Multiscale Modeling and Simulation

Alfons Hoekstra
University of Amsterdam

MAPPER Summerschool 2013

The Mapper project receives funding from the EC's Seventh Framework Programme (FP7/2007-2013) under grant agreement n° RI-261507.

Main contributors

- Bastien Chopard – University of Geneva
- Jean-Luc Falcone – University of Geneva
- Eric Lorenz – University of Amsterdam
- Joris Borgdorf – University of Amsterdam

Agenda

1. Scales
2. Multiscale Modeling
3. Multiscale Computing

Acknowledgements

www.complex-automata.org

www.meddica.eu

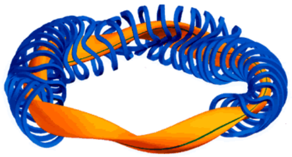
www.mapper-project.eu

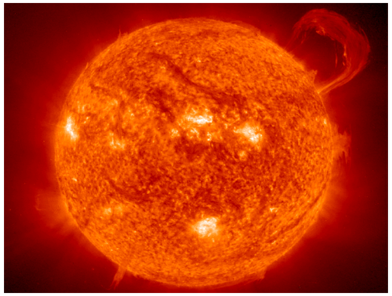
Special thanks to Pat Lawford, Rod Hose and Julian Gunn from Sheffield, UK

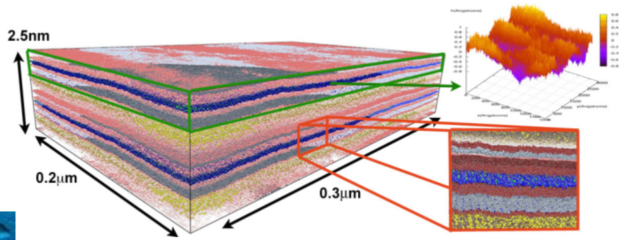
What do we mean with ‘scale’?

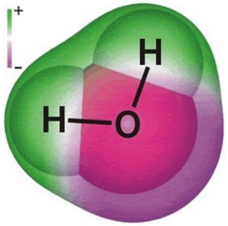
Nature is Multiscale

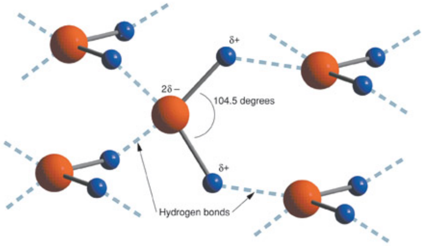
- Natural processes are multiscale
  - 1 H_2O molecule
  - A large collection of H_2O molecules, forming H-bonds
  - A fluid called water, and, in solid form, ice.

