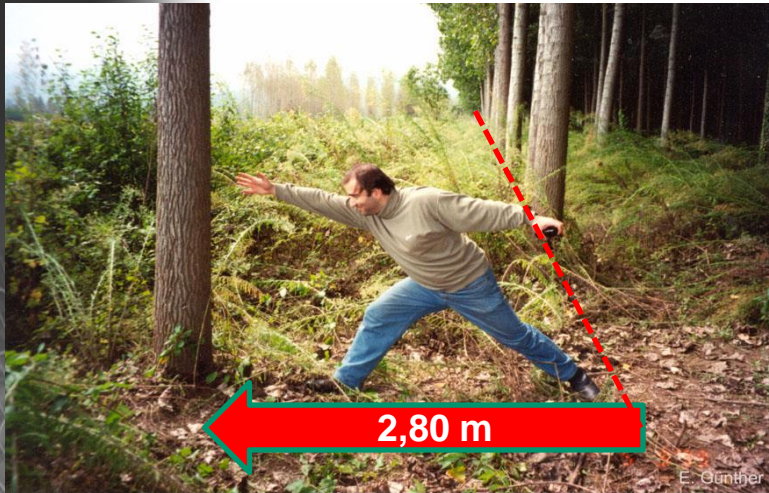


Quelle:
<http://pubs.usgs.gov/circ/2000/c1193/c1193.pdf>

Izmet-Beben: M 7,4 ~ 125-fache Energie der Hiroshima-Bombe

- 20.000 zerstörte Häuser
- 40.000 Verletzte
- 24.000 Tote

Letzte Warnung 1999: Izmit-Beben



A right-lateral displacement of 2.8 m was observed at the Sapanca segment near Caybasi (40.703° N, 30.451° E)

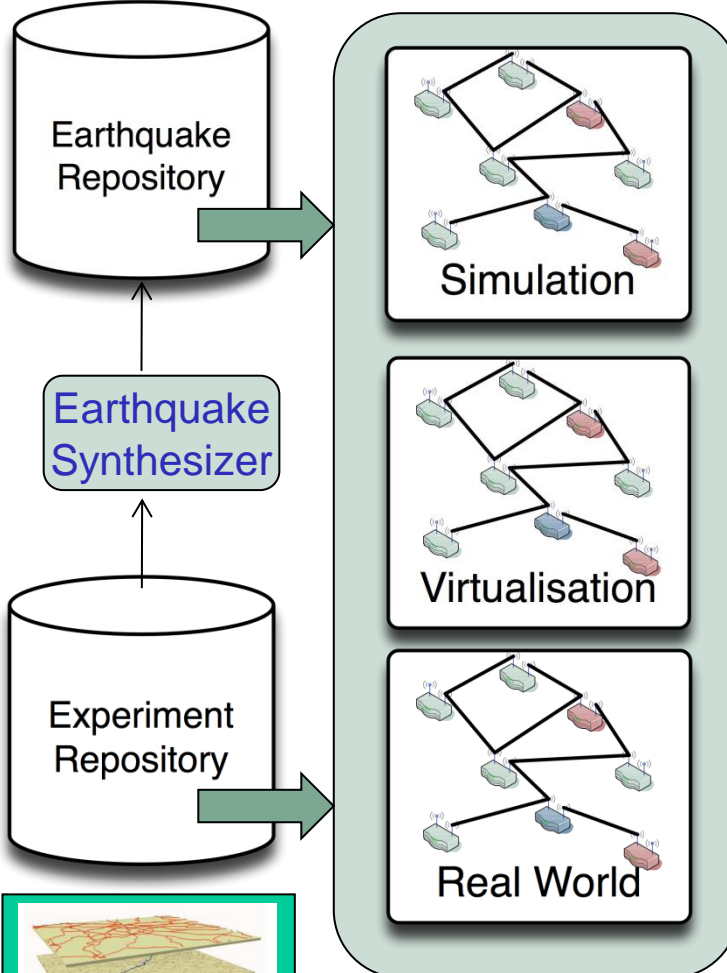
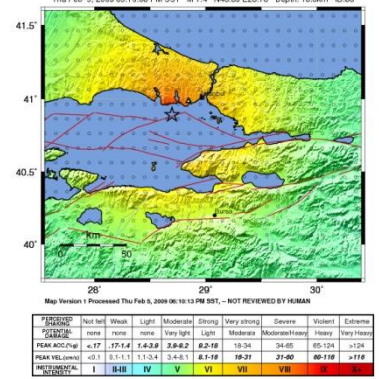


