

MEMILIO WORKSHOP ON MULTISCALE INFECTIOUS DISEASE MODELING

Julia Bicker, Carlotta Gerstein, David Kerkmann, Sascha Korf, René Schmieding, Anna Wendler, Henrik Zunker, Kilian Volmer, Martin Kühn (based on contributions of many others)

Pre-Conference Workshop, 4th (Inter-) National Conference on Infectious Disease Modeling

Halle, 2026/05/19

Gefördert durch:



GEFÖRDERT VOM



With funding from the:



HELMHOLTZ

DFG



Welcome

9:00 – 9:15

Agenda

8:30 – 9:00 Arrival with coffee

9:00 – 9:15 Welcome

9:15 – 10:00 **General introduction to**

- Software modularization
- Unit tests
- Aspects of computational performance
- Model types for infectious disease dynamics

10:00 **Fire alarm...**

10:00 – 10:45 Exercise: Setting up a **simple model**

10:45 – 11:00 Coffee break

11:00 – 11:30 Exercise:

*Flows to compute **new infections** or **hospitalizations***

11:30 – 12:15 Exercise:

Realizing **demographic stratification**

12:15 – 12:45 Lunch break

12:45 – 14:15 Exercise:

Two ways for **integrating mobility and its effects**

14:15 – 15:00 Exercise: **Locations and**

Nonpharmaceutical interventions

15:00 – 15:15 Coffee break

15:15 – 16:00 Exercise:

Non-pharmaceutical interventions with **dynamic conditions**

16:00 – 16:15 How to generate your own (ODE) model

16:15 – 16:30 Country-specific simulations

16:30 – 17:00 Overview of the **full modeling landscape**

17:00 – 17:15 Closing





GEFÖRDELT VOM

Bundesministerium
für Bildung
und Forschung

Workshop on multiscale infectious disease modeling

May 19, Halle (Saale)

Today's mentors:



Anna

Anna.Wendler
@DLR.de



Carlotta

Carlotta.Gerstein
@uni-bonn.de



David

David.Kerkmann
@Theoretical-
Biology.de



Kilian

Kilian.Volmer
@uni-bonn.de



Martin

Martin.Kuehn
@DLR.de



René

Rene.Schmieding
@DLR.de



Federal Ministry
of Research, Technology
and Space

Gefördert durch:



Bundesministerium
für Digitales
und Verkehr

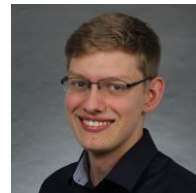
Further active core-developers:



Henrik



Julia



Max



Nils



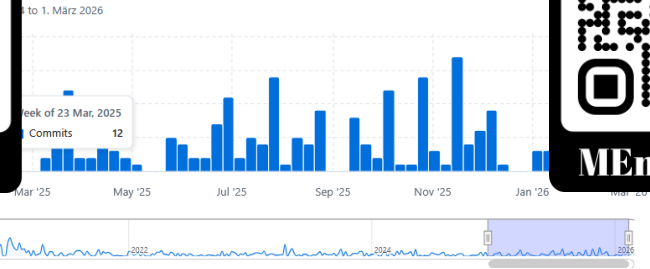
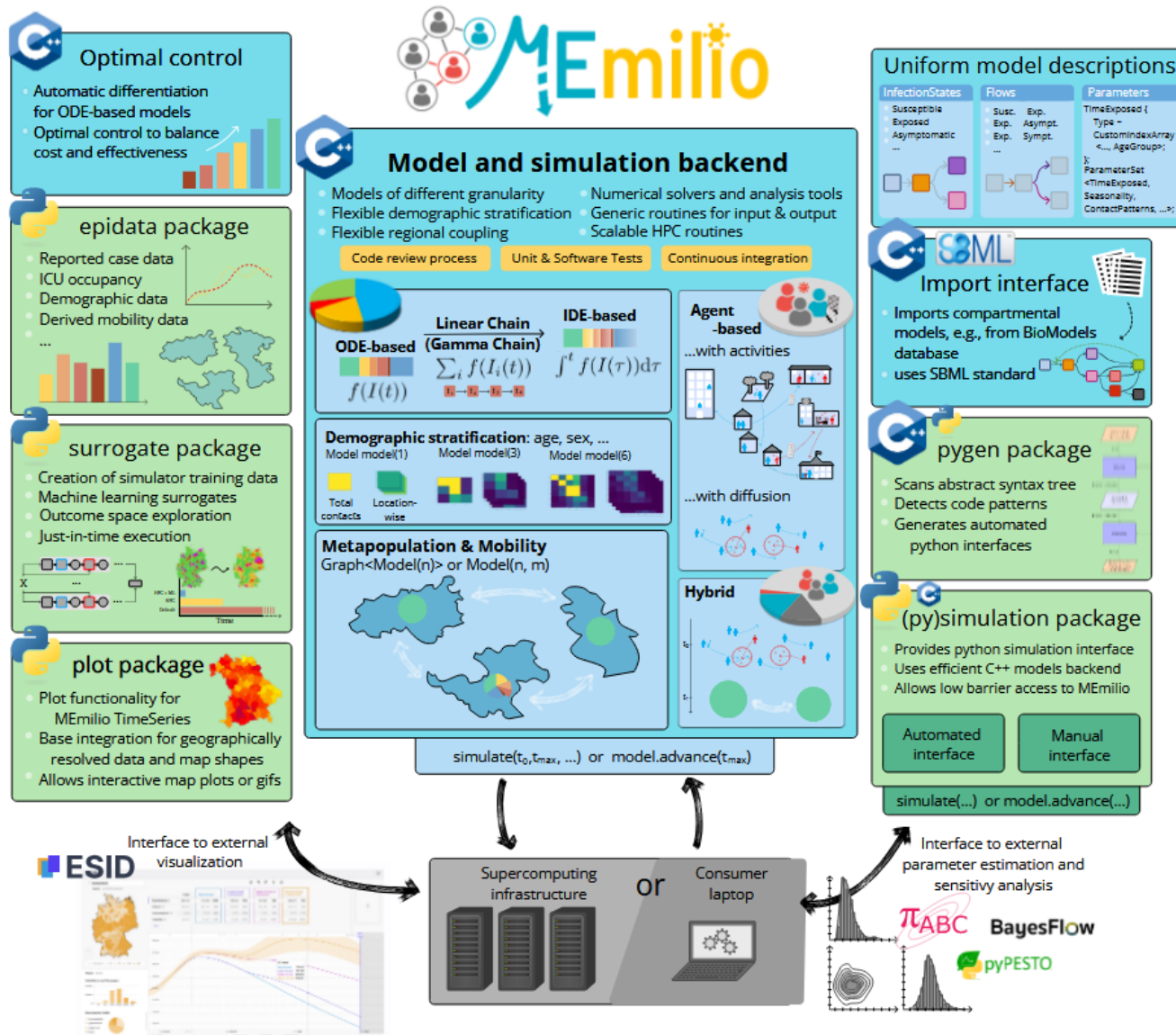
Sascha

Further acknowledgements:

Daniel Abele, Khoa Nguyen, Lena Plötzke, Agatha Schmidt, Lenz, Daniel Richter, Hannah Tritschak, Ralf Hannemann-Tamas, Julian Litz, Paul Johannsen, Marielena Borges, Annika Jungklaus, Manuel Heger, Annalena Lange, Elisabeth Kluth, Kathrin Rack, Wadim Koslow, Jan Kleinert, Vincent Wieland, Jonas Arruda, Sebastian Binder, Margrit Klitz, Martin Siggel, Manuel Dahmen, Achim Basermann, Michael Meyer-Hermann, Jan Hasenauer



A high performance Modular EpideMIcs simuLatIOn software



Contributors

- 2025: 20
- Overall: 42

Activity (on main) February 2026

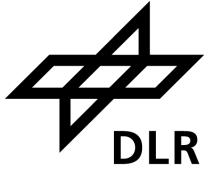
- 14 active pull requests (12 merged)
- 14 active issues (12 closed)
- 10 active authors

Software & Unit Tests



- C++: ca. 750 (2023: ca. 300)
- Python: ca. 230 (2023: ca. 200)

Talks @ MONID conference



For more detailed scientific talks, we would be happy to see you at the following talks:

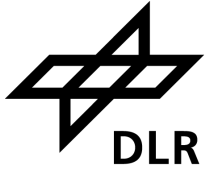
Date	Talk	Location
Wed, 15:00-15:15	Maximilian Betz, "Optimizing non-pharmaceutical interventions under constraints on epidemic dynamics" → DynamicNPIs	Festsaal
Thu, 09:15 - 09:30	Anna Wendler, "Comparing agent-based and equation-based models with respect to their inherent assumptions" → ODE, LCT, IDE, ABM	Vortragssaal
Thu, 09:30 - 09:45	Julia Bicker, "Hybrid frameworks for accurate and efficient simulation of epidemiological processes in highly stochastic regimes" → Hybrid	Vortragssaal
Thu, 09:45 - 10:00	Henrik Zunker, "A semi-continuous flow-based formulation to avoid quadratic system growth in Lagrangian metapopulation models" → Mobility	Vortragssaal
Thu, 09:45 - 10:05	Kilian Volmer, "Modeling the spread of Foot-and-mouth disease on the German cattle trading network" → Stochastic metapopulation	Festsaal
Thu, 16:15 - 16:30	Carlotta Gerstein, "Evaluating performance potentials for metapopulation models of infectious diseases" → Mobility & performance	Festsaal
Thu, 16:30 - 16:45	Sascha Korf, "Impact of transmission chain information loss on agent-based epidemic model predictions" → ABM	Festsaal

Questions, expectations, and wishes ?

General introduction

9:15 – 10:00

What is software modularization and why should I care?



Monolithic script: a common problem

```
// Import data
...

// Transform data for input
...

// Write model structure
// (potentially with demography and mobility)
...

// Initialize model
...

// Do calibration routine
// use simulation and
// numerical solver routine inside
```

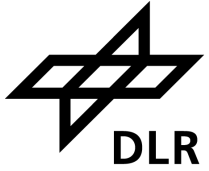
→ easy and fast to write from scratch

- Difficult to generalize or extend
→ „works“ per copy&paste of code segments
→ prone to errors through global variables or locally hard-coded ones
- Difficult to adapt
→ prone to errors through overlooking sections
→ validation becomes difficult
- Difficult to debug
→ needs substantial time to localize errors
- Difficult to maintain
→ for each change, entire script needs to be maintained
- Difficult to hand-over
→ project will probably die with creator leaving

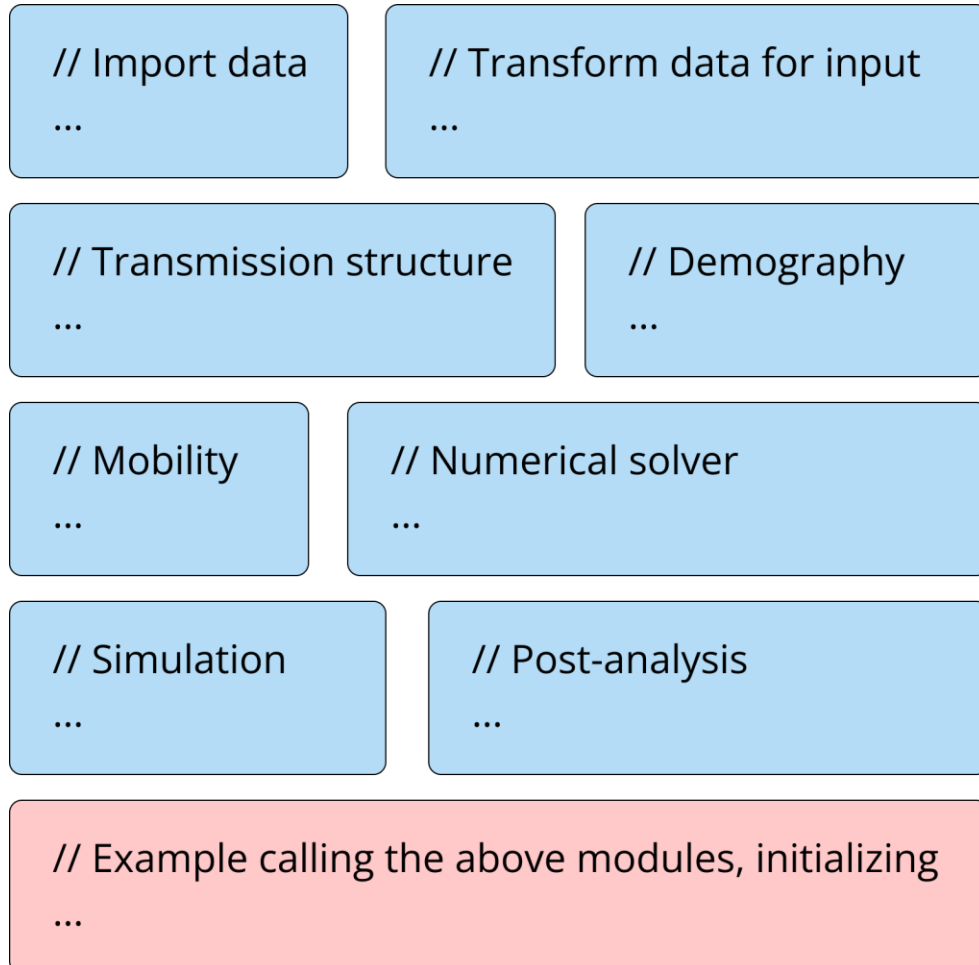
In general:

→ time-consuming to adapt/generalize and high risk of errors

What is software modularization and why should I care?

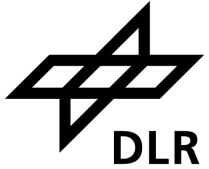


Modular design: challenges and advantages

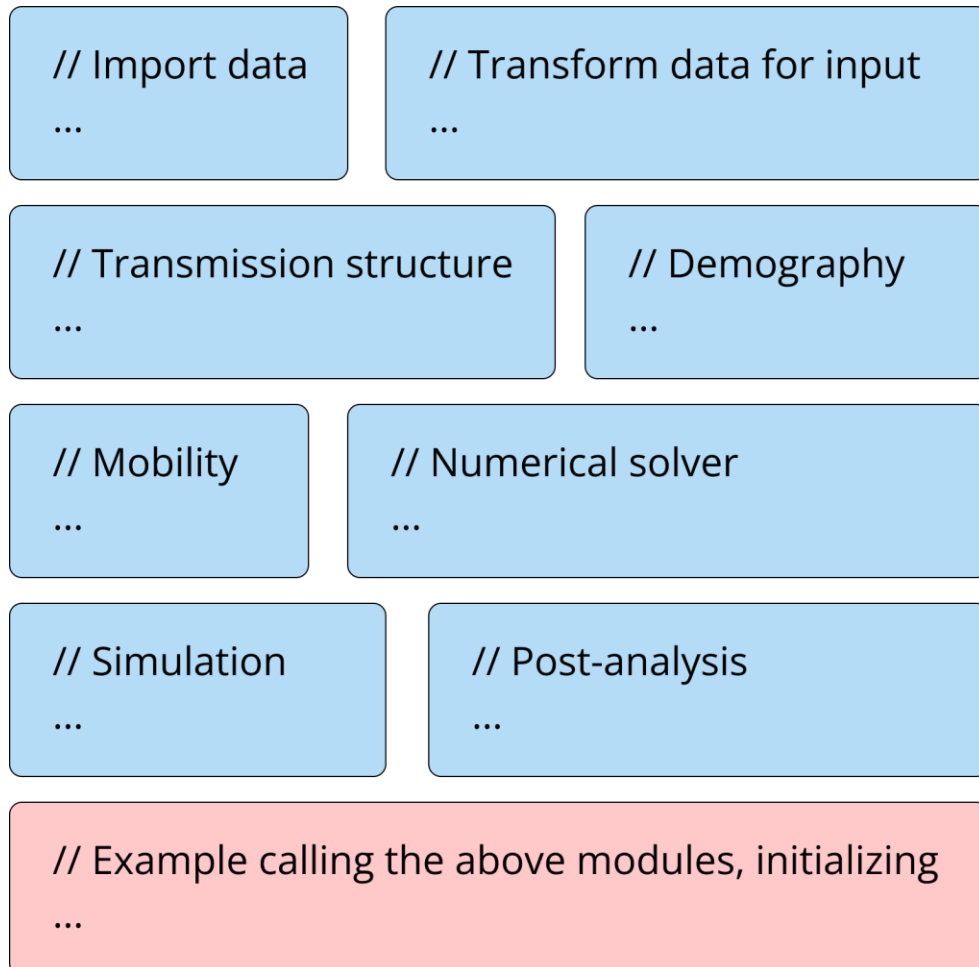


- Breaking a system into independent, self-contained modules
 - Each module has a clear responsibility
 - Modules interact through well-defined interfaces
- Divide a complex system into manageable pieces
- Time-consuming from scratch

What is software modularization and why should I care?



Modular design: challenges and advantages

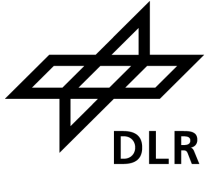


- Generalization or extension
→ generalize or select additional modules
- Adaptation
→ adapt or select additional modules
- Debugging
→ focus on failing module (see unit tests)
- Maintenance
→ focus on relevant module (see unit tests)
- Hand-over
→ Better understanding through functional divisions

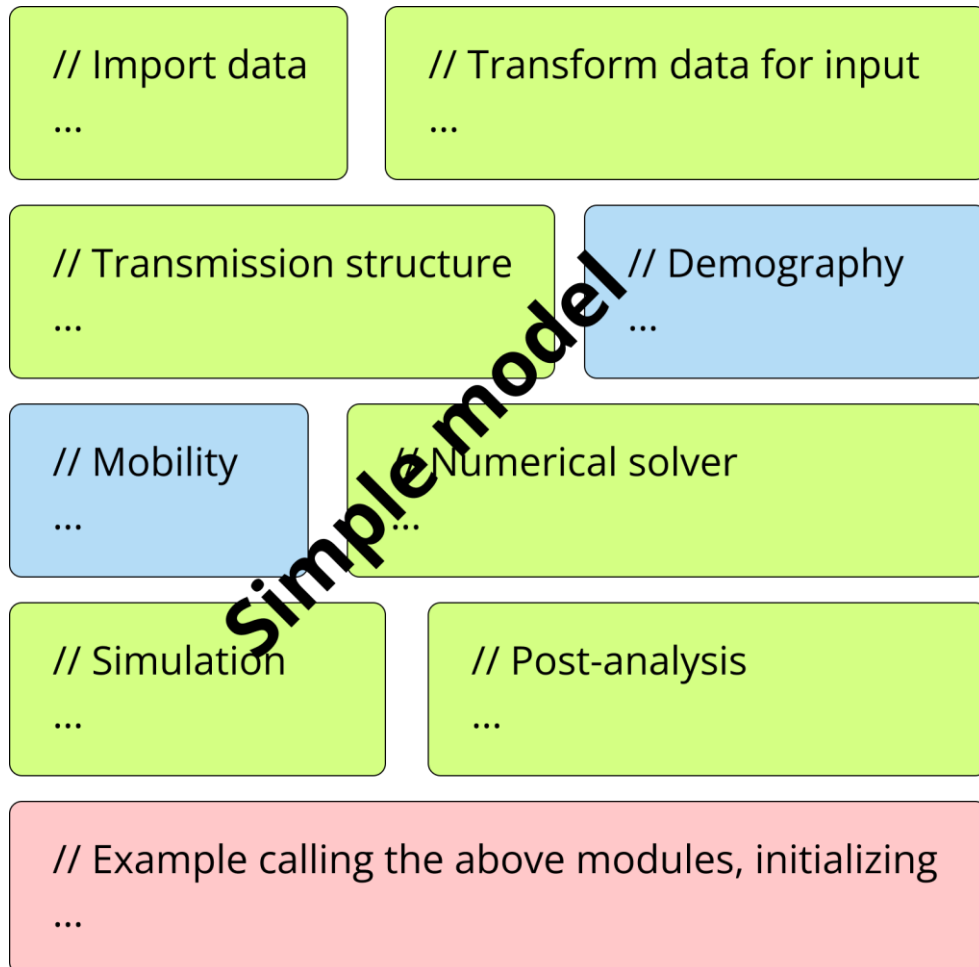
In general:

→ faster to adapt/generalize and reduced risk of errors

What is software modularization and why should I care?