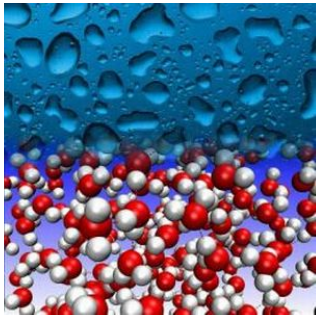




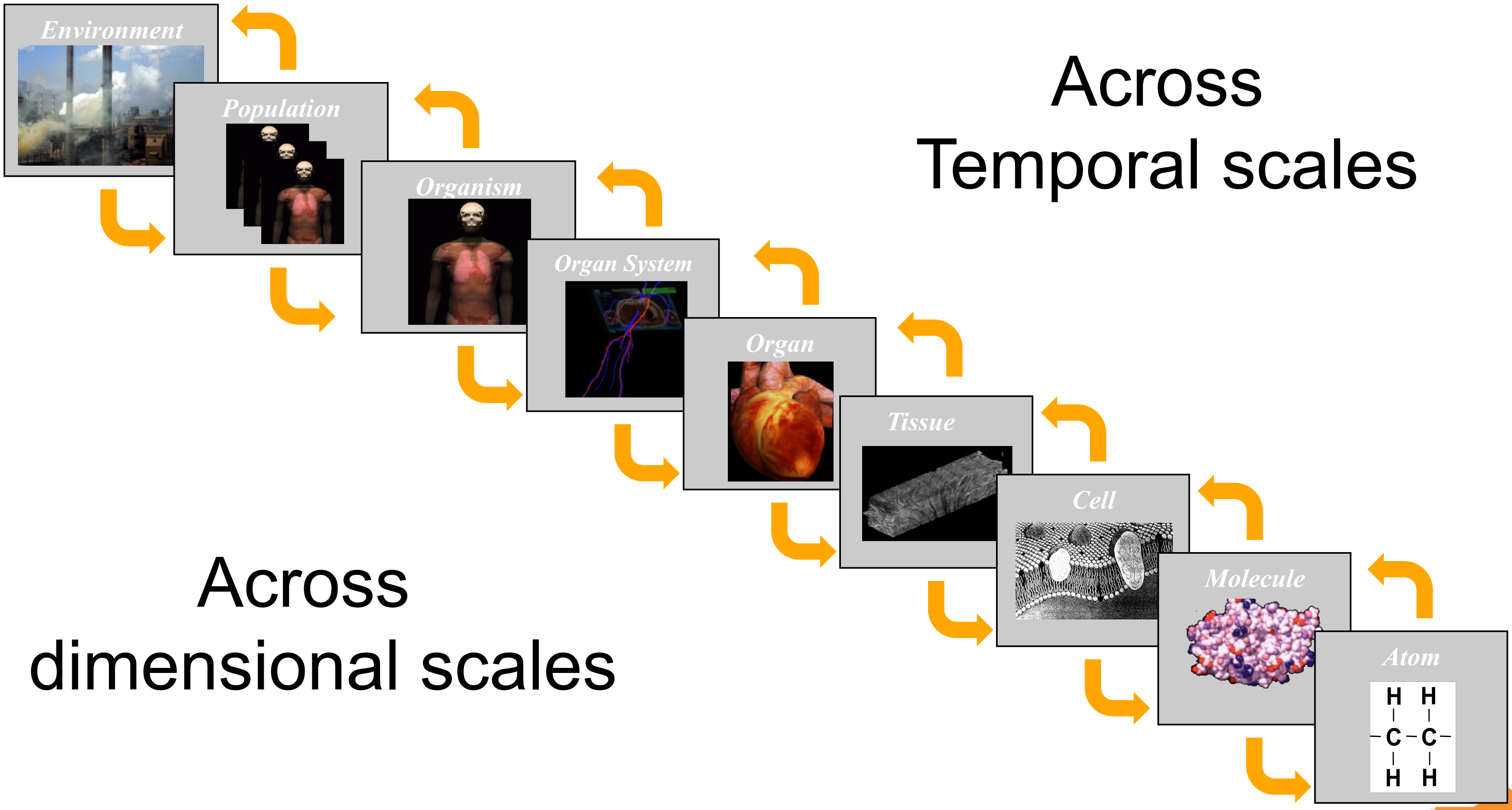








Multiscale models in Biomedicine



Across
Temporal scales

Across
dimensional scales

What do we mean with ‘scale’?

‘typical’ length scale

(level of organisation)

- quantum
- molecular
- macro-molecular
- sub-cellular
- cellular
- tissue
- organ
- organ system
- organism
- environment

‘typical’ time scale

- ‘fast’
- ‘intermediate’
- ‘slow’

In most multiscale studies we couple something ‘small and fast’ to something ‘large and slow’

What is the size of the scales?