erte Simulation mit ODEMx

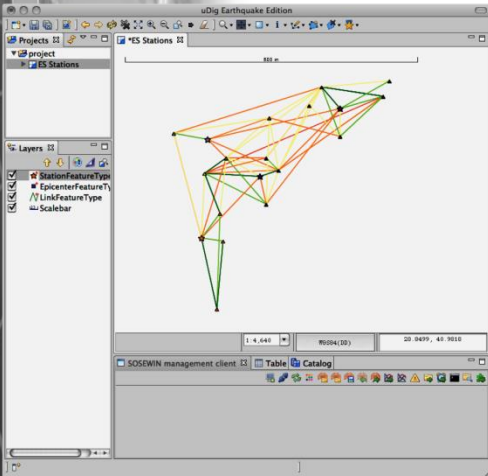
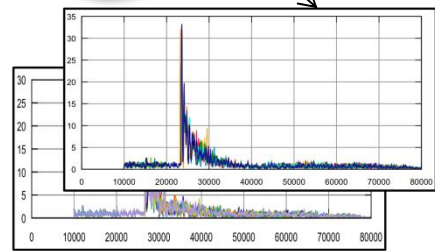
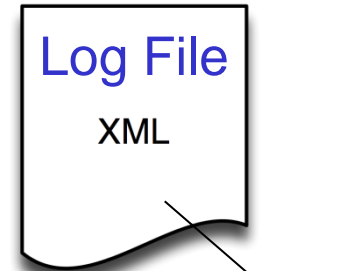
Experiment Management System

Executables

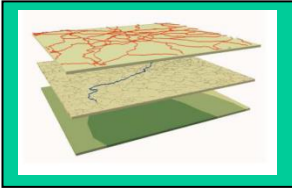
GFZ ShakeMap : Location description default. Created by Network Editor.



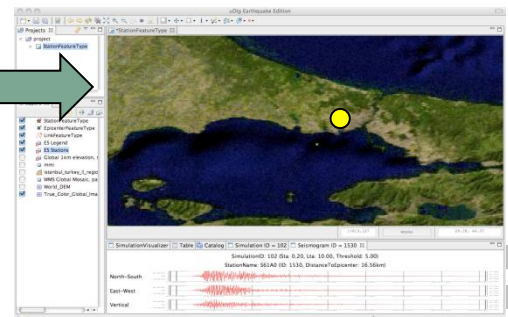
Simulation ID	Simulation Name	Status	Start Time	End Time	Duration	File Size	Output Files
100	Simulation 100	Completed	2009-08-24 12:00:00	2009-08-24 12:05:00	00:05:00	100 MB	Output 100
101	Simulation 101	Completed	2009-08-24 12:05:00	2009-08-24 12:10:00	00:05:00	100 MB	Output 101
102	Simulation 102	Completed	2009-08-24 12:10:00	2009-08-24 12:15:00	00:05:00	100 MB	Output 102
103	Simulation 103	Completed	2009-08-24 12:15:00	2009-08-24 12:20:00	00:05:00	100 MB	Output 103
104	Simulation 104	Completed	2009-08-24 12:20:00	2009-08-24 12:25:00	00:05:00	100 MB	Output 104
105	Simulation 105	Completed	2009-08-24 12:25:00	2009-08-24 12:30:00	00:05:00	100 MB	Output 105
106	Simulation 106	Completed	2009-08-24 12:30:00	2009-08-24 12:35:00	00:05:00	100 MB	Output 106
107	Simulation 107	Completed	2009-08-24 12:35:00	2009-08-24 12:40:00	00:05:00	100 MB	Output 107
108	Simulation 108	Completed	2009-08-24 12:40:00	2009-08-24 12:45:00	00:05:00	100 MB	Output 108
109	Simulation 109	Completed	2009-08-24 12:45:00	2009-08-24 12:50:00	00:05:00	100 MB	Output 109
110	Simulation 110	Completed	2009-08-24 12:50:00	2009-08-24 12:55:00	00:05:00	100 MB	Output 110