Modular design: challenges and advantages

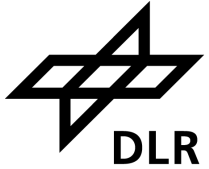


- Generalization or extension
→ generalize or select additional modules
- Adaptation
→ adapt or select additional modules
- Debugging
→ focus on failing module (see unit tests)
- Maintenance
→ focus on relevant module (see unit tests)
- Hand-over
→ Better understanding through functional divisions

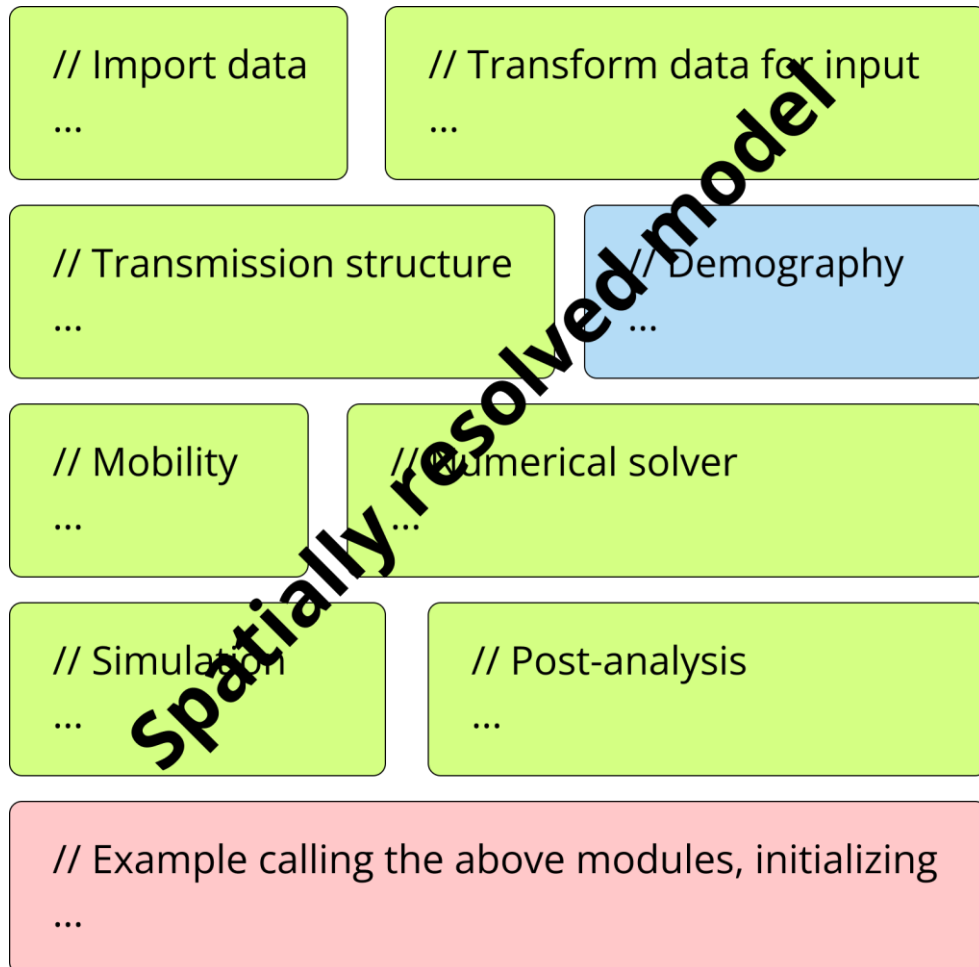
In general:

→ faster to adapt/generalize and reduced risk of errors

What is software modularization and why should I care?



Modular design: challenges and advantages



- Generalization or extension
→ generalize or select additional modules
- Adaptation
→ adapt or select additional modules
- Debugging
→ focus on failing module (see unit tests)
- Maintenance
→ focus on relevant module (see unit tests)
- Hand-over
→ Better understanding through functional divisions

In general:

→ faster to adapt/generalize and reduced risk of errors

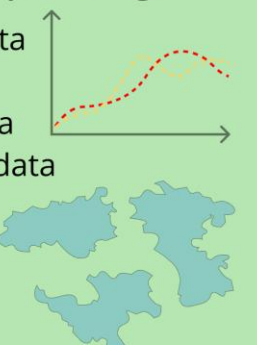
What is software modularization and why should I care?

Initial insights and example of modular design in MEmilio



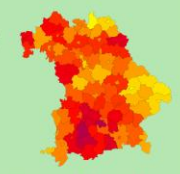
epidata package

- Reported case data
- ICU occupancy
- Demographic data
- Derived mobility data
- ...

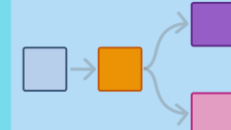
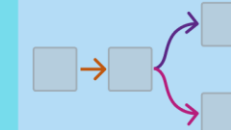


plot package

- Plot functionality for MEmilio TimeSeries
- Base integration for geographically resolved data and map shapes
- Allows interactive map plots or gifs

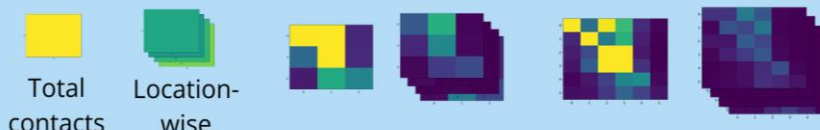


Uniform model descriptions

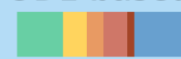
InfectionStates	Flows	Parameters
- Susceptible - Exposed - Asymptomatic ... 	- Susc. → Exp. - Exp. → Asympt. - Exp. → Sympt. ... 	<pre>TimeExposed { Type = CustomIndexArray <..., AgeGroup>; }; ParameterSet <TimeExposed, Seasonality, ContactPatterns, ...>;</pre>

Demographic stratification: age, sex, ...

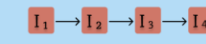
Model model(1) Model model(3) Model model(6)



ODE-based


$$f(I(t))$$

Linear Chain (Gamma Chain)

$$\sum_i f(I_i(t))$$


IDE-based


$$\int^t f(I(\tau))d\tau$$

Metapopulation & Mobility

Graph<Model(n)> or Model(n, m)



// Nonpharmaceutical interventions

...

// Numerical solver

...

// Simulation

...

// Post-analysis

...

// Example calling the above modules, initialization, parameter setting

...

What are unit tests and why should I care?



- *Unit tests* test individual units / separated modules to validate expected behavior
- *Mocking* is used to separate modules from the full modeling stack
- Unit and more general software test do ensure error-free software but minimize room for errors

```
TEST(TestParameterSet, set)
{
    mio::ParameterSet<IntParam1, DoubleParam, IntParam2> params;
    params.set<IntParam1>(3);
    ASSERT_EQ(params.get<IntParam1>(), 3);
}
```

What are unit tests and why should I care?



- *Unit tests* test individual units / separated modules to validate expected behavior
- *Mocking* is used to separate modules from the full modeling stack
- Unit and more general software test do ensure error-free software but minimize room for errors

```
TEST(TestOdeSecir, simulateDefault)
{
    double t0    = 0;
    double tmax = 1;
    double dt    = 0.1;

    mio::osecir::Model<double> model(1);
    mio::TimeSeries<double> result = mio::simulate<double>(t0, tmax, dt, model);

    EXPECT_NEAR(result.get_last_time(), tmax, 1e-10);
}
```

What are unit tests and why should I care?

- *Unit tests* test individual units / separated modules to validate expected behavior
- *Mocking* is used to separate modules from the full modeling stack
- Unit and more general software test do ensure error-free software but minimize room for errors
- Code coverage represents percentage of code „covered“ with tests 

```
...
[ OK ] TestSaveResult.save_result_order (1 ms)
[ RUN ] TestSaveResult.save_percentiles_and_sums
[ OK ] TestSaveResult.save_percentiles_and_sums (11 ms)
[ RUN ] TestSaveResult.save_edges
[ OK ] TestSaveResult.save_edges (0 ms)
[-----] 5 tests from TestSaveResult (18 ms total)
[-----] 1 test from TestSaveEdges

[ RUN ] TestSaveEdges.save_edges_empty_ts
[ OK ] TestSaveEdges.save_edges_empty_ts (0 ms)
[-----] 1 test from TestSaveEdges (0 ms total)
[-----] Global test environment tear-down

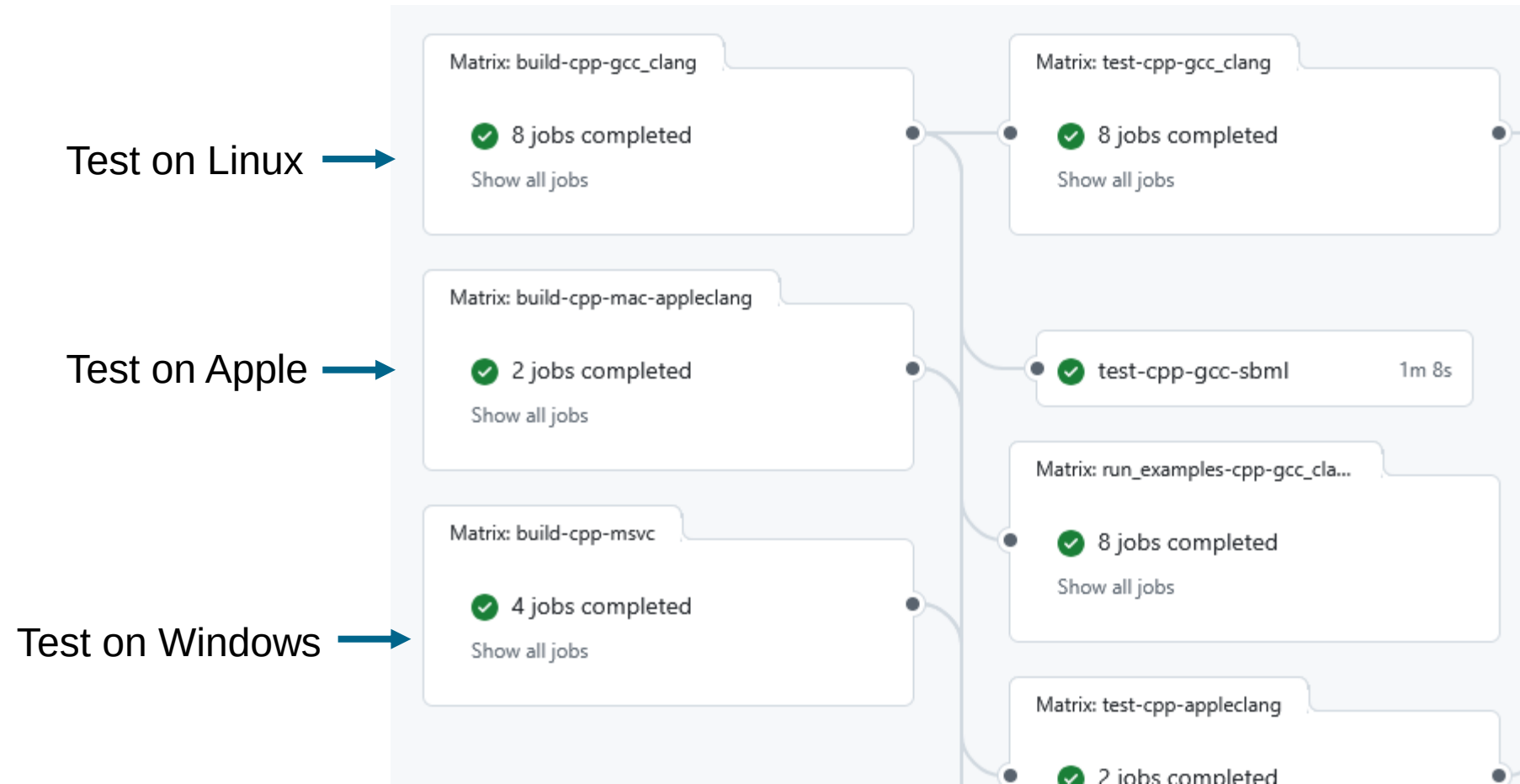
[=====] 748 tests from 152 test suites ran. (611 ms total)

[ PASSED ] 748 tests.
```

What are unit tests and why should I care?

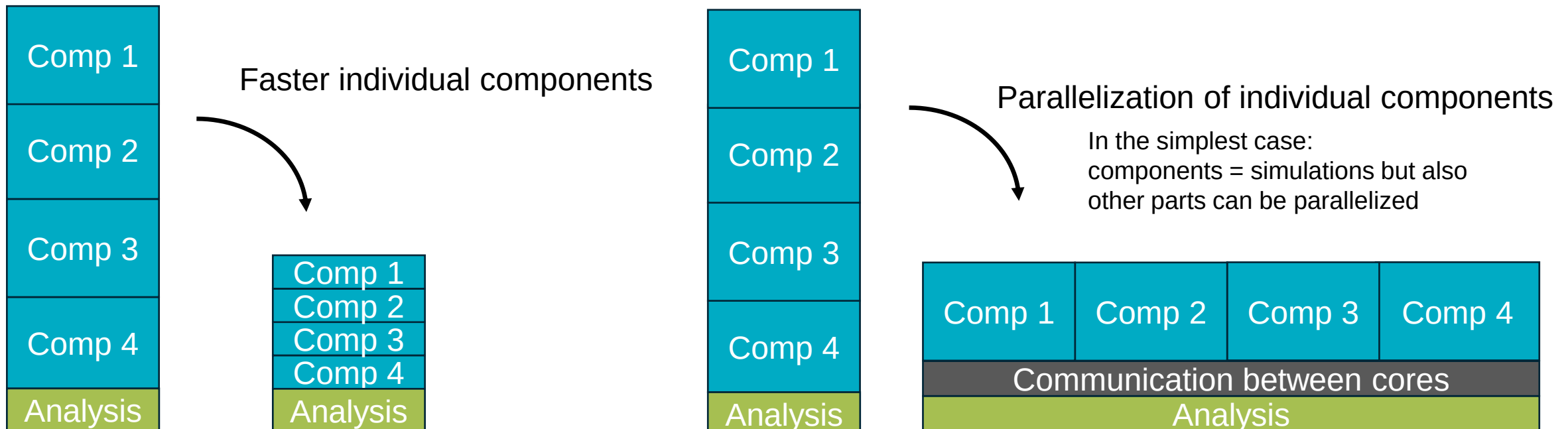


- In combination with continuous integration (CI) and, e.g., GitHub tests can also be executed automatically for many different hard- and software configurations
- Software can be considered stable after each, e. g., overnight execution or novel integration



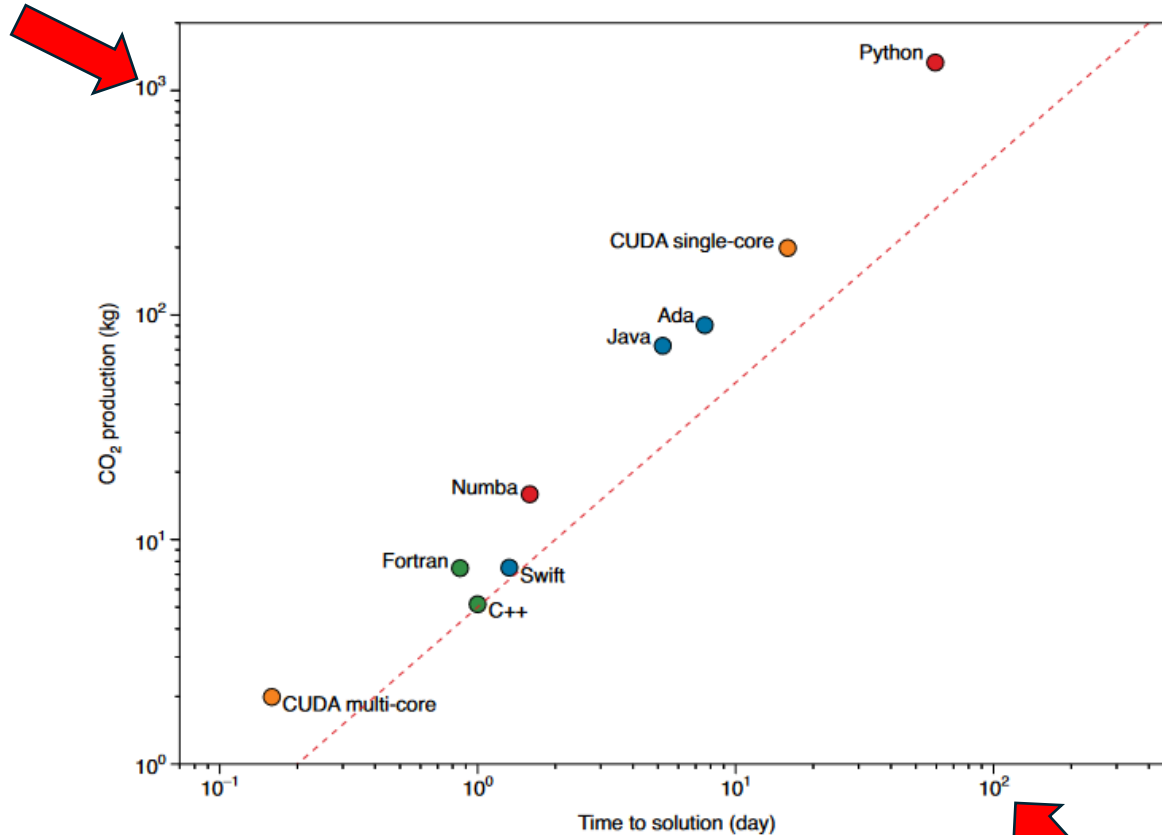
Why should I care about computational performance?

- Epidemic outbreaks need swift reactions
- Individual simulations (of seconds or minutes) rarely present a bottleneck
- Iterative development (integration of nonpharmaceutical interventions, vaccinations, new disease states) require multiple iterations → creates waiting delays
- Calibration and parameter inference require hundreds or thousands of runs → creates waiting delays
- Better performance and parallelization can substantially speedup the process between emergency and reaction (e.g., communication with decision makers)



Chances of efficiency and performance

Factor 100-1000



Factor 100

	Time
(c) C	1.00
(c) Rust	1.04
(c) C++	1.56
(c) Ada	1.85
(v) Java	1.89
(c) Chapel	2.14
(c) Go	2.83
(c) Pascal	3.02
(c) Ocaml	3.09
(v) C#	3.14
(v) Lisp	3.40
(c) Haskell	3.55
(c) Swift	4.20
(c) Fortran	4.20
(v) F#	6.30
(i) JavaScript	6.52
(i) Dart	6.67
(v) Racket	11.27
(i) Hack	26.99
(i) PHP	27.64
(v) Erlang	36.71
(i) Jruby	43.44
(i) TypeScript	46.20
(i) Ruby	59.34
(i) Perl	65.79
(i) Python	71.90
(i) Lua	82.91

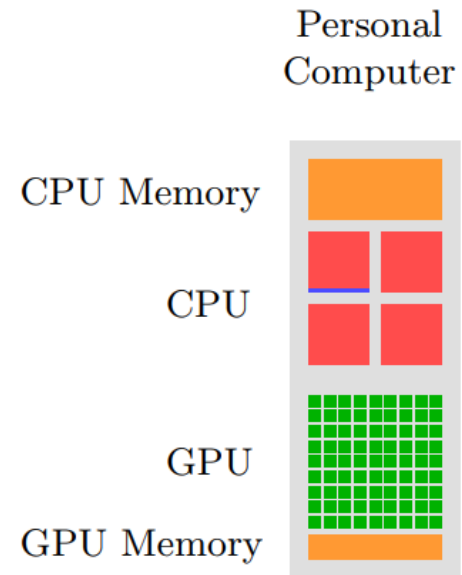
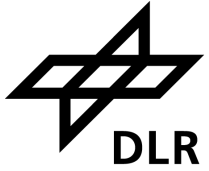
Pereira et al., SLE'17 (2017)



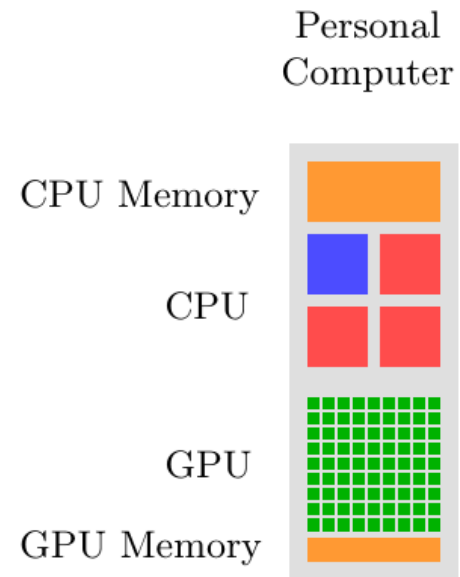
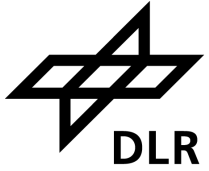
→ In hardware or other contexts: Would we build solutions for (time) critical tasks that are more than 50 times slower or use that more energy for the same task?