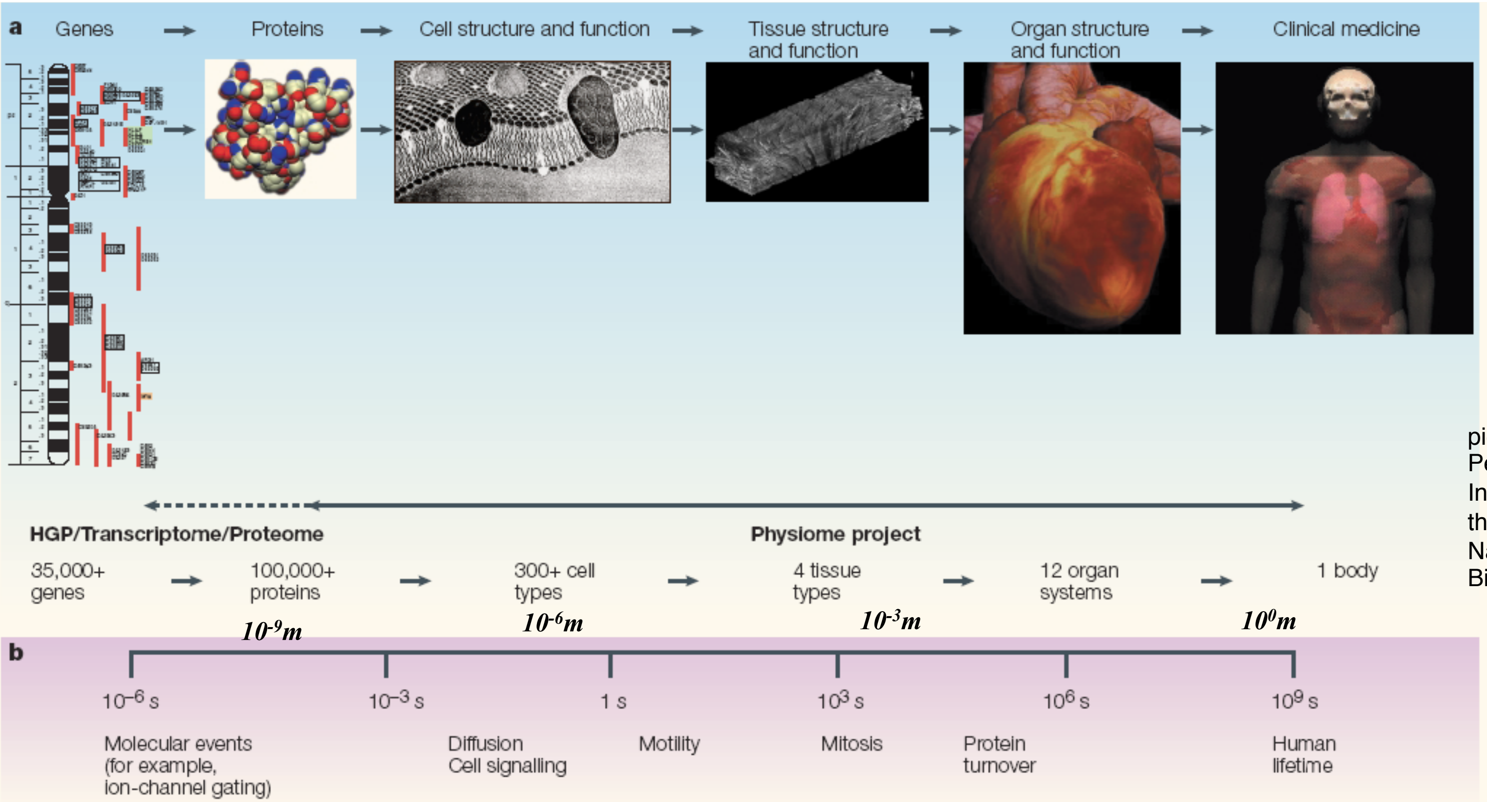


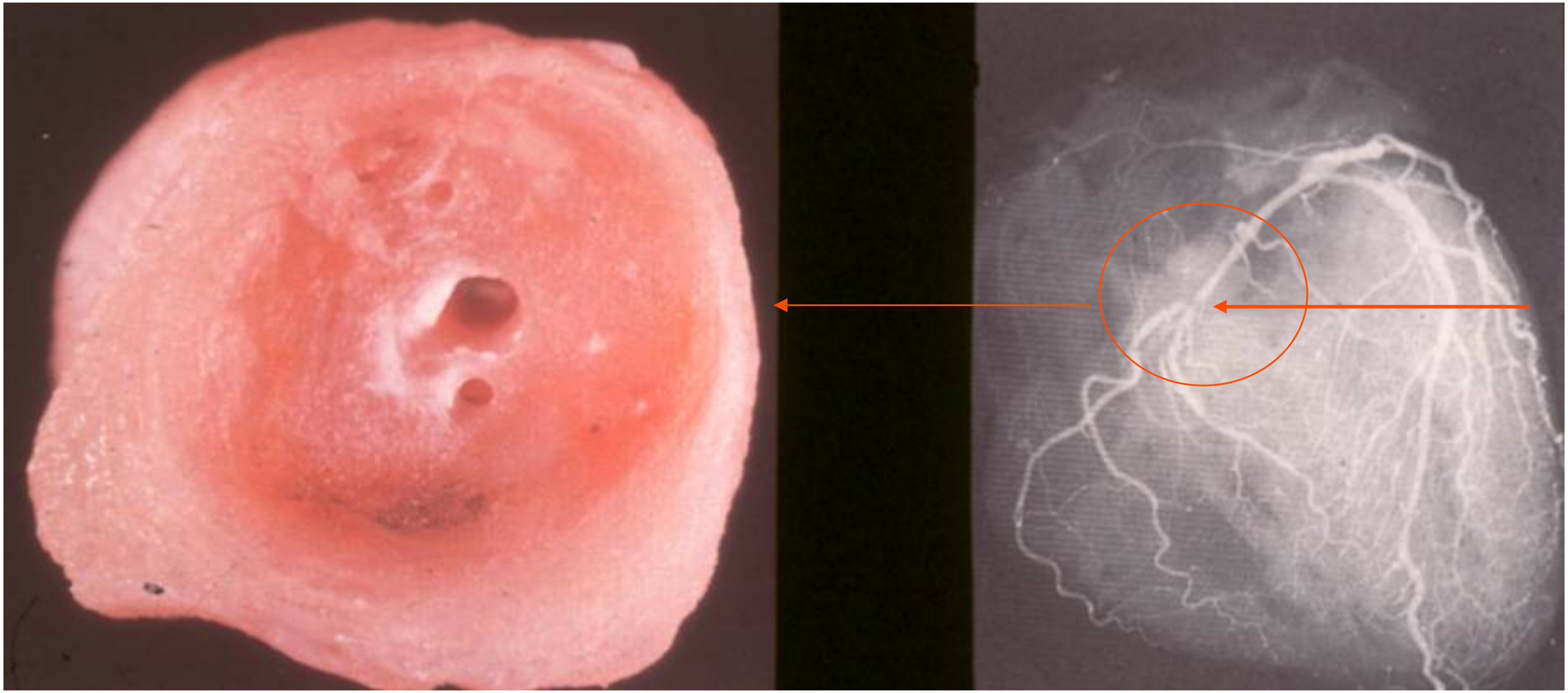
Figure 2 | **Linking molecular and cellular events with physiological function must deal with wide ranges of length scales and timescales. a** | Levels of biological organization from genes to proteins, cells, tissues, organs and finally the whole organism. The range of spatial scales — from ~1 nm for proteins to ~1 m for the whole body — requires a hierarchy of models. Different types of model are appropriate to each level, and relationships must be established between models at one level and the more detailed, but spatially or temporally limited, models at the level below. The organ-level and whole-body-level models shown are the Auckland heart and torso models, respectively[32,38]. The tissue figure is a reconstructed three-dimensional confocal image of a transmural section of rat myocardium, which is also from the Auckland Bioengineering Institute, New Zealand[28]. **b** | The range of temporal scales as shown here is even more daunting and again calls for a hierarchy of models. HGP, human genome project. Modified with permission from REF. 39 © Springer-Verlag (2002).

picture taken from:
Peter J. Hunter and Thomas K. Borg,
Integration from Proteins to Organs, the Physiome Project,
Nature Reviews Molecular Cell Biology, **4**, 237-243, 2003

Example: drug eluting stent in a coronary artery

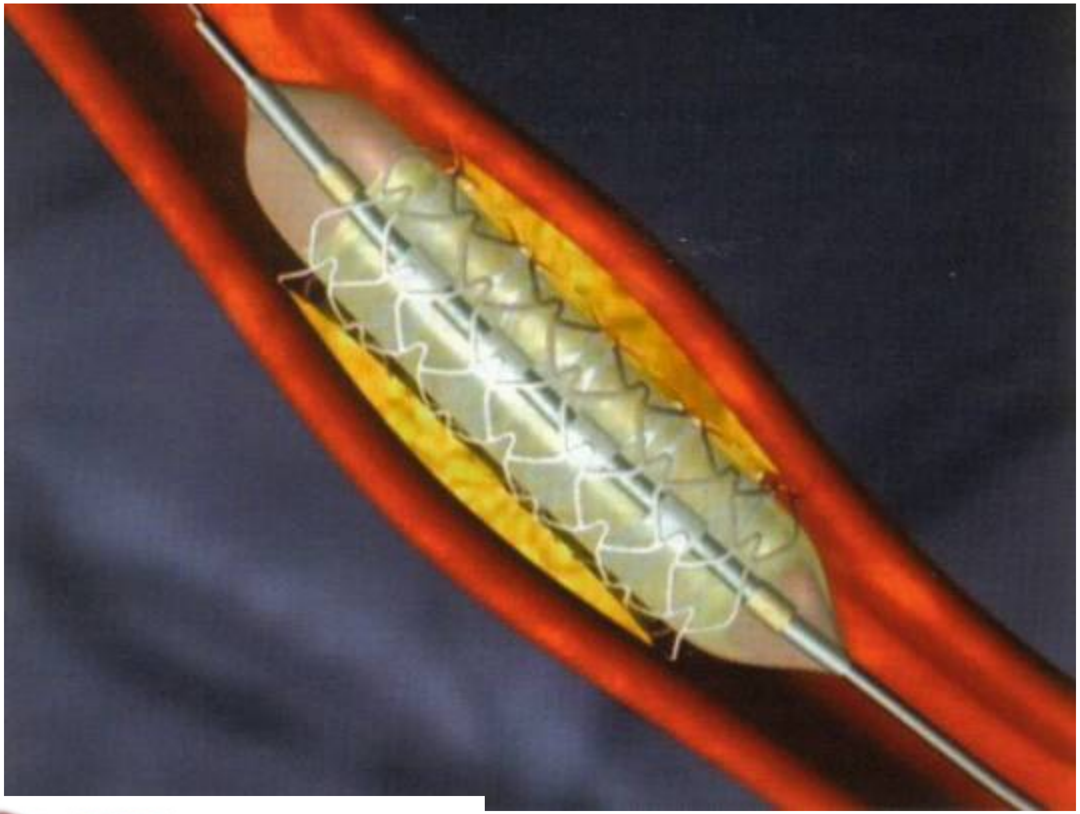
- So, how do you figure out the scales?
- First a few pictures to get going