GIS-based editor

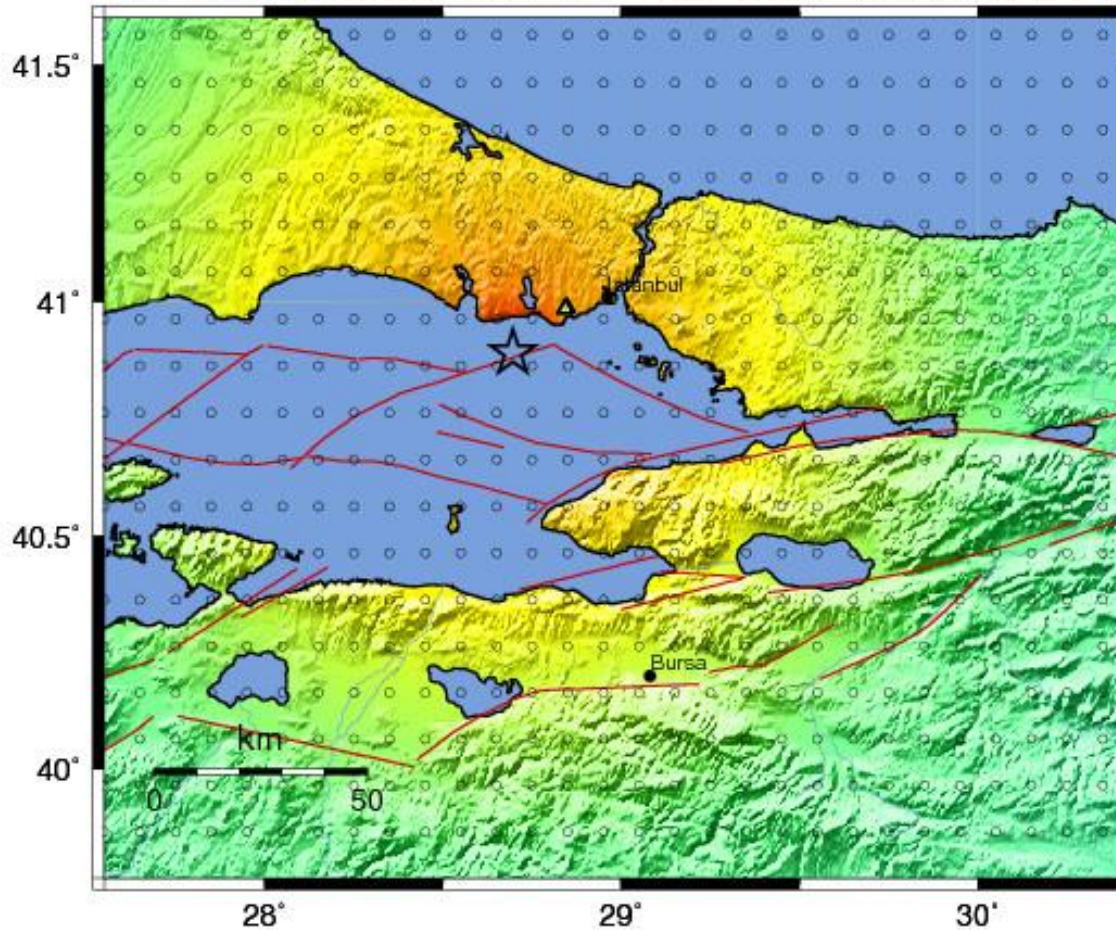


orientierte Simulation mit ODEMx



GFZ ShakeMap : Location description default. Created by Network Editor.