→ Do large-scale models in an efficient and scalable language (R, python... have other use cases but can be used as frontend)

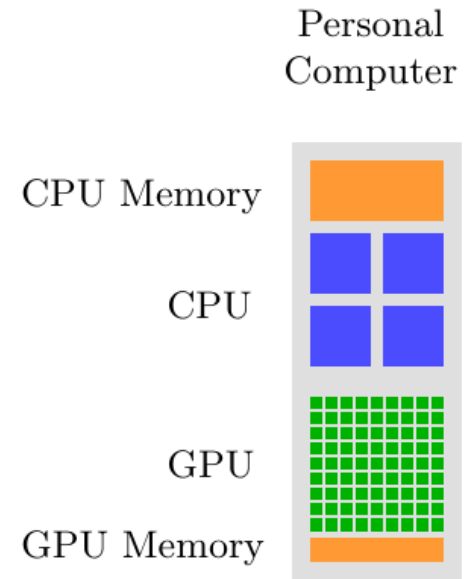
Realizing the potential with sequential execution



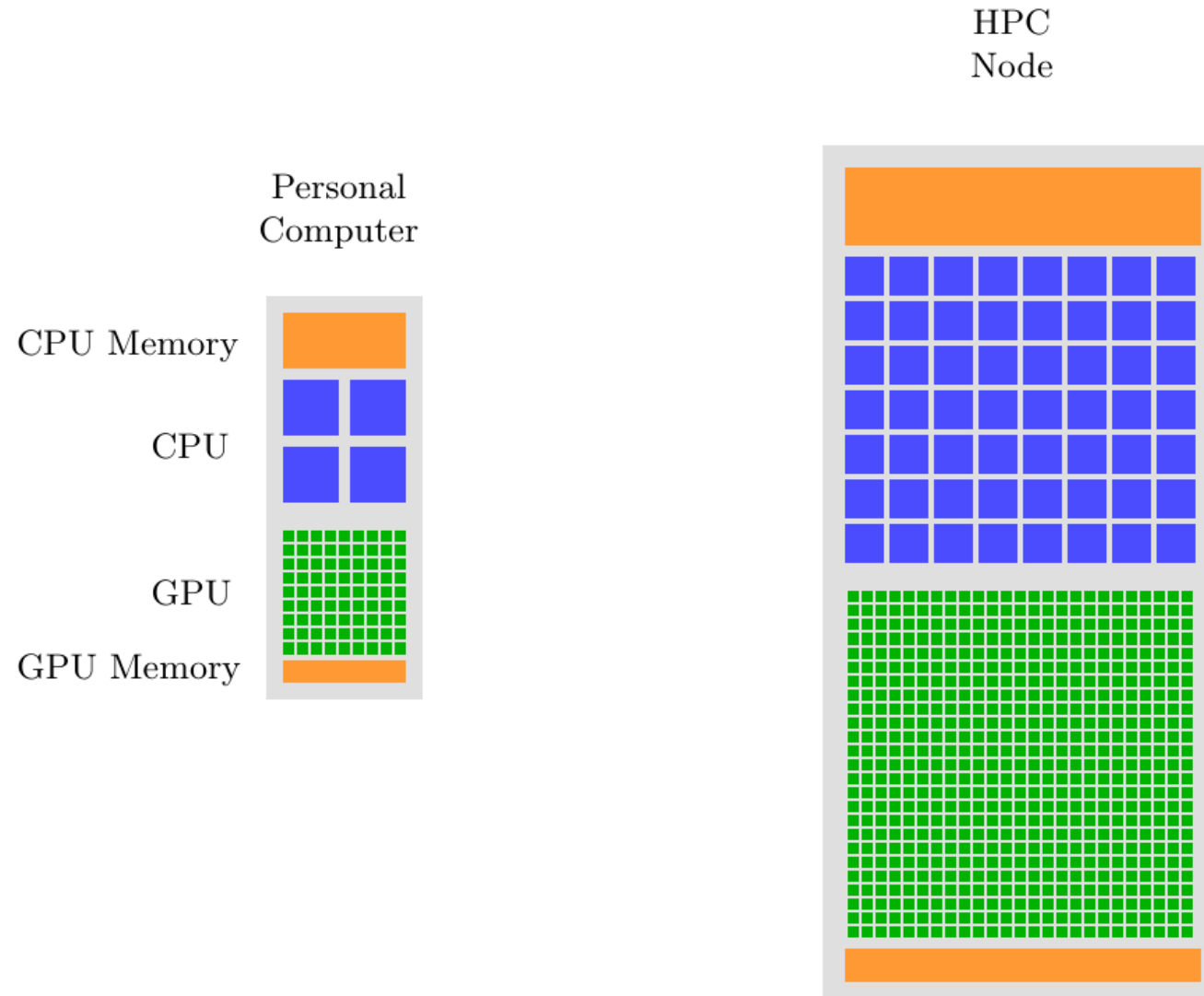
Realizing the potential with sequential execution



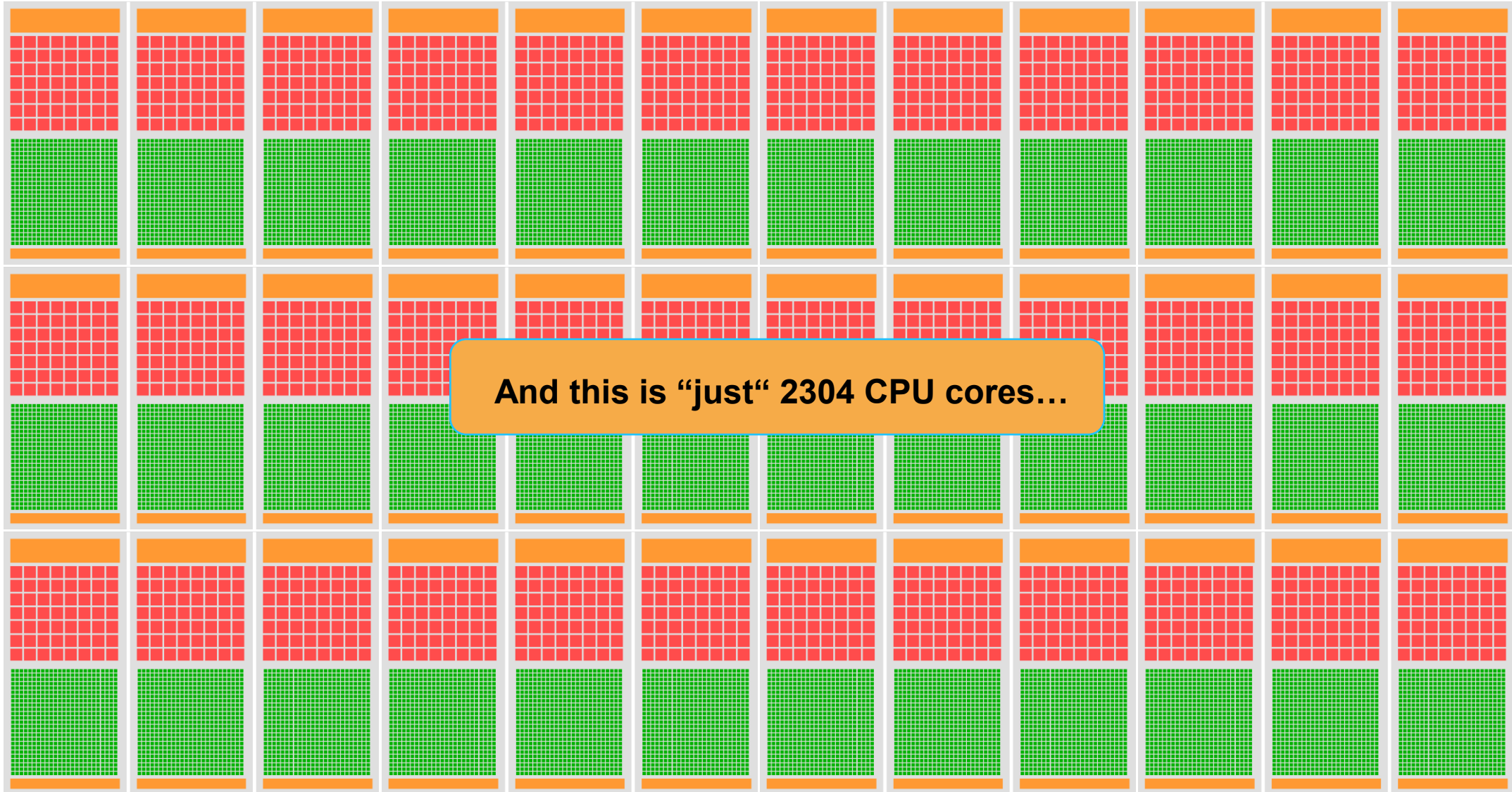
Realizing the potential with parallel computing



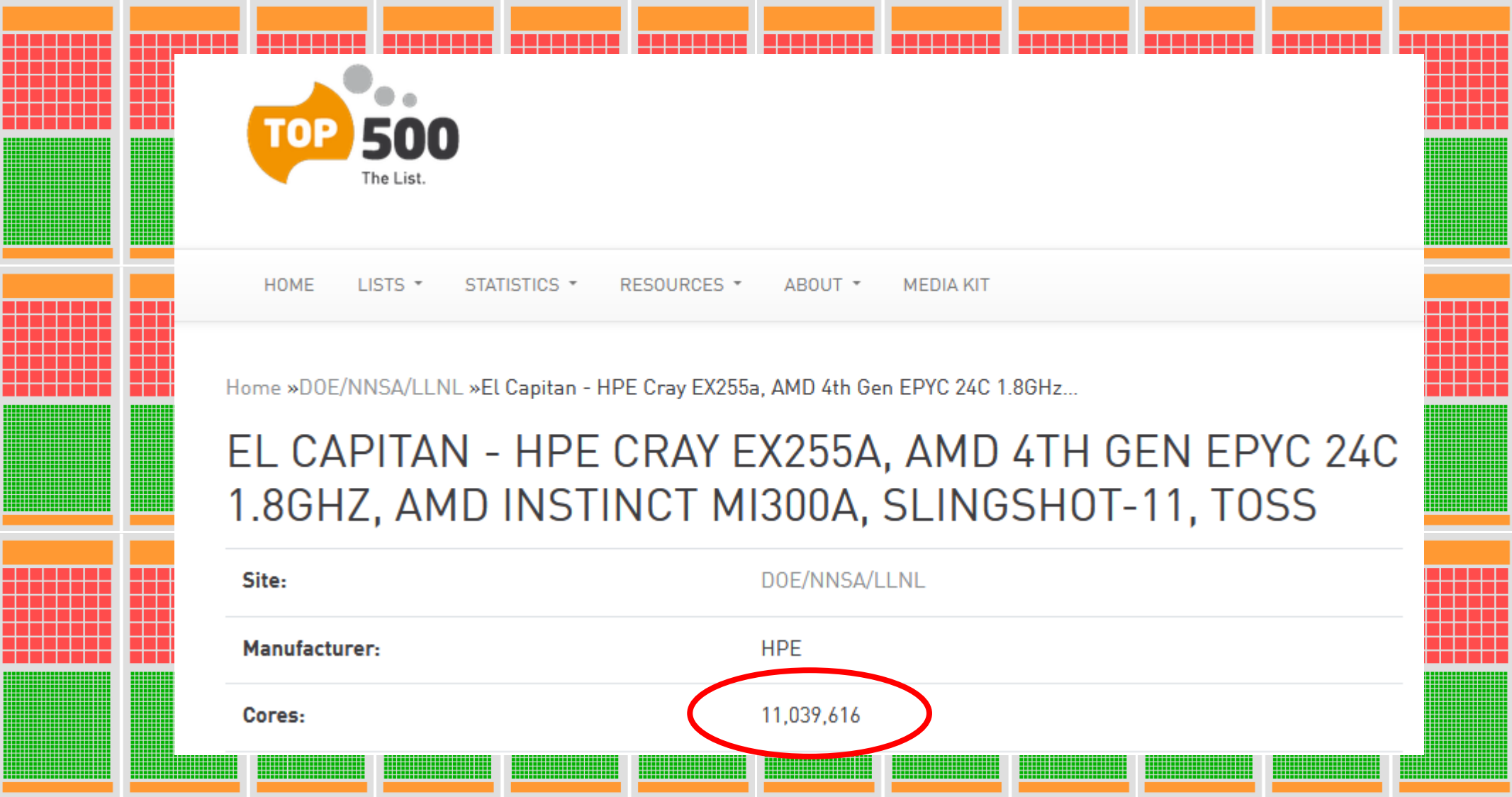
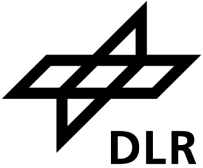
Realizing the potential with parallel computing



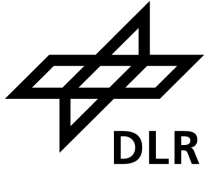
Realizing the potential with parallel computing



Realizing the potential with parallel computing



Realizing the potential with parallel computing



In computational physics, problems with

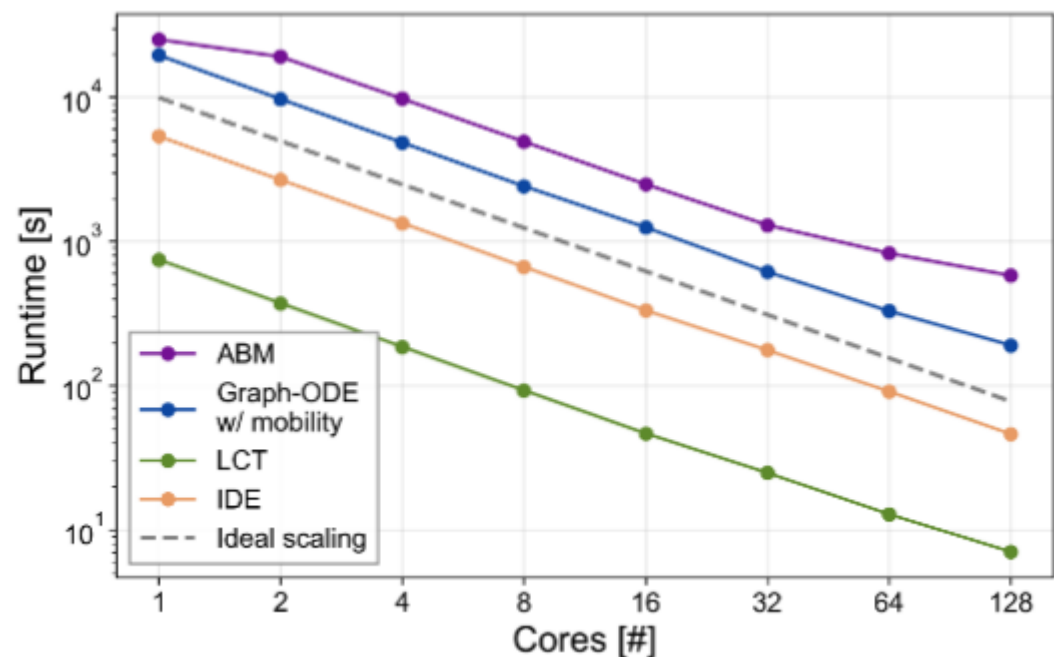
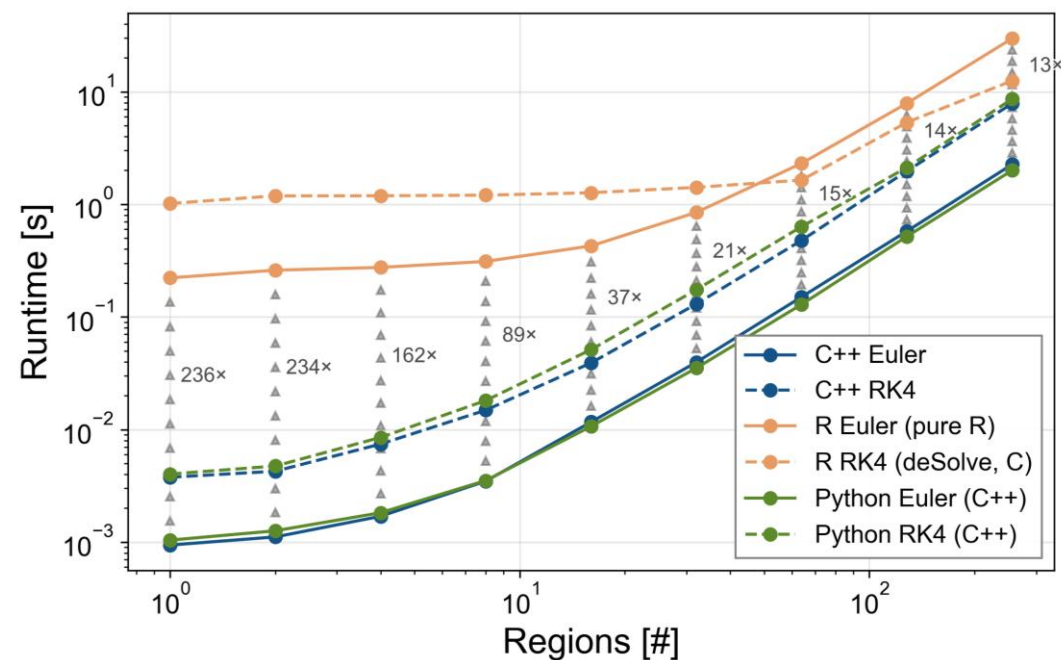
- **trillions of degrees of freedom**
- on approximately **half a million of parallel computing units**
- were solved in approximately **12 minutes**
- in **2016***

→ Why shouldn't we be able to solve some hundred regions?

→ Why shouldn't we be able to solve some million agents?

* Gmeiner et al., Journal of Computational Science, 2016.