Coronary artery disease



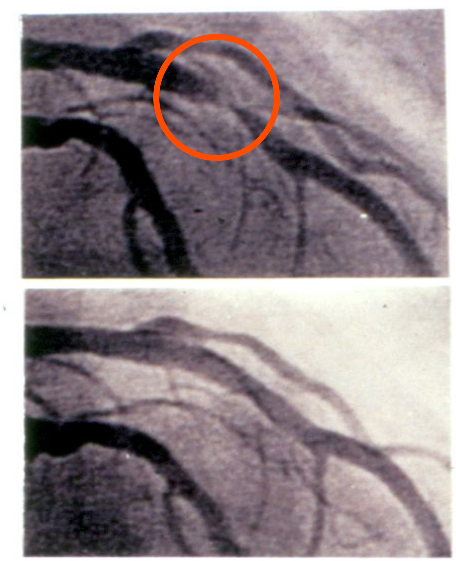
Gross appearance *Angiogram*

Balloon Angioplasty and Stent implantation

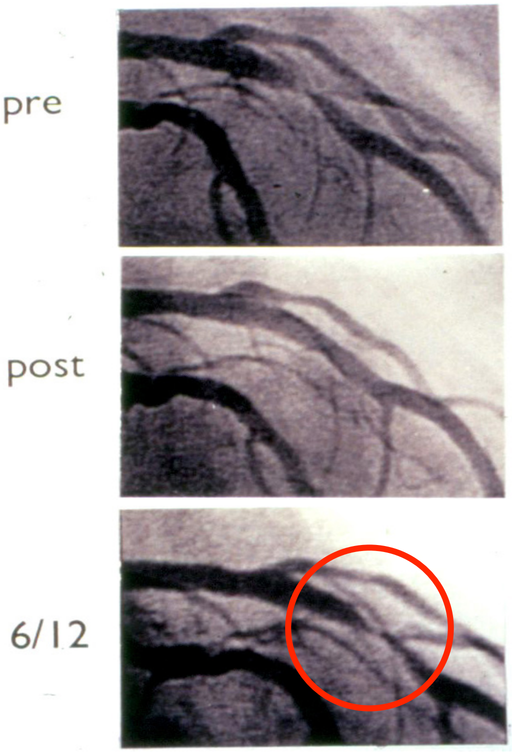


pre

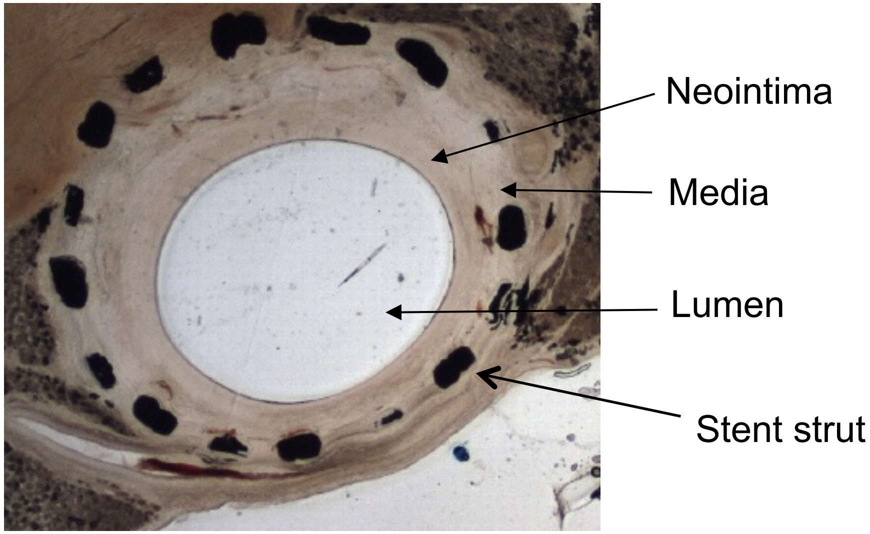
post



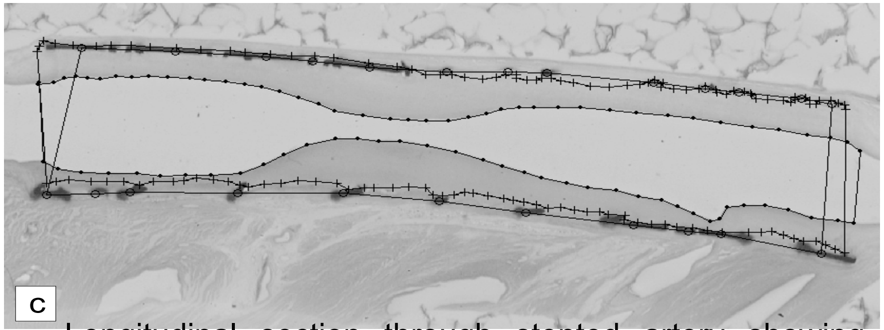
What is Restenosis?