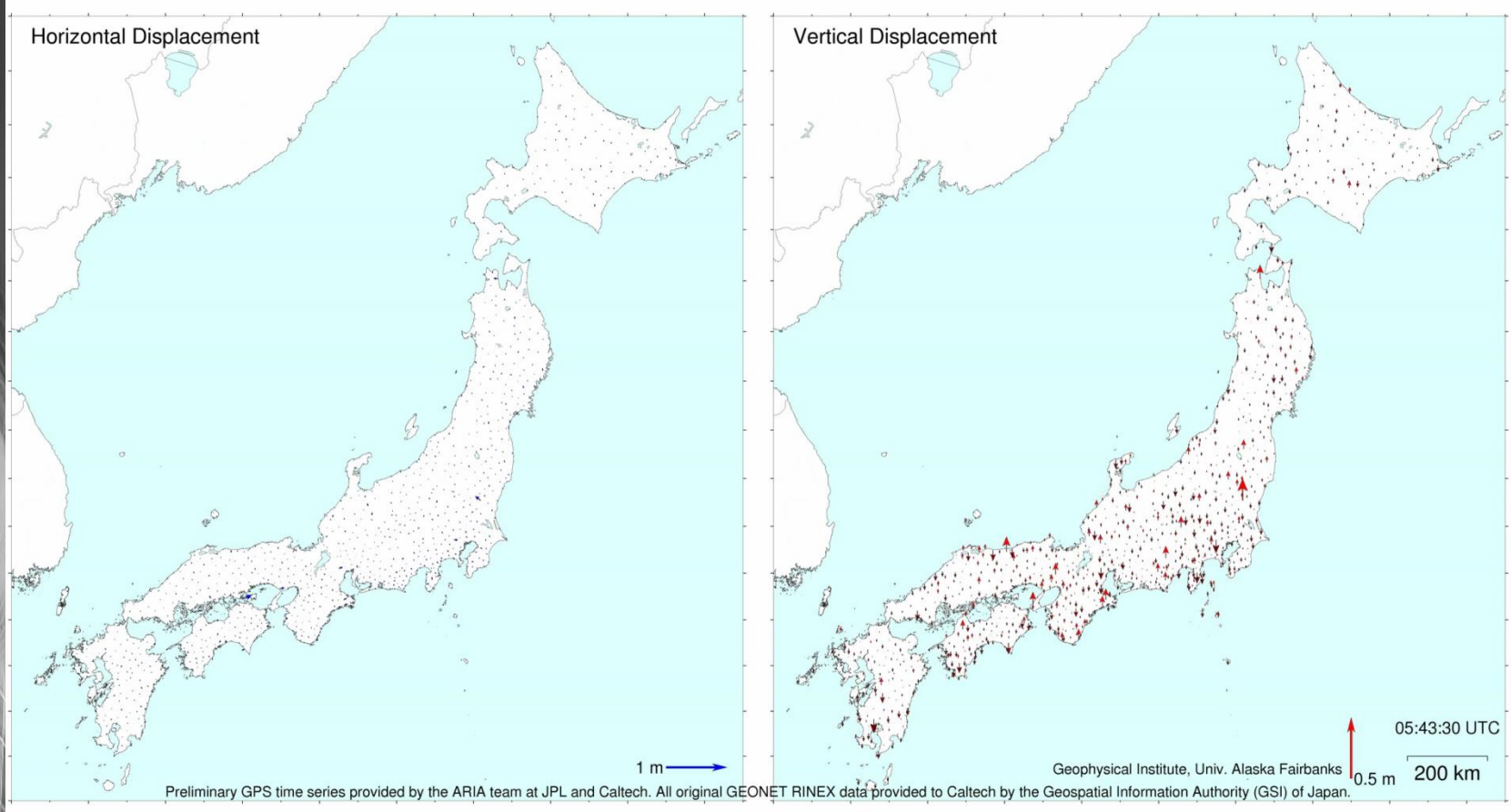
Thu Feb 5, 2009 05:16:06 PM SST M 7.4 N40.89 E28.70 Depth: 10.0km ID:60



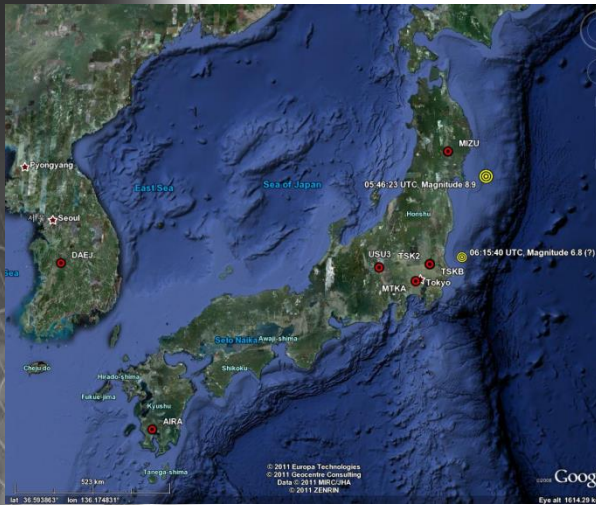
Map Version 1 Processed Thu Feb 5, 2009 06:10:13 PM SST, -- NOT REVIEWED BY HUMAN