Realizing the potential: Practical insights



Bicker et al., **MEmilio – A high performance Modular Epidemics simuLatIOn** software for multi-scale and comparative simulations of infectious disease dynamics, 2026

Thu, 09:45 - 10:00

Henrik Zunker, “A semi-continuous flow-based formulation to avoid quadratic system growth in Lagrangian metapopulation models“

Vortragssaal

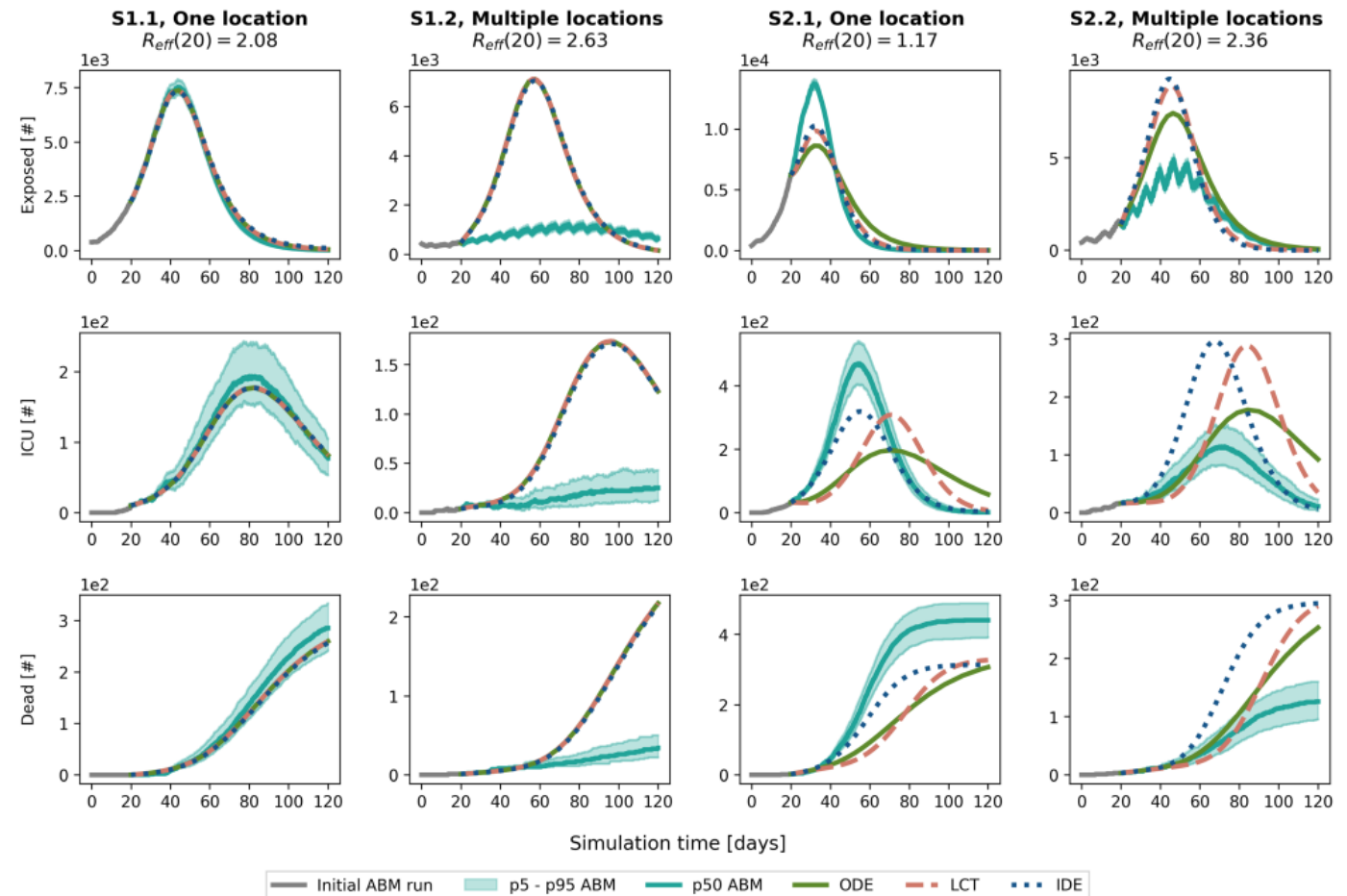
Thu, 16:15 - 16:30

Carlotta Gerstein, “Evaluating performance potentials for metapopulation models of infectious diseases“

Festsaal

Model types for infectious disease dynamics

- Distinguish between
 - the course of the disease in a model, e.g., SIR, SEIR, SECIR etc.
 - and the underlying model type, e.g., ordinary- (ODE) or integro-differential equations- (IDE) or agent-based models (ABM)
- Model types generalizations of others
- Under given conditions, outputs reduce to simpler models
- A broader overview on available model types at the end of the day



Bicker et al., MEmilio – A high performance Modular Epidemics simulation software for multi-scale and comparative simulations of infectious disease dynamics, 2026

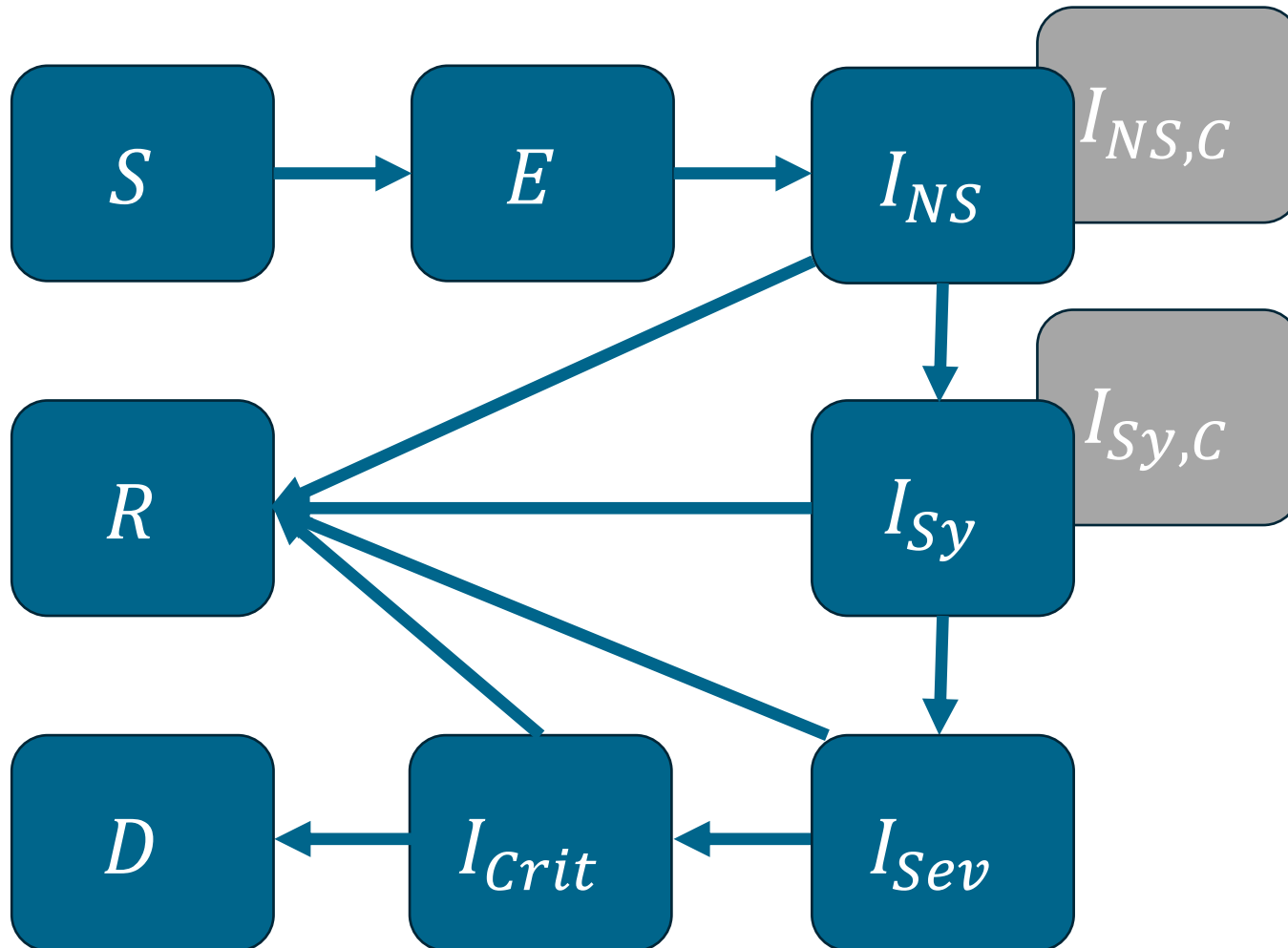
A simple model

10:00 – 10:45

A simple ODE-SEIR model

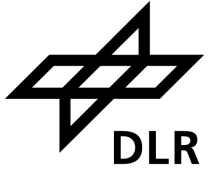
- We start by considering a „simple“ model without
 - demographic stratification
 - spatial resolution and mobility
 - different contact locations and interventions
- The above features will be introduced step-by-step over the day
- We consider a model based on ordinary differential equations (ODEs)
but many features / modules can also be combined with more complex models
 - Intended for (early) SARS-CoV-2, the model allows for
 - Presymptomatic transmission
 - Asymptomatic infection
 - Hospitalization, Intensive Care and Death
 - The model has 10 compartments from which two are for positively tested cases not used here

A simple ODE-SECIIR model: infection states



- S : Susceptible
- E : Exposed to the virus (transmitted, not infectious)
- I_{NS} : Infectious, nonsymptomatic [sometimes also: Carrier]
- I_{Sy} : Infectious, symptomatic
- I_{Sev} : Severely infected (hospitalized and well isolated)
- I_{Crit} : Critically infected (intensive care and well isolated)
- D : Deceased
- R : Recovered and immune

A simple ODE-SEIR model: parameters and force of inf.



- We introduce $\xi_{I_{NS}}$ and $\xi_{I_{Sy}}$ as the fractions of nonisolated individuals in nonsymptomatic and symptomatic infectious state
- With a time-dependent contact rate, the force of infection writes

$$\phi(t) \rho \frac{\xi_{I_{NS}} I_{NS}(t) + \xi_{I_{Sy}} I_{Sy}(t)}{N - D(t)}$$

- We furthermore introduce $\mu_{Z_1}^{Z_2}$ as the probability to go from state Z_1 to state Z_2
- and the average time spent in state Z as T_Z

A simple ODE-SEIR model: model equations

$$S'(t) = -\phi(t)\rho \frac{\xi_{I_{NS}} I_{NS}(t) + \xi_{I_{Sy}} I_{Sy}(t)}{N - D(t)} S(t)$$

$$E'(t) = \phi(t)\rho \frac{\xi_{I_{NS}} I_{NS}(t) + \xi_{I_{Sy}} I_{Sy}(t)}{N - D(t)} S(t) - \frac{1}{T_E} E(t)$$

$$I'_{NS}(t) = \frac{1}{T_E} E(t) - \frac{1}{T_{I_{NS}}} I_{NS}(t)$$

$$I'_{Sy}(t) = \frac{\mu_{I_{NS}}^{I_{Sy}}}{T_{I_{NS}}} I_{NS}(t) - \frac{1}{T_{I_{Sy}}} I_{Sy}(t)$$

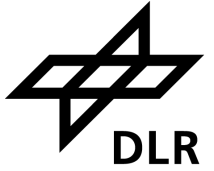
$$I'_{Sev}(t) = \frac{\mu_{I_{Sy}}^{I_{Sev}}}{T_{I_{Sy}}} I_{Sy}(t) - \frac{1}{T_{I_{Sev}}} I_{Sev}(t)$$

$$I'_{Crit}(t) = \frac{\mu_{I_{Sev}}^{I_{Crit}}}{T_{I_{Sev}}} I_{Sev}(t) - \frac{1}{T_{I_{Crit}}} I_{Crit}(t)$$

$$R'(t) = \frac{1 - \mu_{I_{NS}}^{I_{Sy}}}{T_{I_{NS}}} I_{NS}(t) + \frac{1 - \mu_{I_{Sy}}^{I_{Sev}}}{T_{I_{Sy}}} I_{Sy}(t) + \frac{1 - \mu_{I_{Sev}}^{I_{Crit}}}{T_{I_{Sev}}} I_{Sev}(t) + \frac{1 - \mu_{I_{Crit}}^D}{T_{I_{Crit}}} I_{Crit}(t)$$

$$D'(t) = \frac{\mu_{I_{Crit}}^D}{T_{I_{Crit}}} I_{Crit}(t)$$

A simple ODE-SEIR model: model equations



Hands-on: Exercise 1

***Flows* to compute new infections or hospitalizations**

11:00 – 11:30

Flows for newly infected or hospitalized



- In the classical SIR-type formulation, we compute the number of individuals currently in a compartment
- The number of, e.g., daily new infections or hospitalizations is not directly available
- A workaround is to introduce an additional equation and solve via standard numerical solver
- MEmilio by default solves all *flows* between pairs of compartments

$$y'_i(t) = \sum_j \sigma_{j,i}(t) - \sum_j \sigma_{i,j}(t)$$
$$\mathcal{F}'_{j,i}(t) = \sigma_{j,i}(y(t), t)$$

- Through a clever re-use of relevant function evaluations inside the numerical solver, this comes with almost no overhead

	<i>Standard formulation</i>	<i>Flow-aware (mio::Simulation)</i>		<i>Full Flow (mio::FlowSimulation)</i>	
N_G	Wall time (ms)	Wall time (ms)	Overhead	Wall time (ms)	Overhead
5	0.295	0.328	+11.1%	0.336	+13.8%
10	1.003	1.067	+6.4%	1.127	+12.5%
20	3.881	4.120	+6.1%	4.152	+7.0%

Table 2: Wall-clock times and overhead relative to the baseline for the SEIR model with varying numbers of age groups. We consider $N_G \in \{5, 10, 20\}$ simulating over the interval $[0, 100]$ with step size $h = 0.5$ using an RK-4 scheme.

A simple ODE-SEIR model: model equations



Hands-on: Exercise 2

Demographic stratification

11:30 – 12:15

Demographic stratification and age groups



- Often infectious diseases and NPIs hit individuals of different age differently such that we need age-resolved models. In MEmilio, an ODE model with n age groups is created through

Model model(n)

- Each local-specific contact patterns / frequencies then naturally generalize from scalar to an $n \times n$ matrix, e.g.,

$$\phi(t) = \begin{pmatrix} \phi_{1,1} & \cdots & \phi_{1,n} \\ \vdots & \ddots & \vdots \\ \phi_{n,1} & \cdots & \phi_{n,n} \end{pmatrix}$$

where $\phi_{i,j}(t)$ is the contact rate from age group i to age group j

Demographic stratification and age groups



- With the previous stratification, for the SECIR model introduced before, the force of infection onto $S_i(t)$ generalizes to

$$\rho_i \sum_{j=1}^n \phi_{i,j} \frac{\xi_{I_{NS}} I_{NS,j}(t) + \xi_{I_{Sy}} I_{Sy,j}(t)}{N_j - D_j(t)}$$

where we assume that individuals of different age groups are equally isolated

- While the terminology is *AgeGroup*, *Model(n)* could be used to stratify each model by any other dimension such as sex, income, etc.
- By default, multiple stratifications (e.g., age and sex) are only available for selected models but could be provided for others

Demographic stratification and age groups

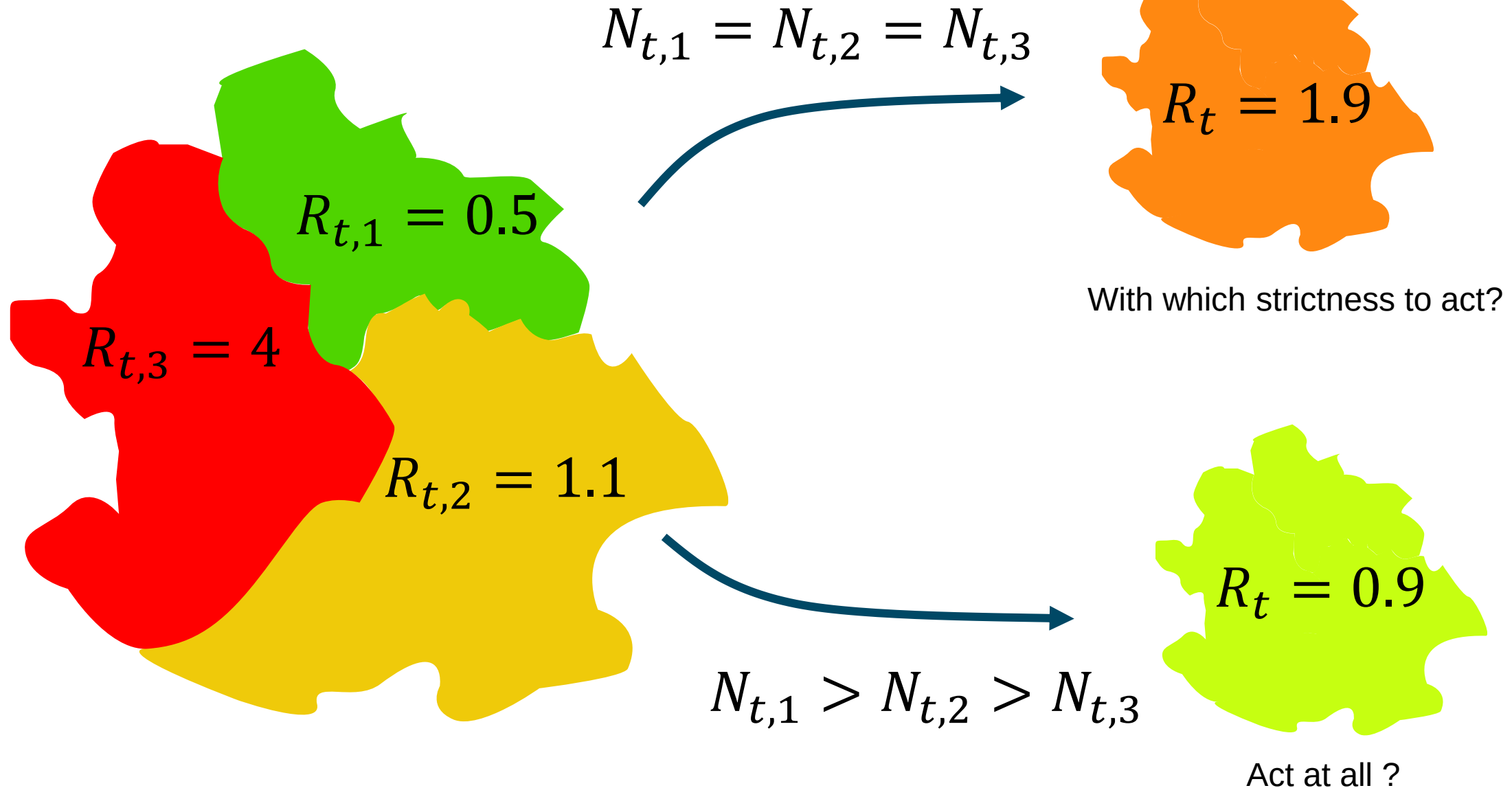


Hands-on: Exercise 5

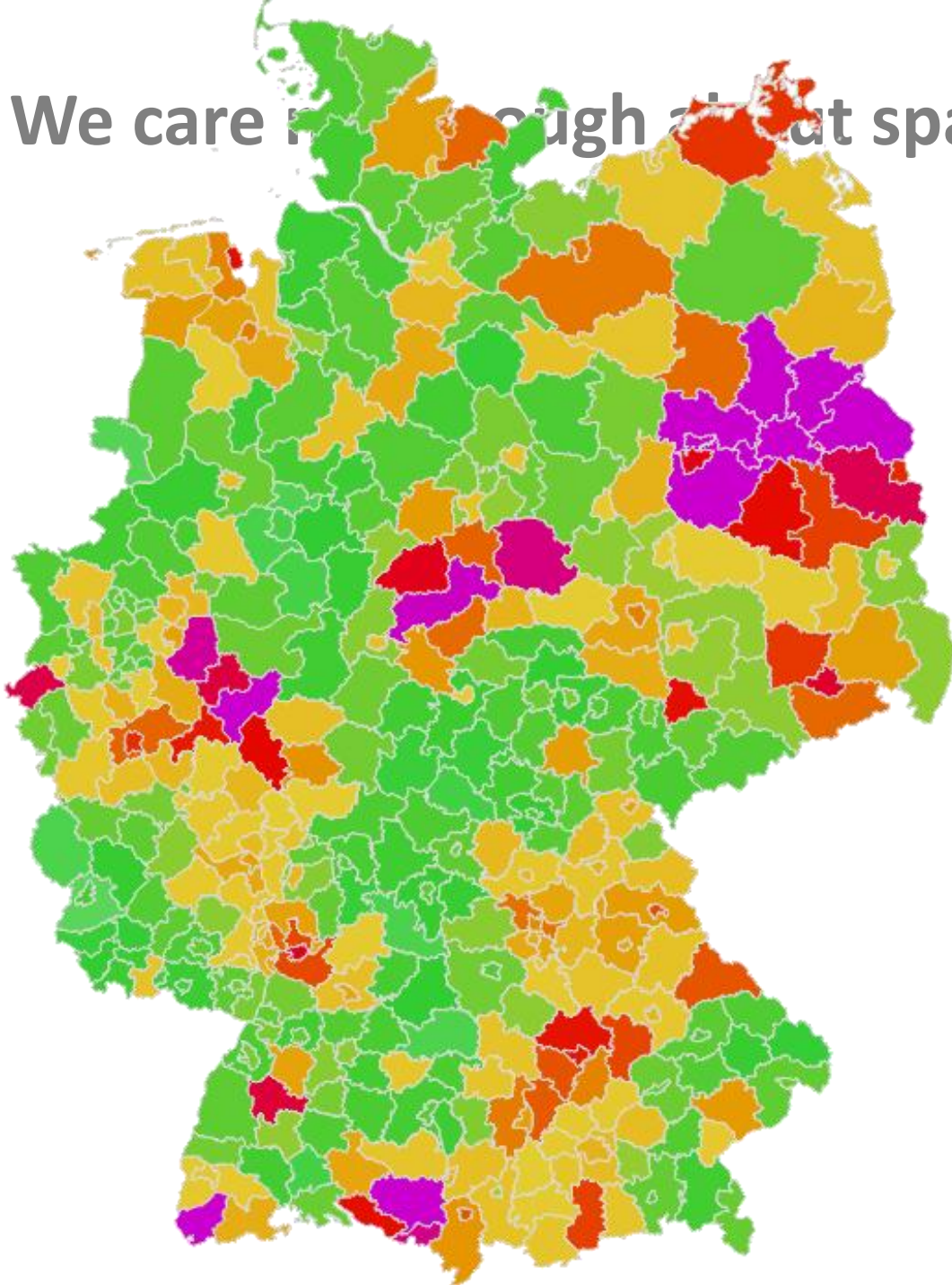
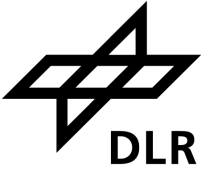
Two ways for integrating mobility and its effects

12:45 – 14:15

We care not enough about spatial resolution



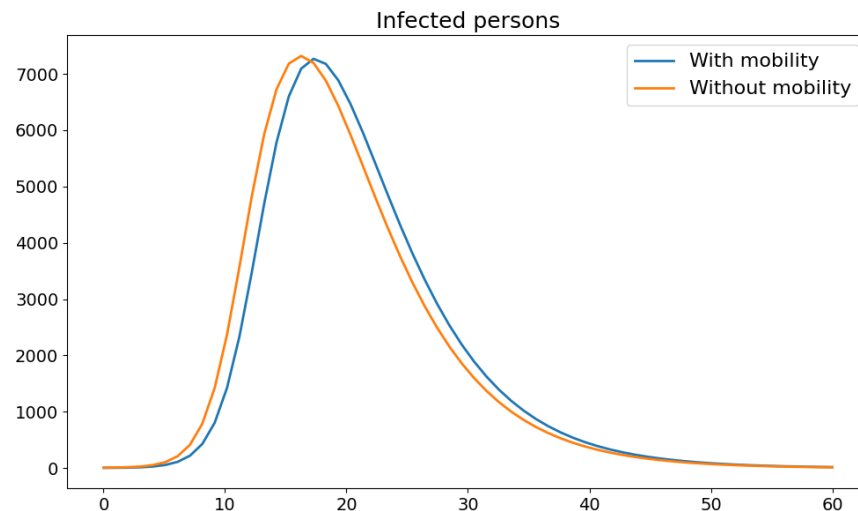
We care not enough about spatial resolution



We care not enough about spatial resolution

Spatial resolution is and mobility can become (or often also is) an important factor in infectious disease spread

- „in the end, it will arrive anyway...”