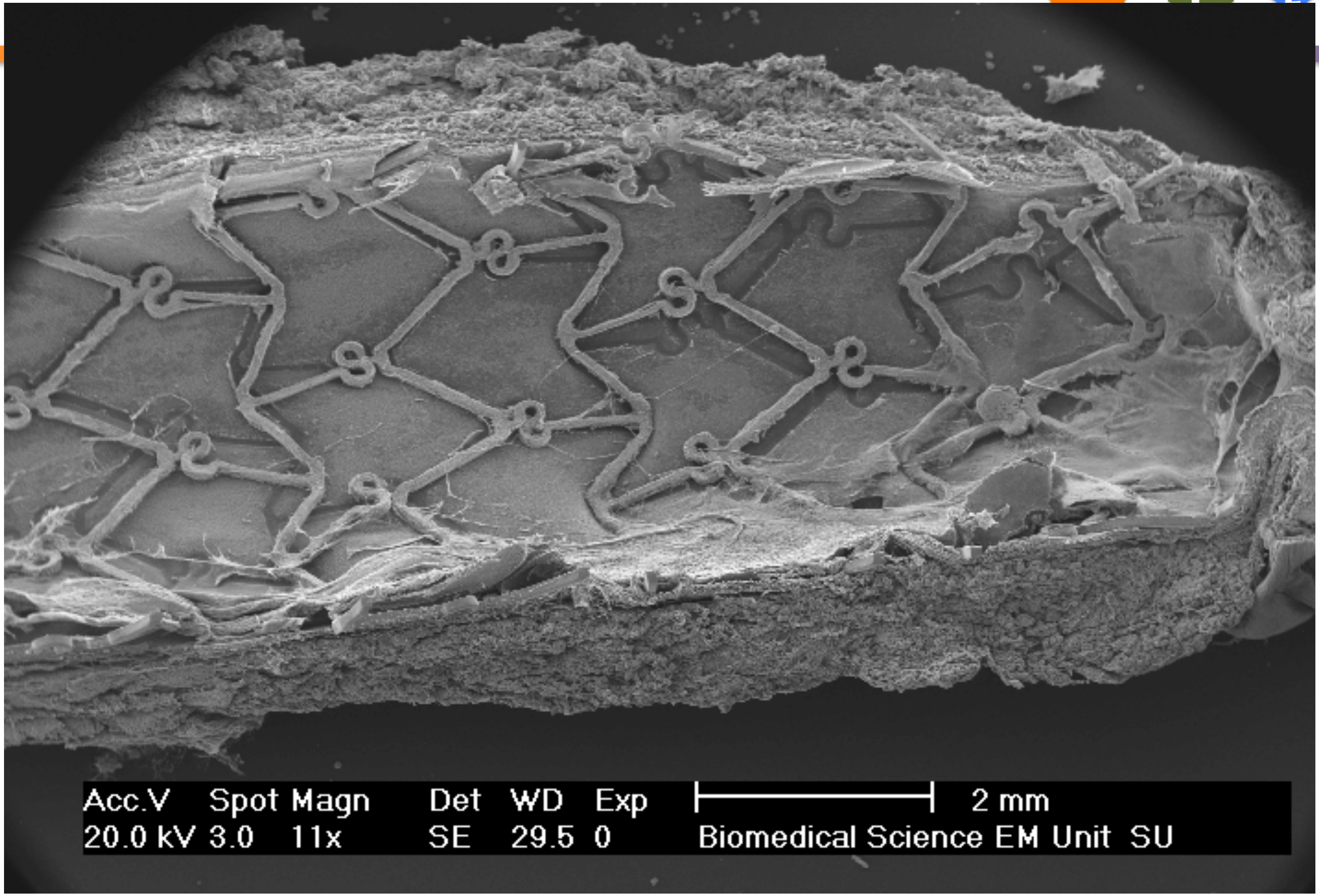
Human angiogram depicting restenosis six months post-PCI.



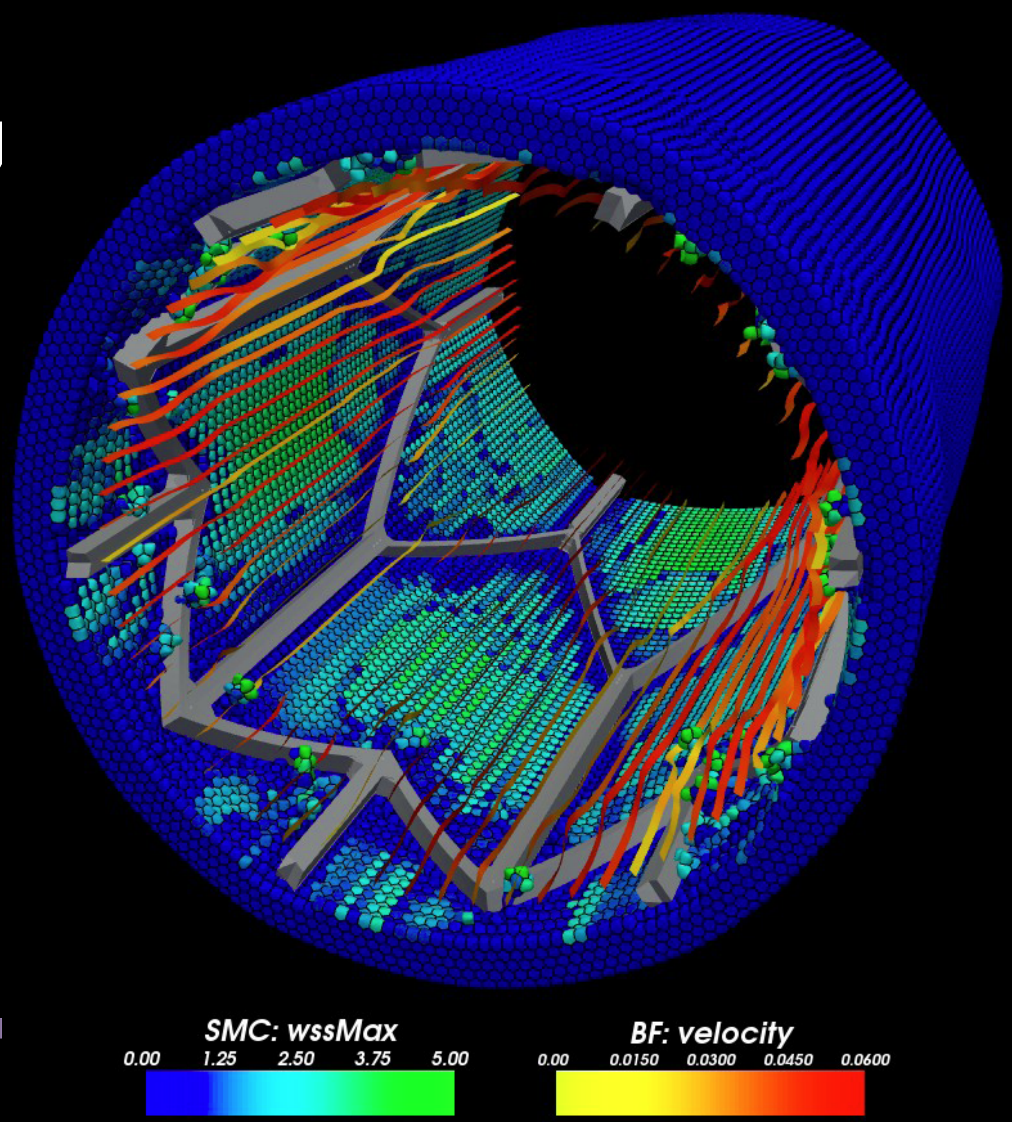
Porcine coronary artery section 28 days post stenting displaying substantial neointima.