PERCEIVED SHAKING	Not felt	Weak	Light	Moderate	Strong	Very strong	Severe	Violent	Extreme
POTENTIAL DAMAGE	none	none	none	Very light	Light	Moderate	Moderate/Heavy	Heavy	Very Heavy
PEAK ACC.(%g)	<.17	.17-1.4	1.4-3.9	3.9-9.2	9.2-18	18-34	34-65	65-124	>124
PEAK VEL.(cm/s)	<0.1	0.1-1.1	1.1-3.4	3.4-8.1	8.1-16	16-31	31-60	60-116	>116
INSTRUMENTAL INTENSITY	I	II-III	IV	V	VI	VII	VIII	IX	X+

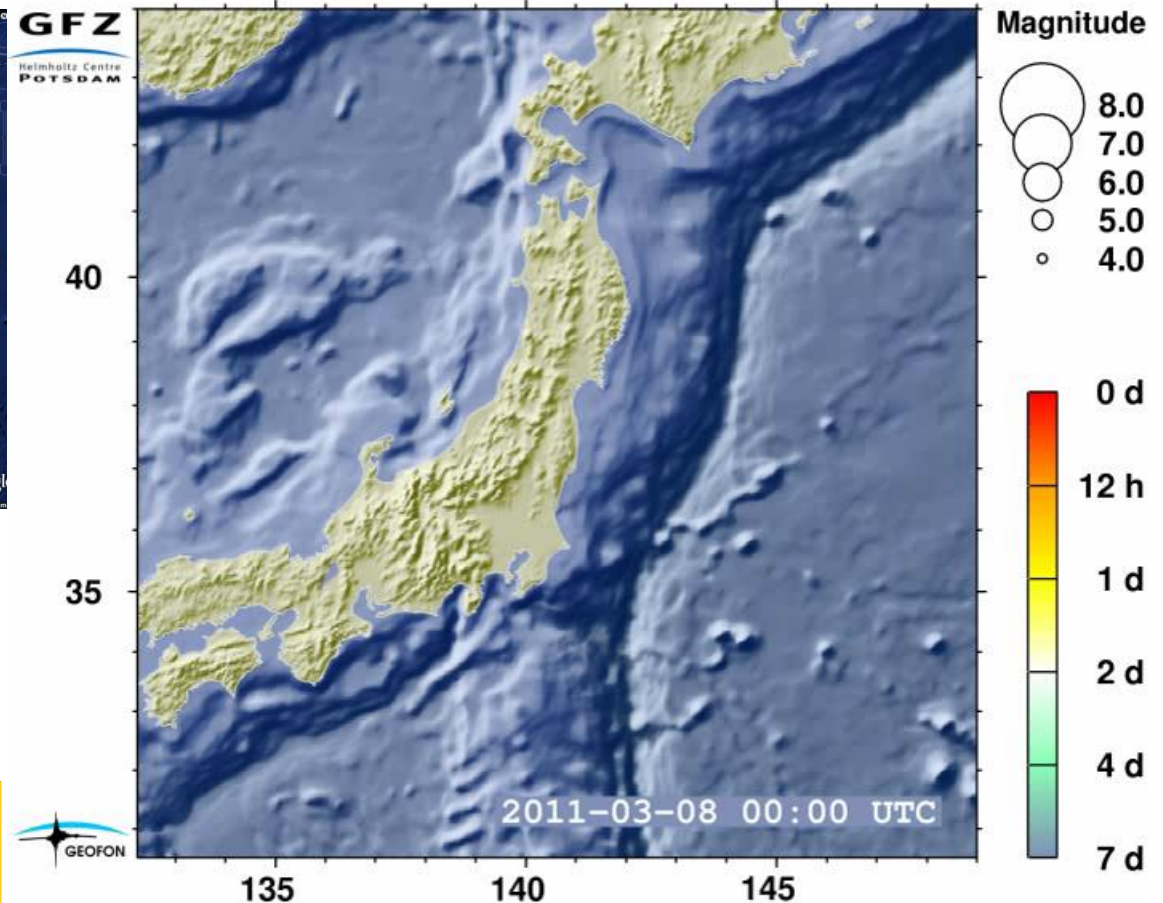
Bedeutung von Animationen: *Visualisierung von Erdbewegungen*



Experimentalauswertung: Bedeutung von Visualisierung



hunderte Nachbeben,
über einen ganzen Monat
verteilt



Objektorientierte Simulation mit ODEMx