→ If it looks like that, you do not use the degrees of freedom and power in spatially resolved models

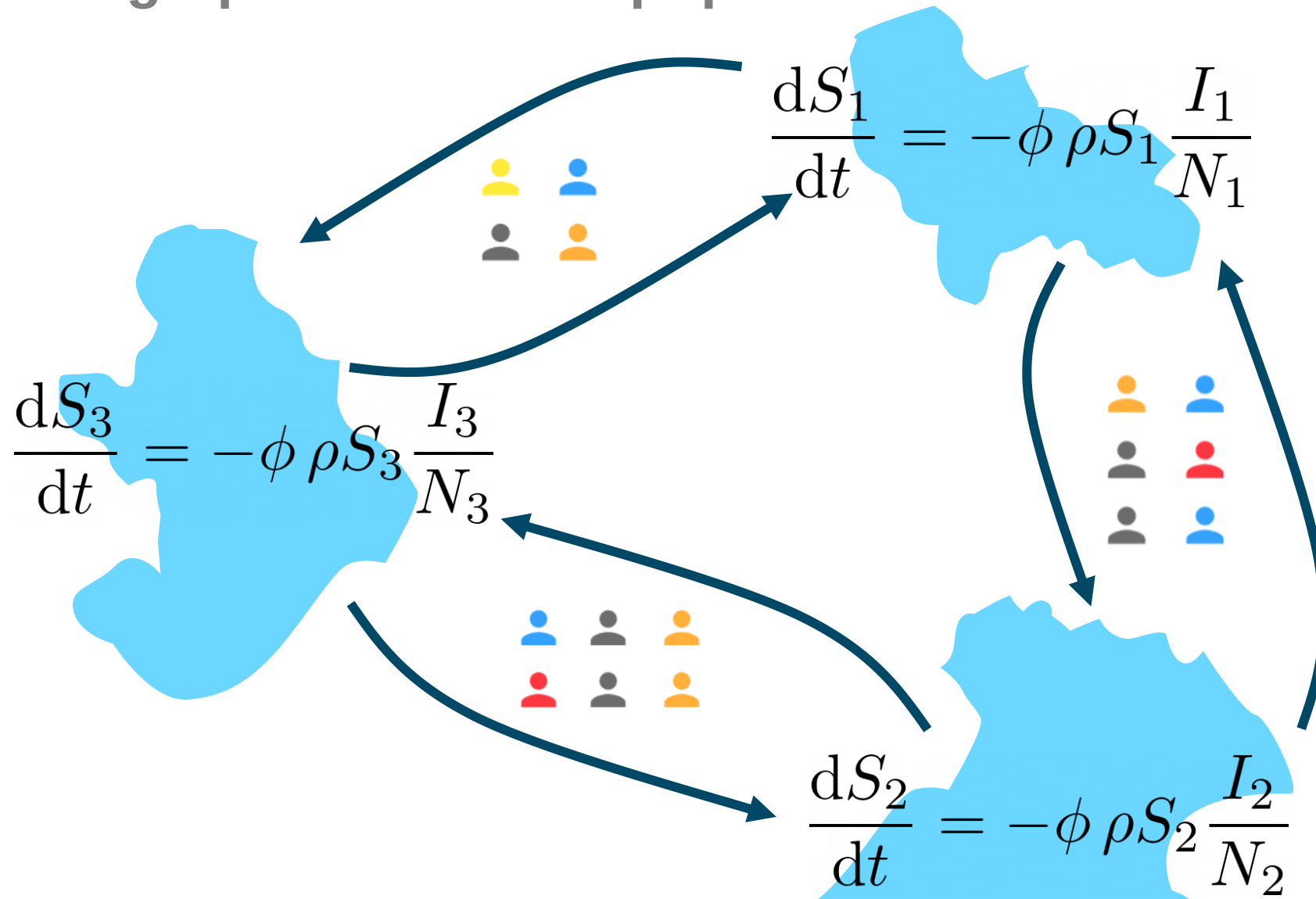
- „you cannot do metapopulation models in a small time frame“
→ well, you can...

Two ways for integrating mobility and its effects



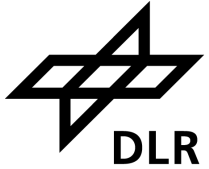
- In order to allow spatially heterogeneous spread and the effect of mobility on global disease spreading, MEmilio implements metapopulation models
 - either *graph-based*
 - or *ODE-based* (similar to age group stratification)
- The graph-based metapopulation model explicitly moves subpopulations for half a day to a destination (workplace) region after which subpopulations return
- The ODE-based metapopulation models mobility implicitly by including the effects of other regions into the force of infection term of the ODEs.

A graph-based metapopulation model



- Use a graph with regions as nodes
- One edge per, e.g., pair (infection state, age group)

An ODE-based metapopulation model



$$\frac{dS_1}{dt} = - \underbrace{\frac{1}{2}\phi_1 \rho \frac{I_1}{P_1} S_1}_{\text{No commuting phase}} - \underbrace{\frac{1}{2} \left(\phi_1 \rho \frac{I_1^r}{N_1} S_1^r + \phi_1 \rho \frac{I_1^c}{N_1} S_1^r + \rho S_1 \sum_{j \neq 1} \phi_j \frac{h_{1j}}{P_1} \frac{I_j^r + I_j^c}{N_j} \right)}_{\text{Commuting phase}}$$

$$\frac{dS_3}{dt} = - \underbrace{\frac{1}{2}\phi_3 \rho \frac{I_3}{P_3} S_3}_{\text{No commuting phase}} - \underbrace{B}_{\text{Commuting phase}}$$

$$\frac{dS_2}{dt} = - \underbrace{\frac{1}{2}\phi_2 \rho \frac{I_2}{P_2} S_2}_{\text{No commuting phase}} - \underbrace{B}_{\text{Commuting phase}}$$

h_{ij} : Commuter from region i to j

I^c : commuting infectious

I^r : remaining infectious

$$P_i = \sum_{j=1}^n h_{ij}$$

$$N_i = P_i - \sum_{j \neq i} h_{ij} + \sum_{j \neq i} h_{ji}$$

Two ways for integrating mobility and its effects

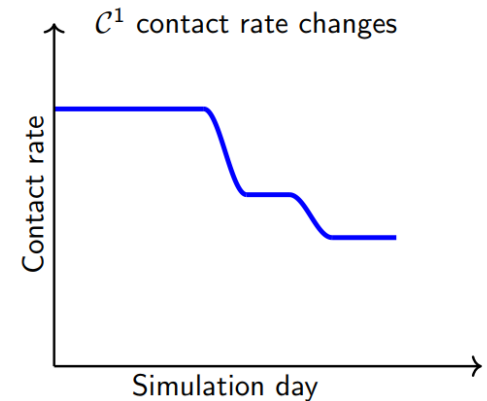
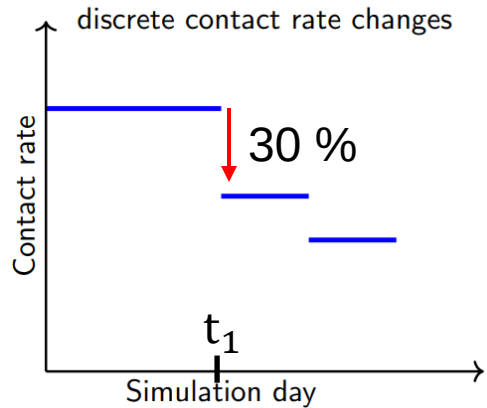


**Hands-on for metapopulation:
Exercise 7 (Graph-ODE) & Exercise 8 (ODE-based)**

Locations and nonpharmaceutical interventions

14:15 – 15:00

Contacts and nonpharmaceutical interventions



- MEmio models realize nonpharmaceutical interventions (NPIS) through *Dampings*

- A *damping* at t_1 is a relative reduction factor on the contact rate, i.e., 30 % reduction is $r = 0.3$ with

$$\phi(t_1) = (1 - r) \phi(t_1 - \delta)$$

- In order to allow a sufficiently smooth solution of the ODE, we smooth the reduction / increase over a short time frame of delta (e.g., (half) a day)

$$\phi(t) := \begin{cases} \phi, & x \leq t_1, \\ (1 - r)\phi, & x \geq t_1 + \delta, \\ \frac{1}{2}(r\phi) \cos\left(\frac{\pi}{\delta}(t - t_1)\right) + \frac{1}{2}(2\phi - r\phi), & \text{else.} \end{cases}$$

Locations, contacts, and nonpharmaceutical interventions



- To effectively realize school or venue closures and remote work through *Dampings*, we separate the contact rate

$$\phi(t) = \phi_1(t) + \phi_2(t) + \dots$$

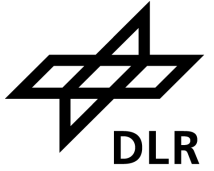
- The stratification is user-defined with a flexible list

```
class Location(IntEnum):  
    Home = 0  
    School = 1  
    Work = 2  
    Other = 3
```

where, then, $\phi_1(t)$ is the contact rate at home, $\phi_2(t)$ at school etc.

- New locations can be added in a single line (or existing can be removed correspondingly)
- Dampings can then be applied to the individual contact rates

Demographic stratification and age groups

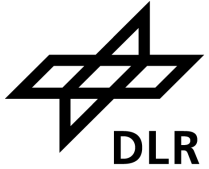


Hands-on: Exercises 3 & 10

Non-pharmaceutical interventions with dynamic conditions

15:15 – 16:00

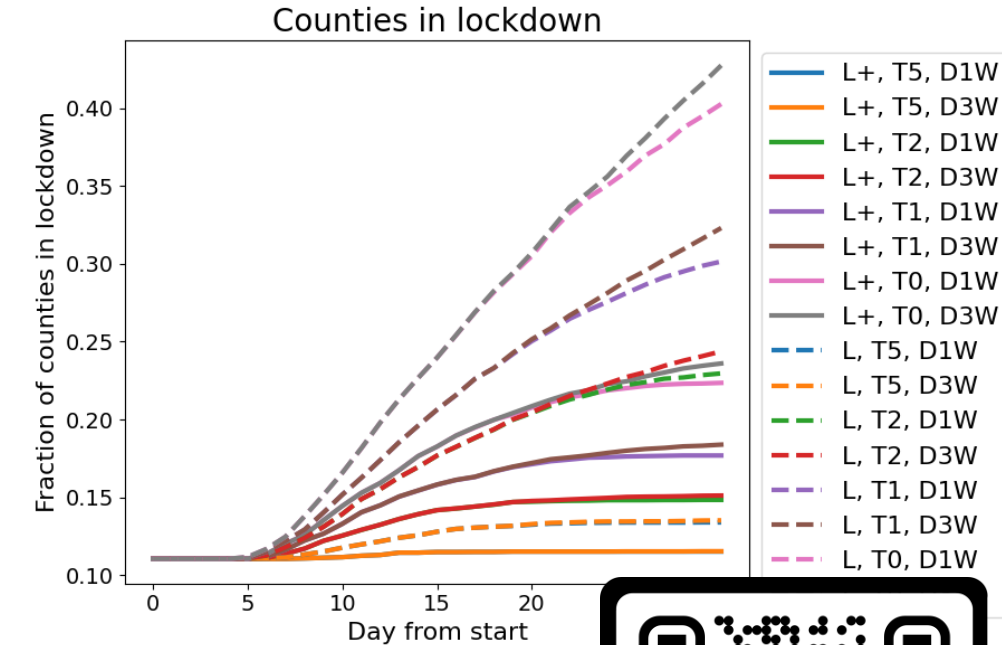
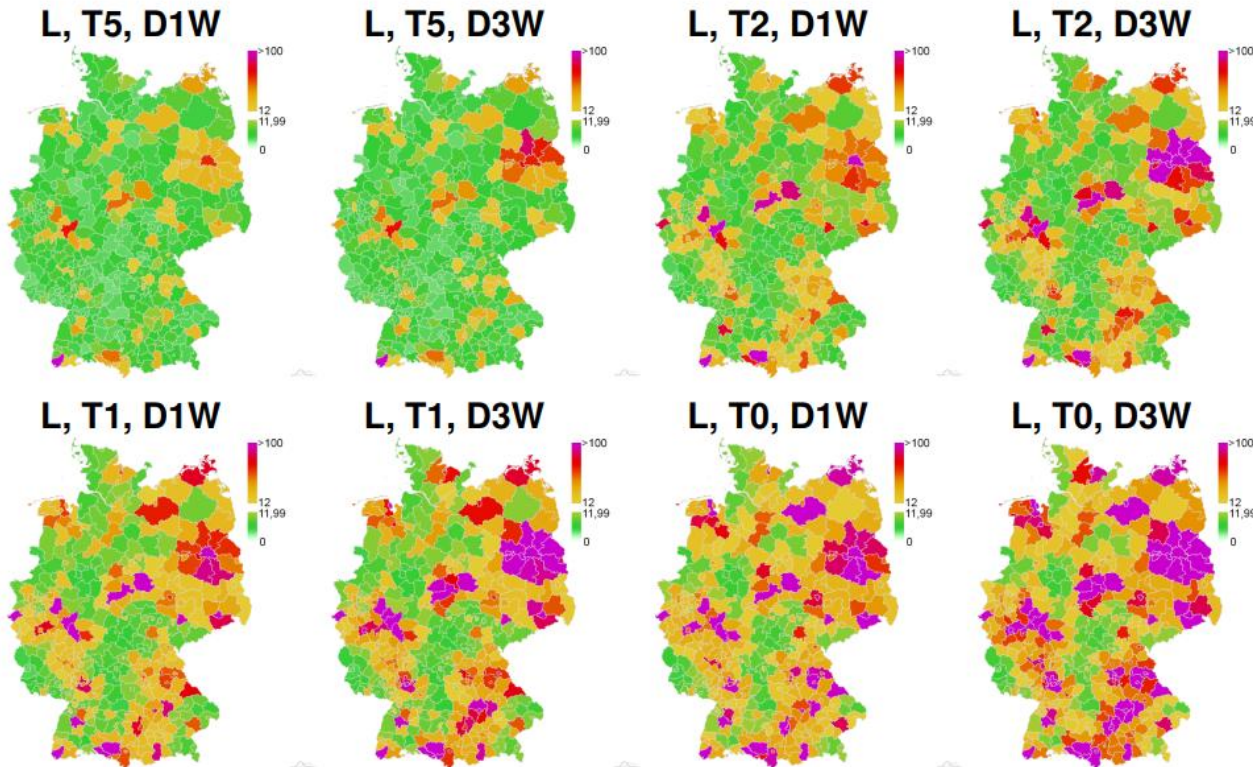
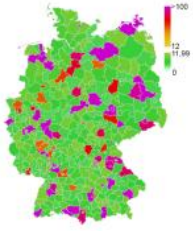
Non-pharmaceutical interventions with dynamic conditions



- Often, we would like to study the effect of an NPI that only gets activated upon a dynamical criterion such as a threshold on the incidence of, e.g., positive or hospitalized cases
- Several of MEmilio's models (SECIR, SECIRVVS, SECIRTS) pre-implement this feature as *DynamicNPIs* where we can set
 - The start and end date between which DynamicNPIs can be implemented
 - The base value e.g., (per) 100k individuals
 - The relative threshold of symptomatic infections (per base value), e.g., 100
 - The minimal implementation time, e.g., 14 days
 - A potential implementation delay realizing delayed reaction, e.g., 7 days
- The criterion will be checked in each time step for each local region and implemented / lifted automatically

Dynamic NPIs: Assessment of the “NoCovid” strategy

- T0, T1, T2, T5: No testing, testing once, twice or five times per week
- L: Lockdown, L+: Strict lockdown
- D1W, D3W: Implementations of interventions with one or three weeks delay



Kühn et al., BMC Infectious Diseases (2022)
<https://doi.org/10.1186/s12879-022-07302-9>



Simulated spread of SARS-CoV-2 cases for one initial scenario of about 18 % red zones and 8 different strategies. Median result after 30 days of simulation time.