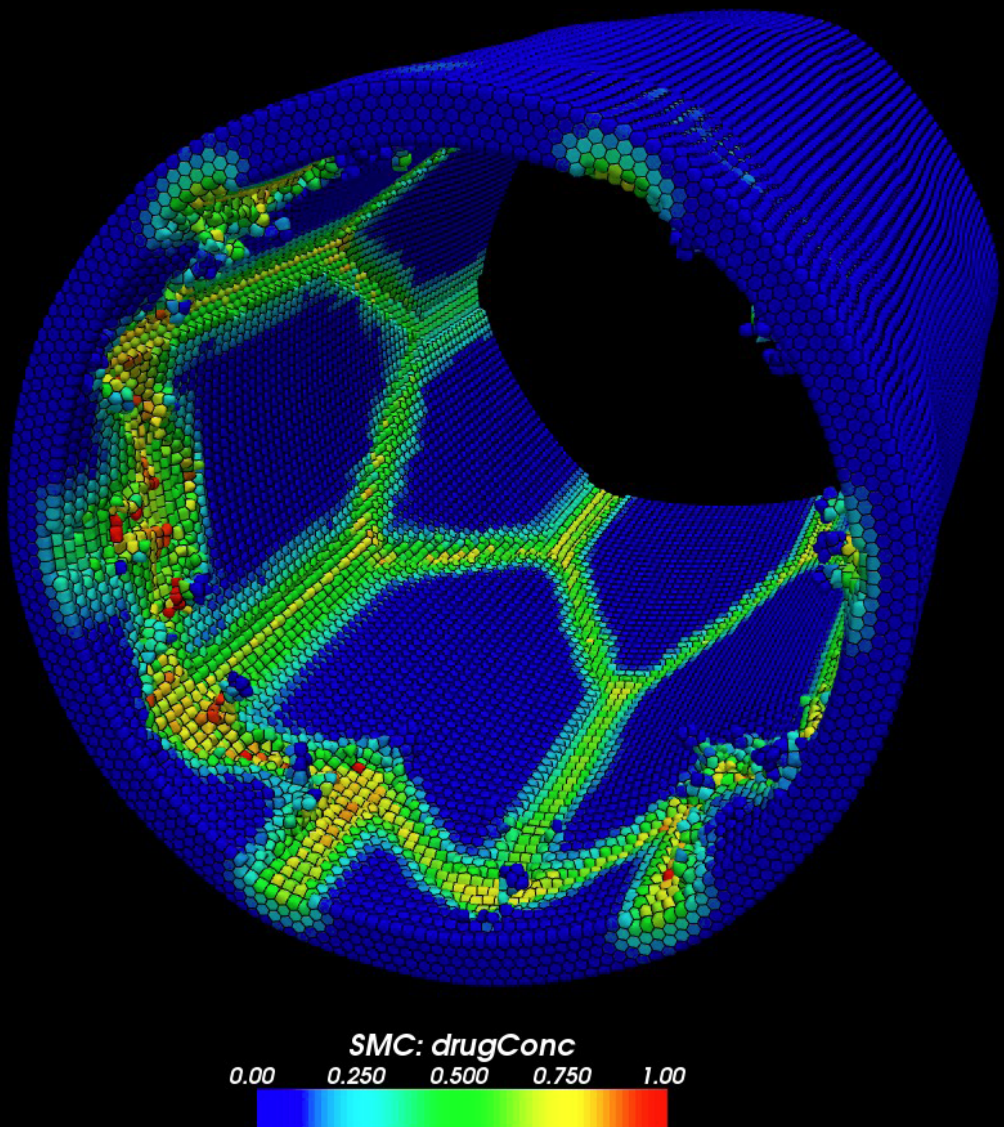
Longitudinal section through stented artery showing variation in reaction along vessel length


Acc.V Spot Magn Det WD Exp
20.0 kV 3.0 11x SE 29.5 0
2 mm
Biomedical Science EM Unit SU

SMCs (WSS color scale)
Stent
Flow (Ribbons, color scale)



Drug concentration coloring



Example: drug eluting stent in a coronary artery

- So, how do you figure out the scales?