Demographic stratification and age groups



Hands-on: Exercise 11

How to generate your own (ODE) model

16:00 – 16:15

How to generate your own (ODE) model

- MEmilio provides an efficient modeling backend but we might be interested in other courses of the disease
- Our new model generator allows to generate the corresponding model code by only providing model information in simple yaml or toml format

infection_states:

- Susceptible
- Exposed
- Infected
- Recovered

transitions:

- from: Susceptible
to: Exposed
type: infection
parameter: TransmissionProbabilityOnContact
infectious_state: Infected
- from: Exposed
to: Infected
type: linear
parameter: TimeExposed
- from: Infected
to: Recovered
type: linear
parameter: TimeInfected

parameters:

- name: TransmissionProbabilityOnContact
description: probability of getting infected from a contact
type: probability
default: 1.0
per_age_group: true
bounds: [0.0, 1.0]
- name: TimeExposed
description: the latent time in day unit
type: time
default: 5.2
per_age_group: true
bounds: [0.1, null] # null = no upper bound
- name: TimeInfected
description: the infectious time in day unit
type: time
default: 6.0
per_age_group: true
bounds: [0.1, null] # null = no upper bound

SEIR example

How to generate your own (ODE) model



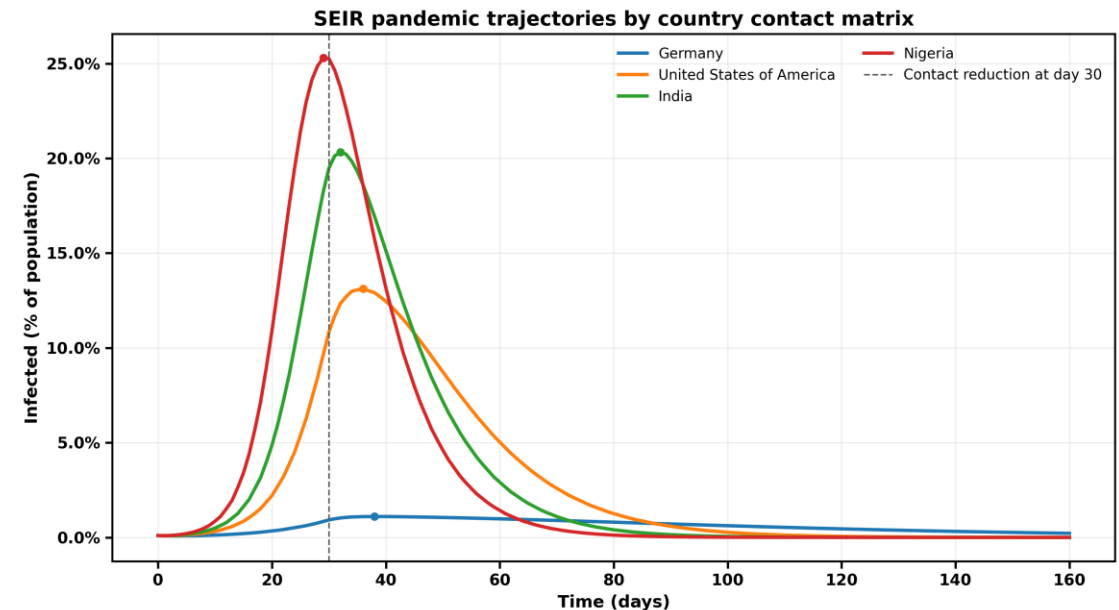
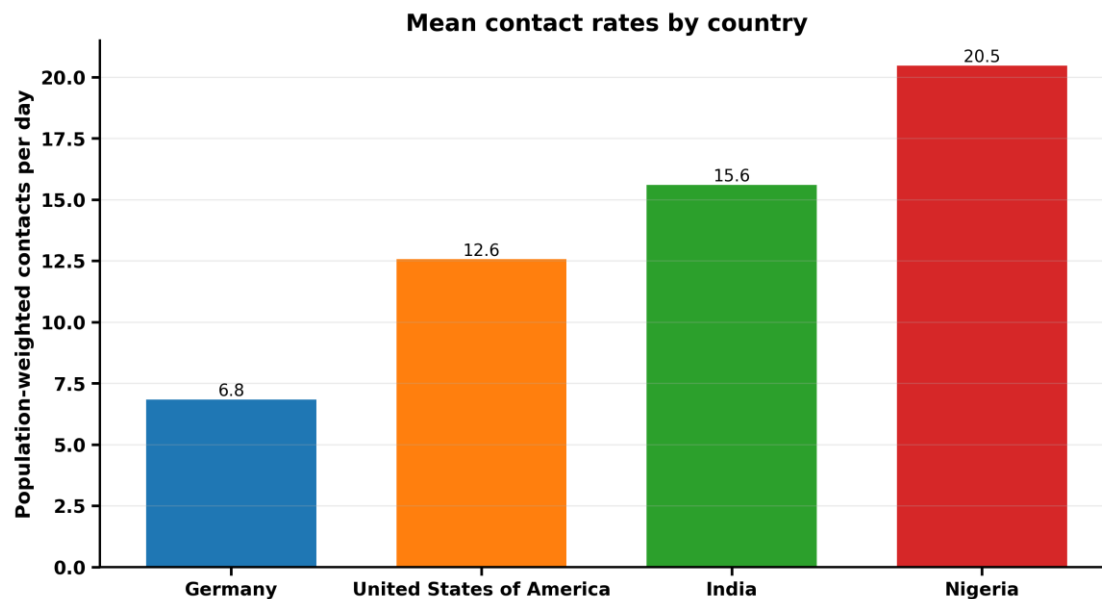
Live Demo

Country-specific simulations

16:15 – 16:20

Country-specific simulations

- MEmilio is very flexible as modeling backend but it (generally) does not import your data
- We provide a small example on how to simulate simulations specifically adapted to a country selected from the POLYMOD database



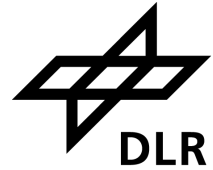
Overview of the full modeling landscape

16:20 – 16:50

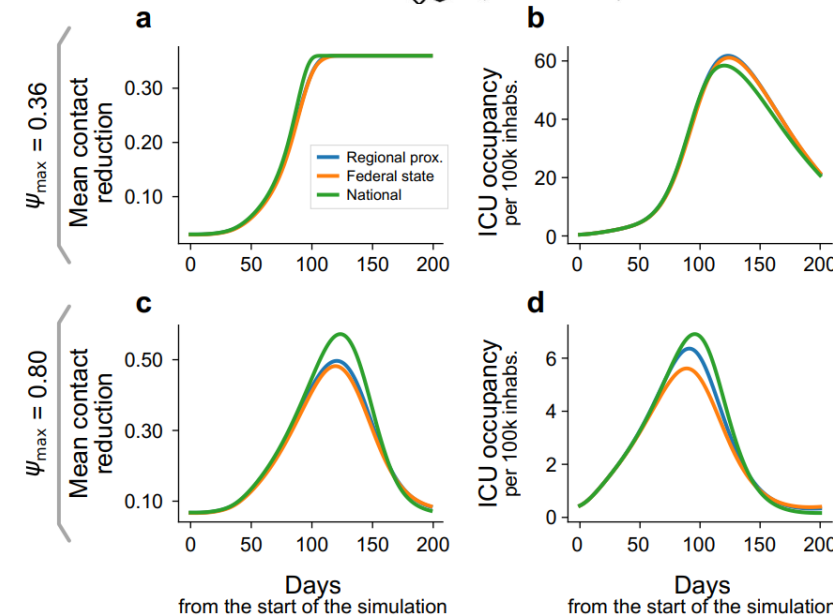
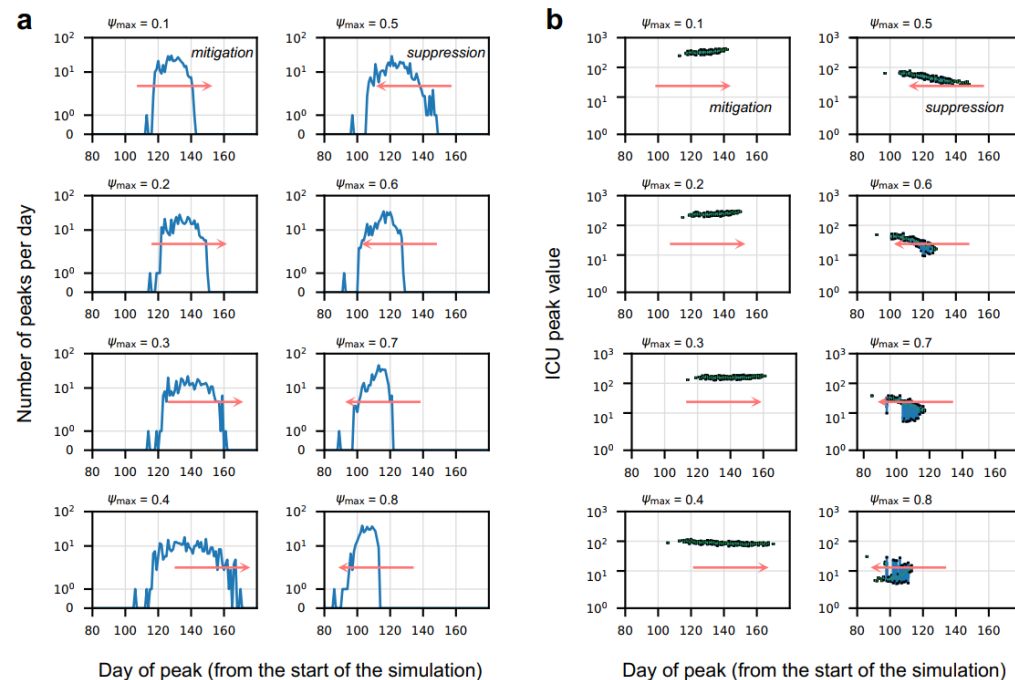
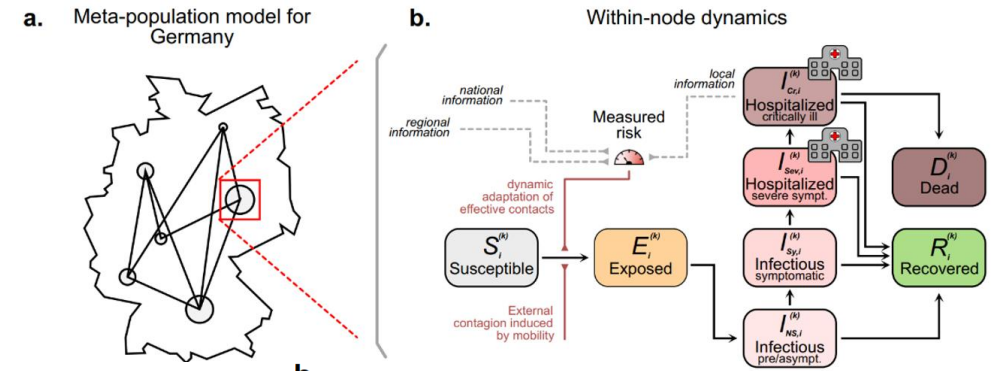
Feedback mechanism on multiple scales



Henrik Zunker



- Metapopulation models allows integration of spatially resolved human behavior
- Feedback loop based on perceived risk with memory kernel
- Nominal ICU capacity as threshold for regulation measures
- Regulation updates the effective contact rate



<https://doi.org/10.1016/j.chaos.2025.116782>

Zunker et al., Chaos, Solitons & Fractals (2025)

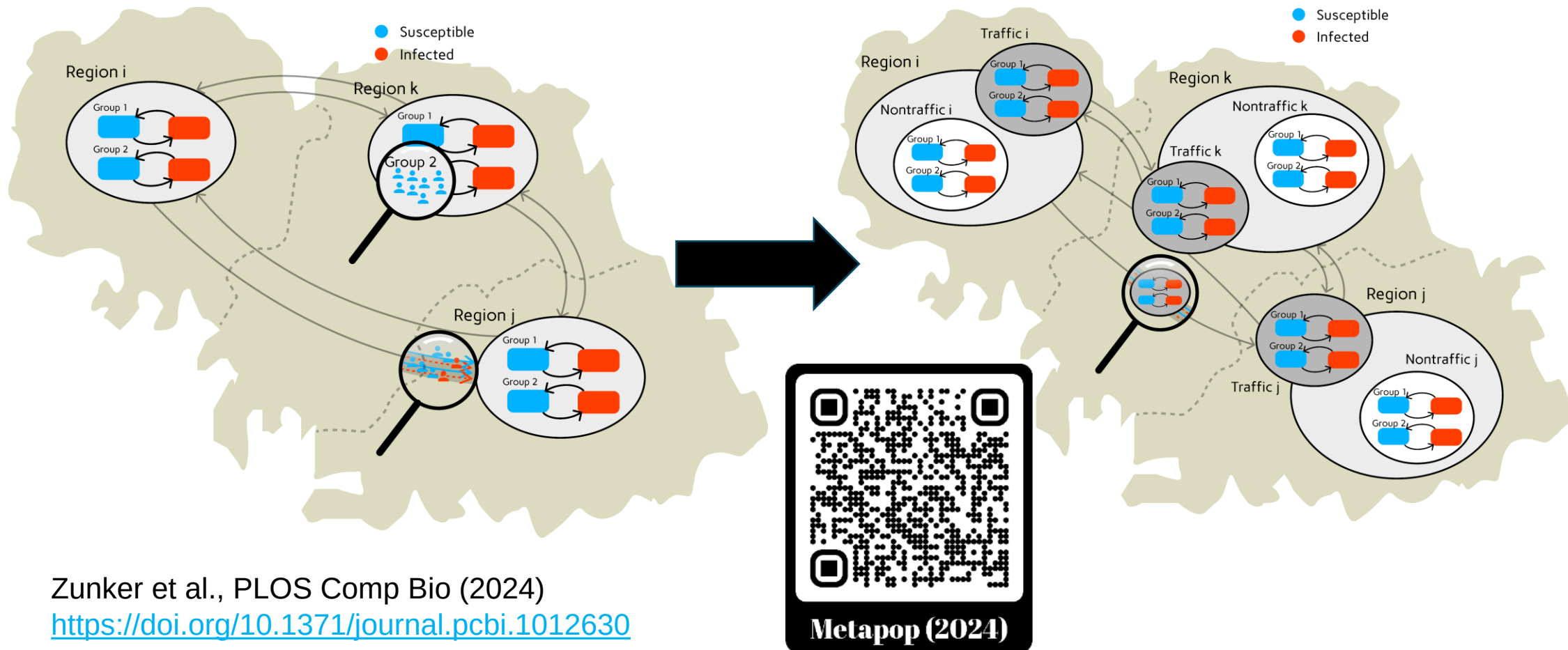
Travel time-aware metapopulation model



Henrik Zunker



- Travel times can be integrated with graph-based metapopulation model
- Allows explicit study of travel-time specific transmissions and interventions



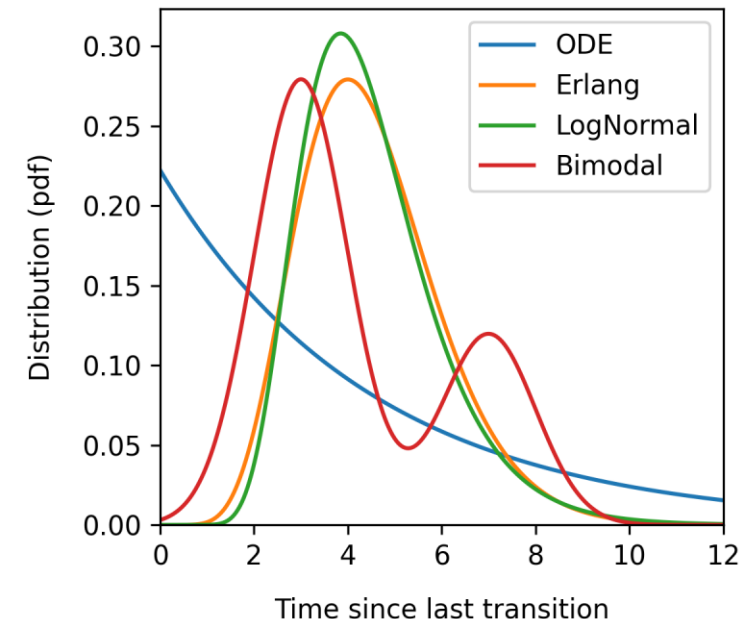
Zunker et al., PLOS Comp Bio (2024)
<https://doi.org/10.1371/journal.pcbi.1012630>

ODEs with demographic and spatial scale: Limitations

- We have seen ODE models that we can stratify
 - Discrete demographic resolution (e.g., age groups) ✓
 - Discrete spatial resolution (regions) ✓
- However, ODE models implicitly use exponential distributions for transitions between compartments

→ often unrealistic

- Linear / Gamma Chain Trick models can realize Erlang distributions (special case of gamma distributions)
- Generalized LCT approximate all positive-valued distributions
- Integro-differential equations allow arbitrary distributions



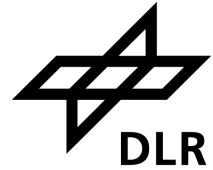
Linear / Gamma Chain Trick based models



Lena Plötzke



Anna Wendler



- Distributions have impact on epidemic peaks and timings
- Literature sometimes mentions over- / underestimation by exponential distribution
- Situation is highly complex and no general rule

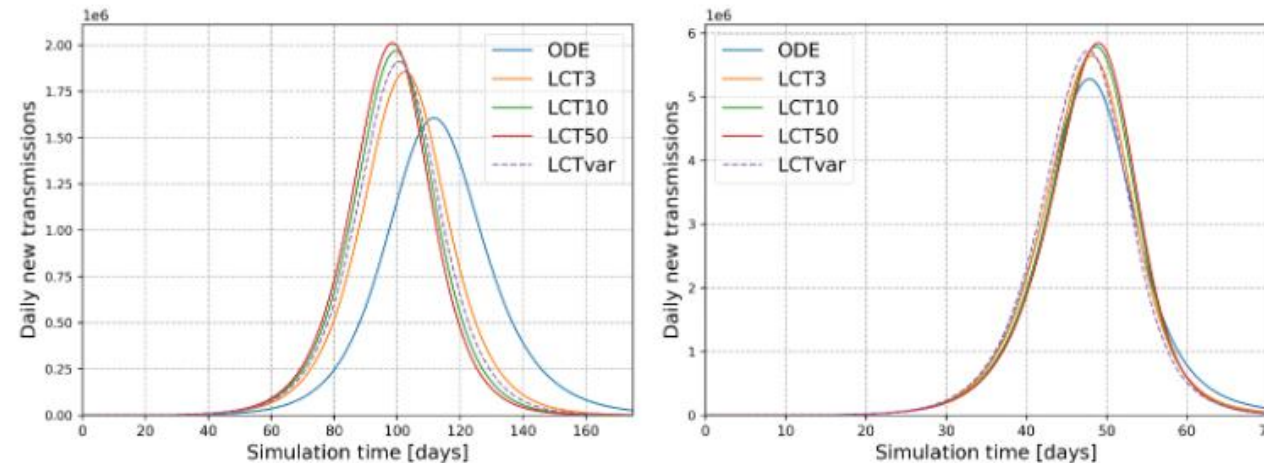
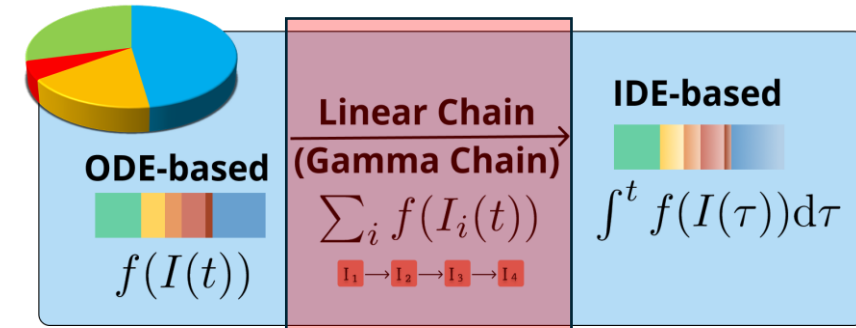


Figure 8: Daily new transmissions to compare the predicted epidemic peaks. Illustration of the predicted peaks of the daily new transmissions of LCT models with different numbers of subcompartments against a simple ODE model. The contact rate is set such that $\mathcal{R}_{\text{eff}}(0) \approx 2$ (left) or $\mathcal{R}_{\text{eff}}(0) \approx 4$ (right). Further notation as in Fig. 3.

<https://doi.org/10.1016/j.matcom.2025.07.045>

Plötzke et al., Mathematics and Computers in Simulation (2025)

Integro-differential equation-based model

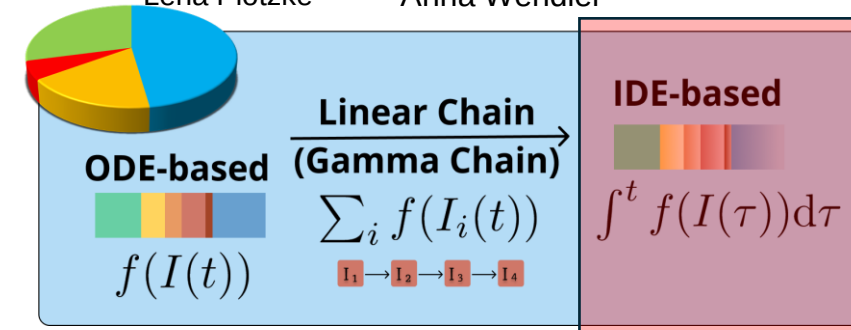
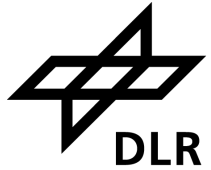
- Parameters can be made dependent on time since infection, e.g. mean infectivity
- Includes history of disease dynamics
- In particular, more realistic at change points



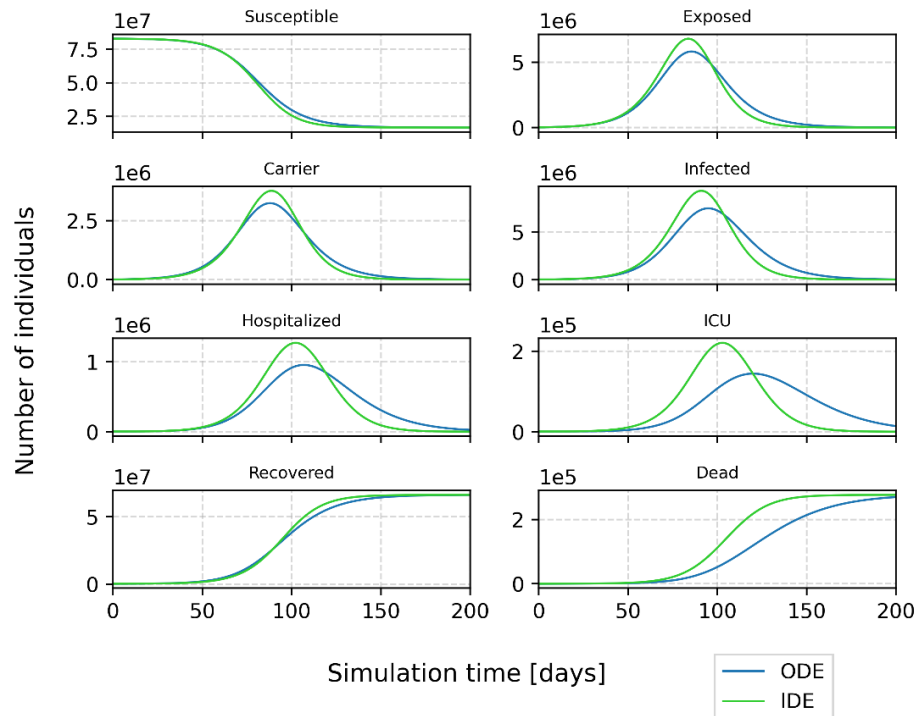
Lena Plötzke



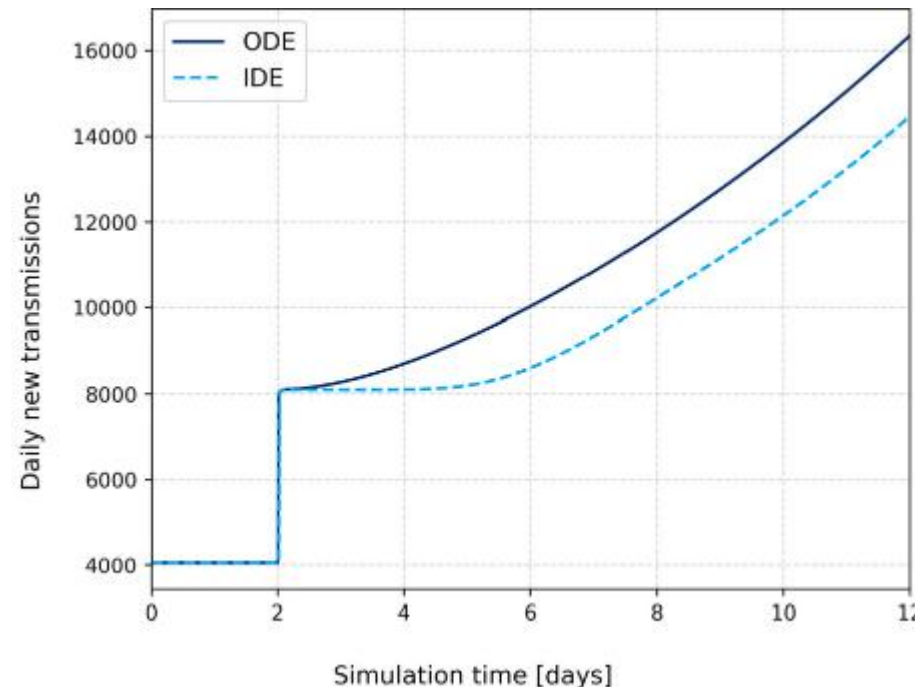
Anna Wendler



Lognormal vs exponential distributions: General



Lognormal vs exponential distributions: Change points



Agent-based modeling for individual features and behavior

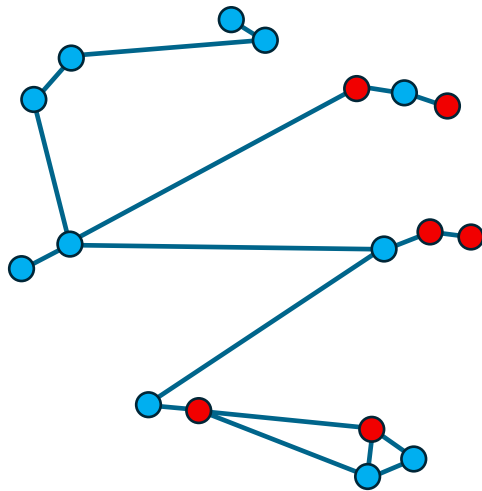


- The aforementioned models can address several questions but not one including individual behavior
→ we need an *agent-based model (ABM)*
- There is no unique understanding or complete definition to what an ABM is
- In different application domains, very different understandings or realizations exist
- While many authors try to distinguish between „agent-based models“, „individual-based models“, „cellular automata“, „network models“ etc., we provide the following inclusive understanding (which is not a complete definition)

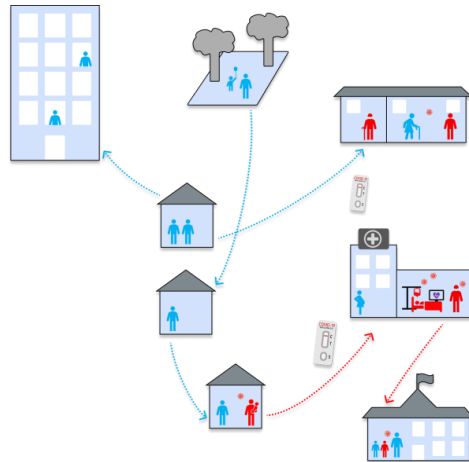
Agent-based modeling for individual features and behavior

An ABM consists of a finite (or countably infinite) number of unique **agents** $\alpha_1, \dots, \alpha_n$ and an **environment** in which and to which an agent reacts. Furthermore,

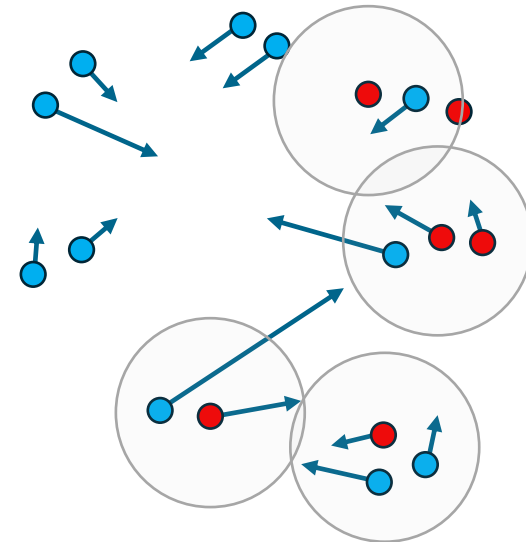
- an agent is characterized by a finite number of **features** that determine its **state**,
- an agent interacts with other agents and their joint environment according to **interaction rules**,
- the state of an agent or the environment changes through interactions or with time.



Network-based



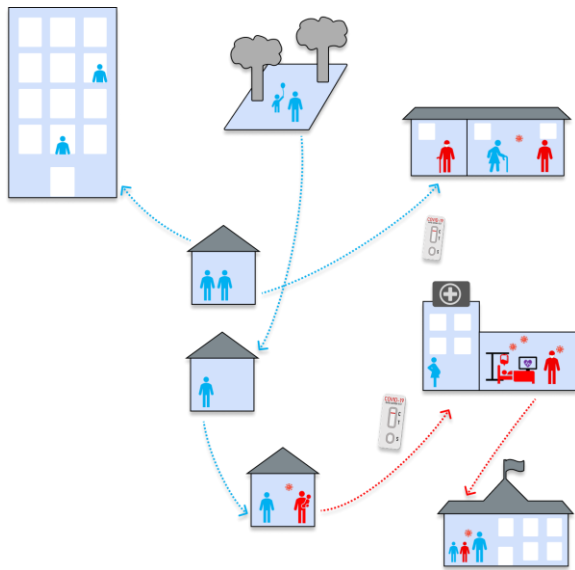
Mobility-based



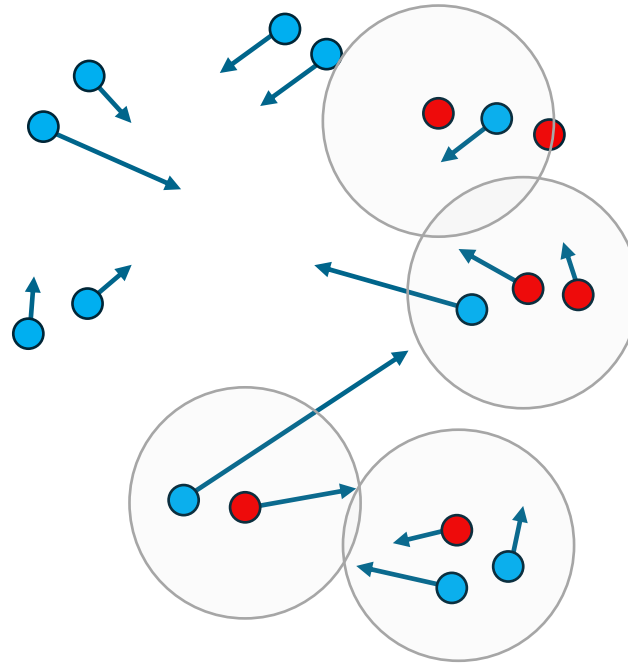
Continuous Movement-based

Agent-based modeling for individual features and behavior

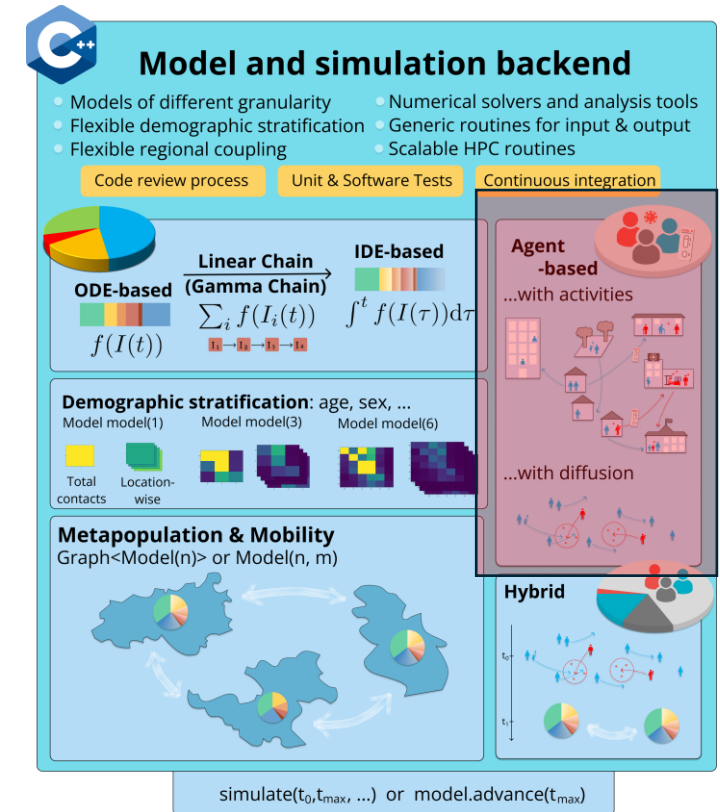
MEmlilio offers two types of agent-based models, where the mobility- or activity-based model is the default and most advanced



Mobility-based



Continuous Movement-based

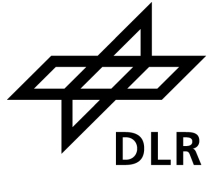


Application: Advanced testing strategies

Each individual tests with probability before conducting a trip (e.g., going to work)

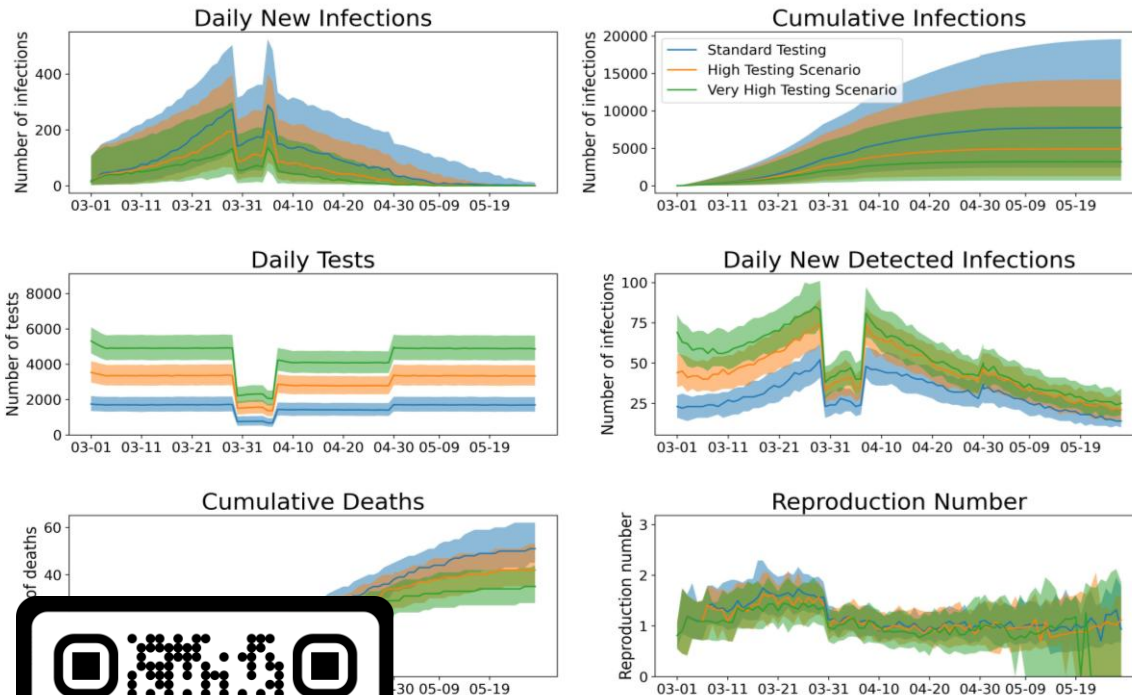


Sascha Korf

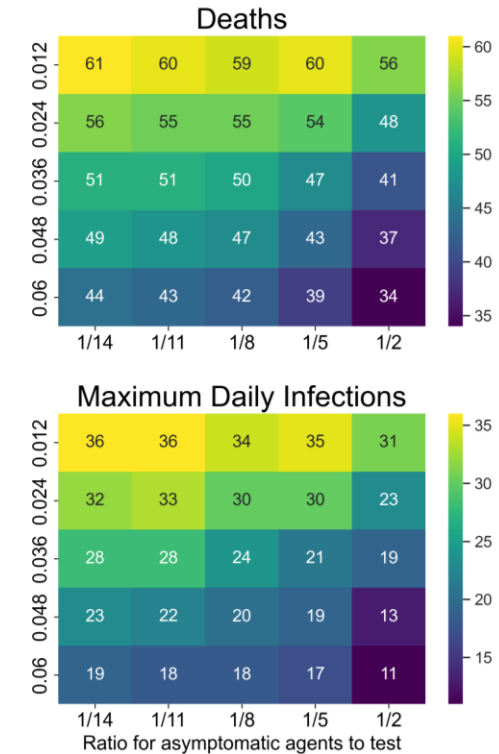
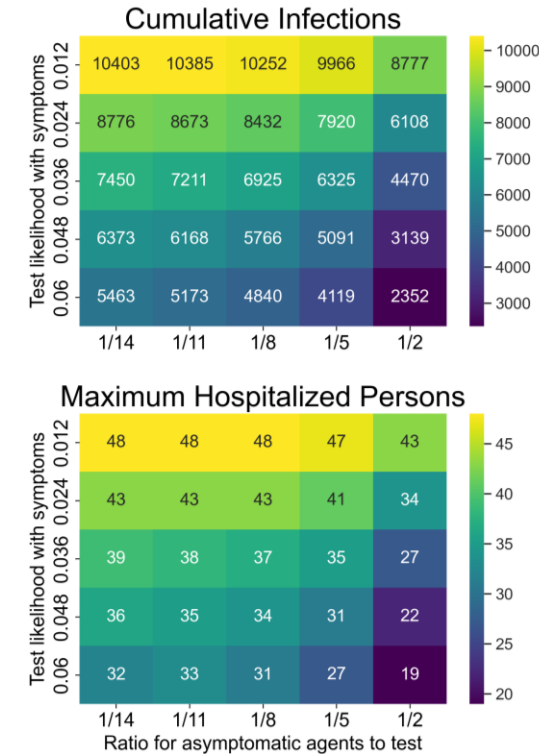


- Scenario analysis

- Symptomatic vs. general testing



More symptomatic testing



More symptom-independent testing

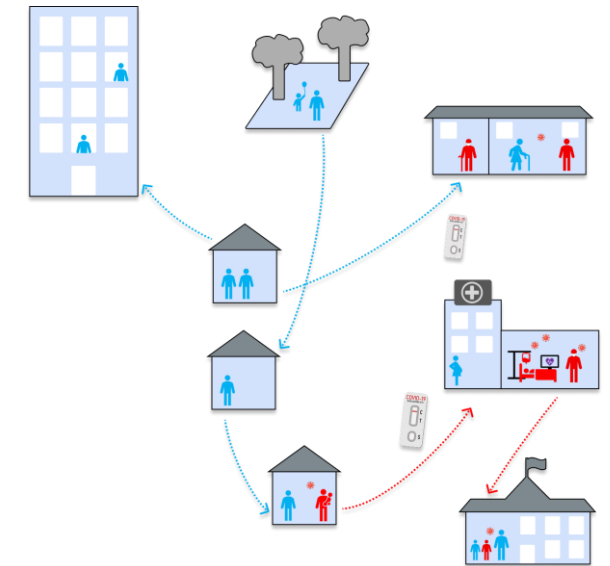
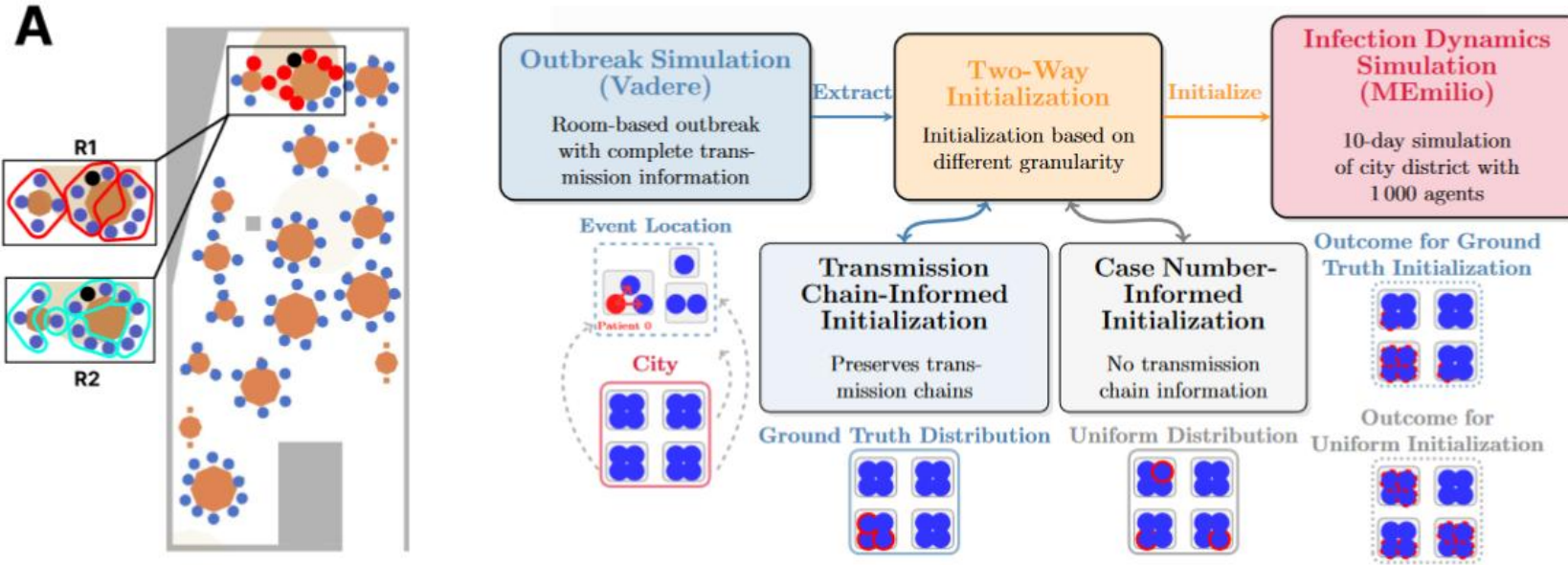


Kerkmann, Korf et al., Computers in Biology and Medicine (2025)
<https://doi.org/10.1016/j.combiomed.2025.110269>