- Your turn

Length scales

diameter of artery	→	1,5 mm
diameter of stent strut	→	100 μm
diameter of smooth muscle cells	→	20 μm
size of red blood cells	→	10 μm

subcellular ?

length of artery ?

curvature of artery ?

thickness of artery wall ?

others?

Time scales

pulsatile flow → O(1 s)

diffusion of drugs in artery wall → O(10 h)

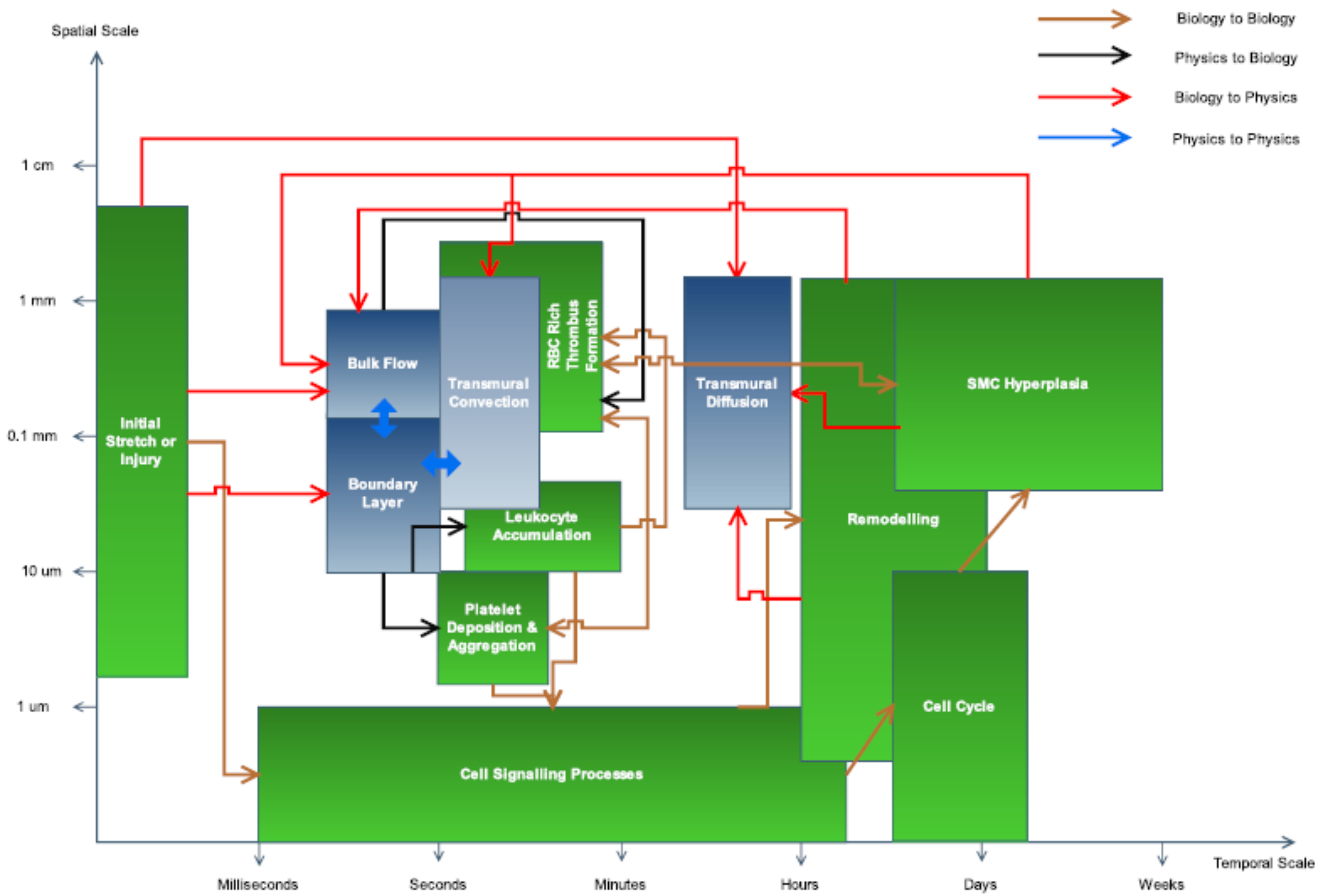
cell cycle of smooth muscle cells → O(30 h)

transport time of cells in boundary layer ?

cell signaling ?

others ?

Scale Separation Map



Multiscale Modelling

Why multiscale models?

- There is simply no hope to computationally track complex natural processes at their finest spatio-temporal scales.
  - Even with the ongoing (exponential) growth in computational power

Minimal demand for multiscale methods

$$\frac{\text{cost of multiscale solver}}{\text{cost of finescale solver}} << 1$$

$$\text{errors in quantities of interest} < tol$$

Multiscale modelling

"Multiscale Mathematics, Modelling and Simulation should address:

- What is the information that needs to be transferred from one model or scale to another?
- What (physical) principles must be satisfied during the transfer of information?
- What is the optimal way to achieve such transfer of information?
- How to quantify variability of physical parameters and how to account for it to ensure design robustness?"

Jacob Fish in the foreword to J. Fish (Ed.) *Multiscale Methods, bridging the scales in science and engineering*, Oxford University Press, 2010.

Example, coupling bloodflow to SMC proliferation

SMC proliferation can be regulated by (oscilatory) wall shear stress and cyclic stretch.