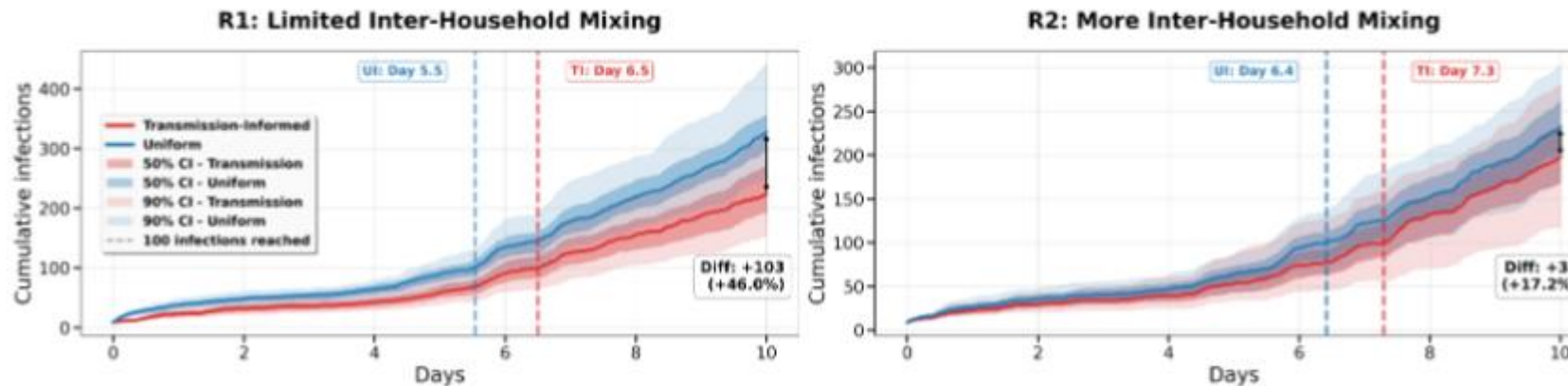
Multiple scales: Local superspreading events and global transmission chains



Sascha Korf



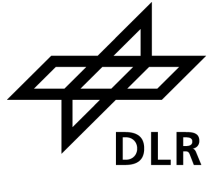
Infection Curves: Transmission-Informed vs Uniform Initialization



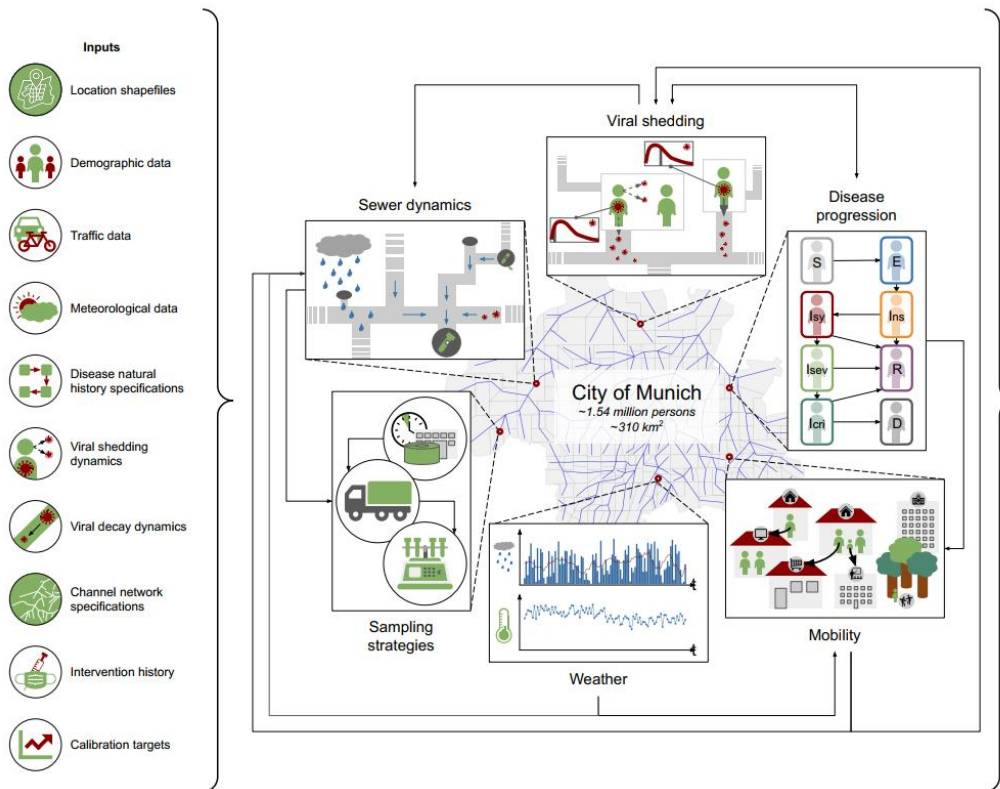
Joint ABM-Wastewater model



Julia Bicker

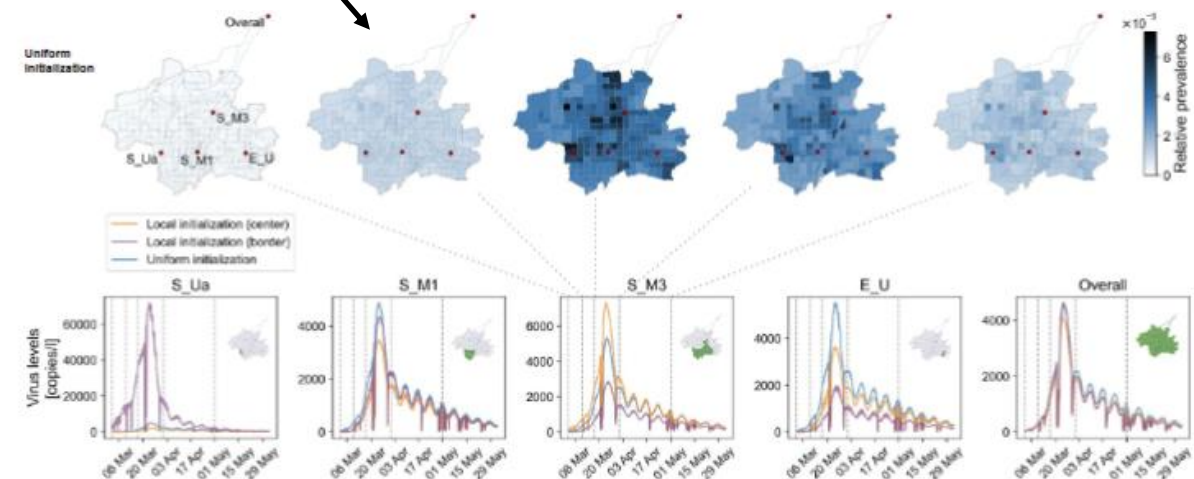
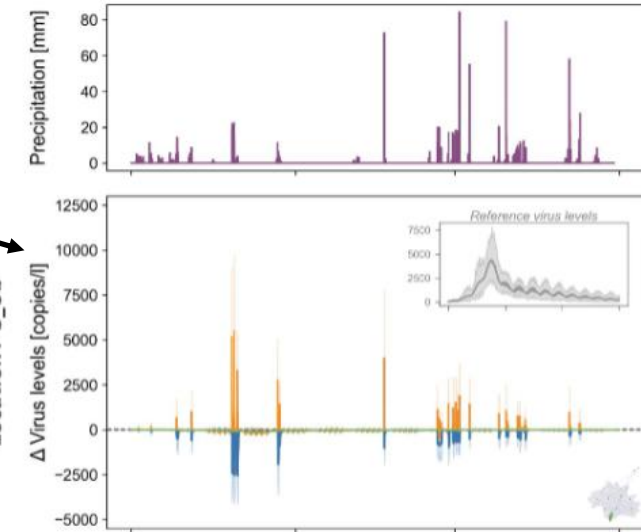


“How does can testing-based detection be improved by combination with wastewater?”



Studying the effect of different normalization strategies for rain events

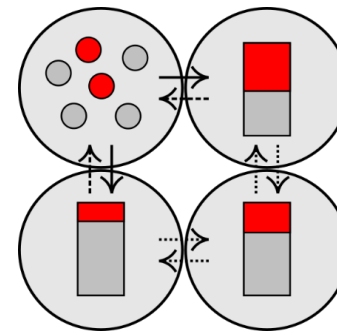
Detailed insights into population and wastewater dynamics



Bicker, Tomza, Wallrafen et al., Submitted (2025)

<https://doi.org/10.1101/2025.09.25.25336633>

Hybrid agent-metapopulation: Spatial hybridization



Julia Bicker



René Schmieding



Problem

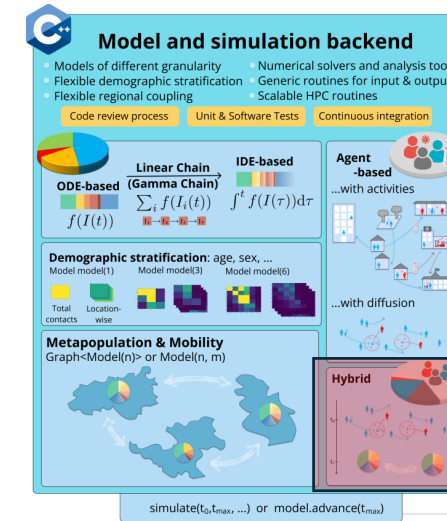
- ABMs are still more costly and need extensive data

Setting

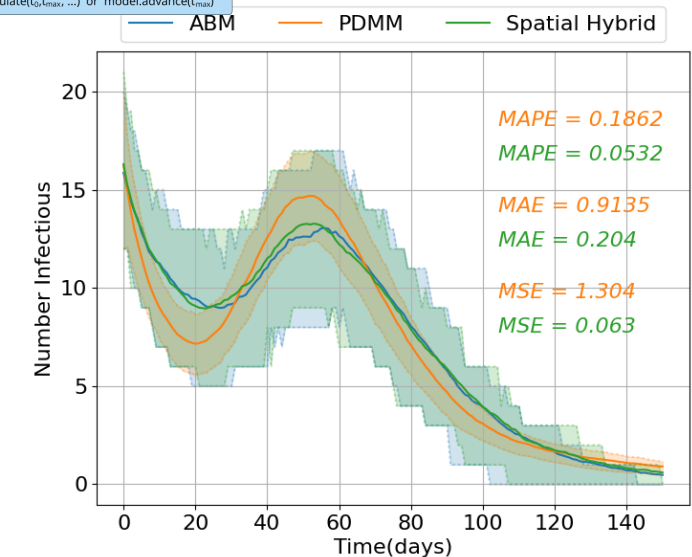
- Interest in infection spread in particular region
- Exclusive availability of data in specific region (or computational resources limited)

Solution

- Agent-based model in region of interest (focus region)
 - ODE-based models for connected regions
- Spatial-hybrid reduces runtime by **>95%**



Hybrid (2025)



AI-based on-the-fly computation



Agatha Schmidt



Henrik Zunker



Problem

- Independent of optimization, no mechanistic model can be executed hundreds of times on-the-fly

Setting

- The exploration of a large number of scenarios through an intuitive and responsive interface, e.g., a website, has much more potential than a lengthy static pdf of outcomes

Solution

- Simulate spatially resolved expert models
- Train spatially resolved AI surrogates on expert model' outcomes
- Deploy trained model to website

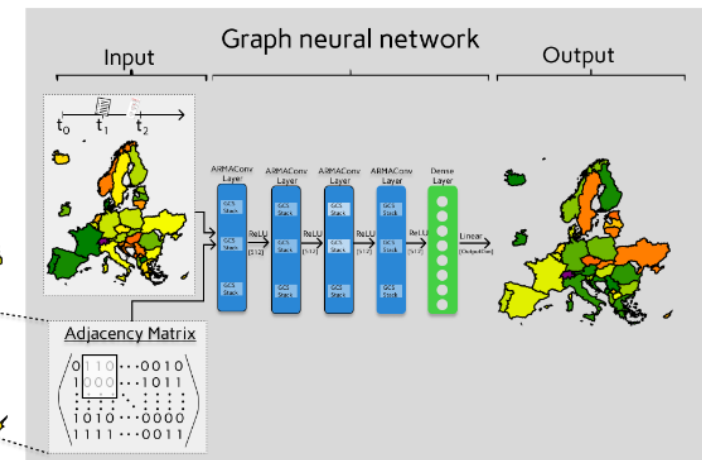
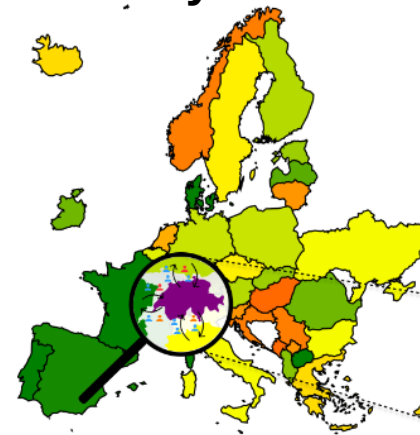
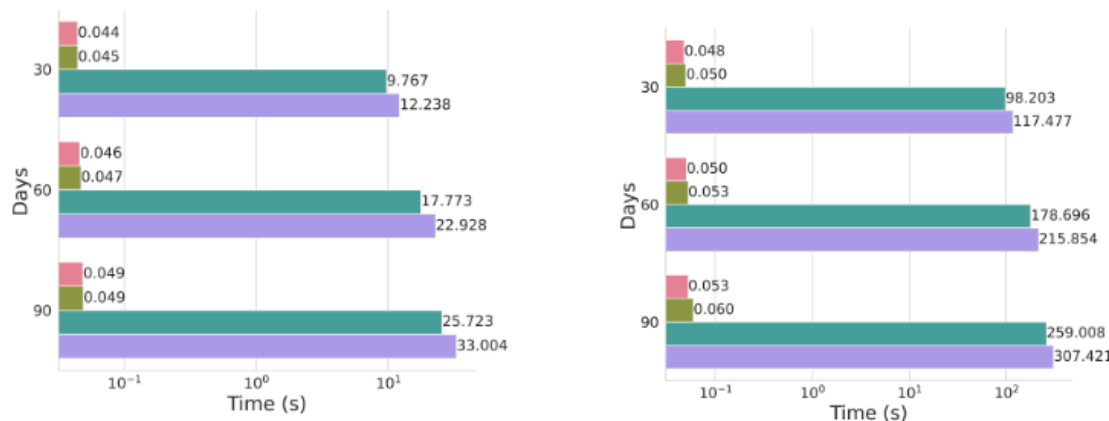


GNN (2026)

Schmidt et al., Scientific Reports (2026)

<https://www.nature.com/articles/s41598-026-39431-5>

→ Enables low-barrier access for decision makers and timely reaction



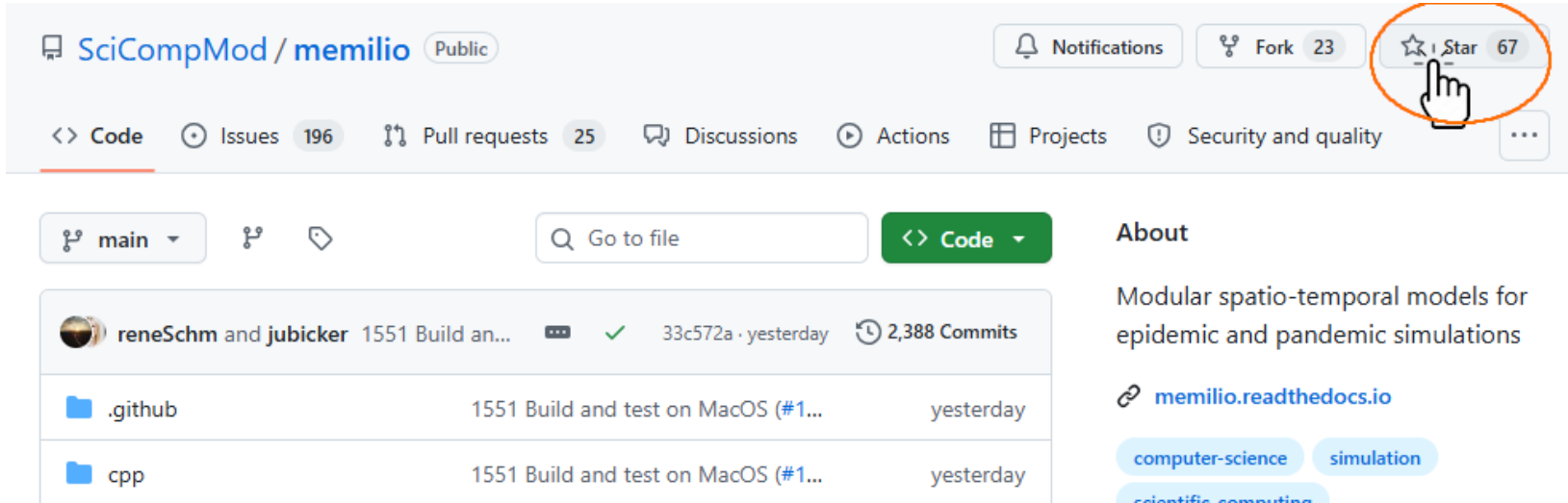
Closing

16:50 – 17:00

Summary



If you want to do us a favor, give us a „star“ on GitHub

A screenshot of the GitHub repository page for SciCompMod/memilio. The repository is public and has 67 stars, 23 forks, 196 issues, and 25 pull requests. A hand cursor is pointing at the star button, which is circled in orange. The repository description is "Modular spatio-temporal models for epidemic and pandemic simulations". The repository is maintained by reneSchm and jubicker. The repository has 2,388 commits. The repository is a C++ project (cpp) and has a .github directory. The repository is built and tested on MacOS (#1...) yesterday. The repository is also linked to the website memilio.readthedocs.io. The repository is tagged with the categories computer-science, simulation, and scientific-computing.

SciCompMod / **memilio** Public

Notifications Fork 23 Star 67

<> Code Issues 196 Pull requests 25 Discussions Actions Projects Security and quality

main Go to file Code

reneSchm and **jubicker** 1551 Build an... 33c572a · yesterday 2,388 Commits

.github	1551 Build and test on MacOS (#1...	yesterday
cpp	1551 Build and test on MacOS (#1...	yesterday

About

Modular spatio-temporal models for epidemic and pandemic simulations

memilio.readthedocs.io

computer-science simulation scientific-computing

Summary



What we provide (and hopefully have shown you today)

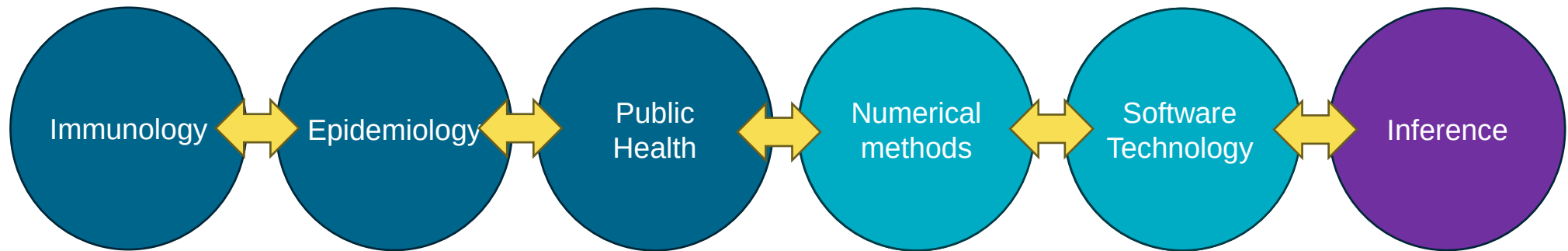
- a high-performance backend for infectious disease modeling
- a large set of model types which can be interchanged
- models that are easy to stratify, generalize, or extend
- a framework that is stable and extensively tested

What could not show in detail, do not yet provide, or are currently developing

- Agent-based models allow highly detailed simulations with testing, vaccinations etc.
- Hybrid models are promising approaches to optimize the tradeoff between details and speed
- We are currently extending MEmilio for post-acute infections (PAIS), transmission chains in hospitals and care home settings, veterinary diseases
- We are actively pushing the computational performance of MEmilio including GPU computing
- For most diseases, new courses of disease might be needed
 - partially, this can be done automatically with our model generator
- Not all modeling types are equally advanced and some functionality might be missing
 - **we are happy to help out**

Summary

- Let's create interdisciplinary synergies (with MEmilio)
- We are always happy about requests for discussion or cooperation
 - on small research projects
 - on joint grant proposals
 - ...



Contact: memilio@DLR.de (if you do not get a reply, try: Martin.Kuehn@DLR.de)

Thank you for your attention!

Feedback?

<https://de.surveymonkey.com/r/F9BJZJR>