- what is the information that needs to be transferred from one model or scale to another;
  - Blood flow solver → SMC solver : (oscillatory) shear stress
  - SMC solver → blood flow solver : lumen geometry
- what (physical) principles must be satisfied during the transfer of information?
  - Correct units
  - Conservation of momentum and energy (in case of FSI)
  - Integrity of flow domain (no holes, etc)
- what is the optimal way to achieve such transfer of information?
  - Mapping lattice based information to agent based information
  - Correct interpolation and integration of stress values
  - Mapping agent-space back to a voxel space

Multiscale Multiscience Modelling framework

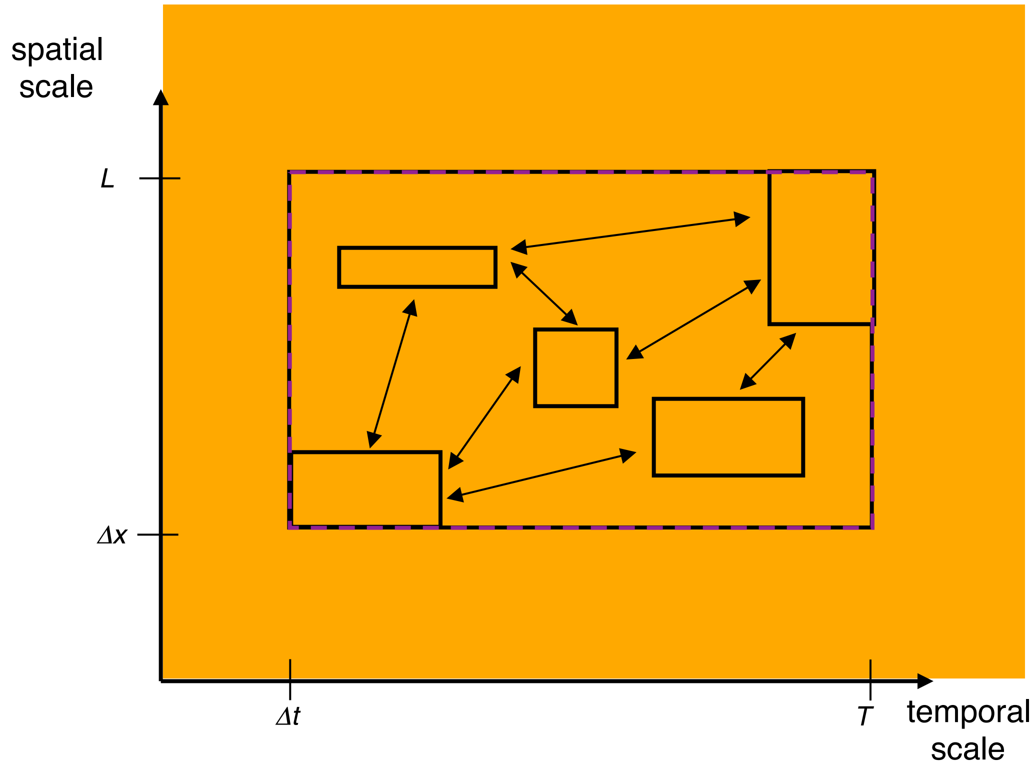
- Propose a modeling and simulation framework for multiscale, multiscience complex systems
- Theorectical concepts
  - Based on our earlier work on Complex Automata (CxA)
- A Multiscale Modeling Language : MML
- Software environment : The MUSCLE coupling library

Motivations

- Very few methodological papers in the literature.
  - but increasing …
- Multiscale strategies are usually entangled with applications.
- Can we develop a framework that help the design and deployment of complex multiscale-multiscience applications ?

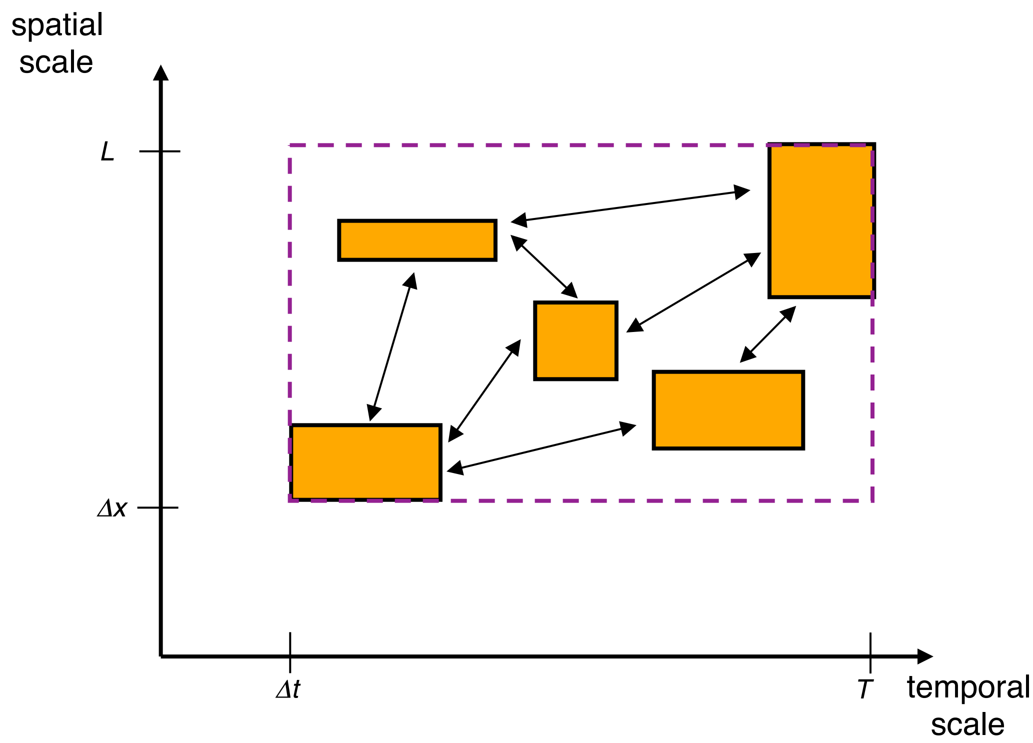
Multi-Scale modeling

- Scale Separation Map
- Nature acts on all the scales
- We set the scales
- And then decompose the multiscale system in single scale sub-systems
- And their mutual coupling



From a Multi-Scale System to many Single-Scale Systems

- Identify the relevant scales
- Design specific models which solve each scale
- Couple the subsystems using a coupling method

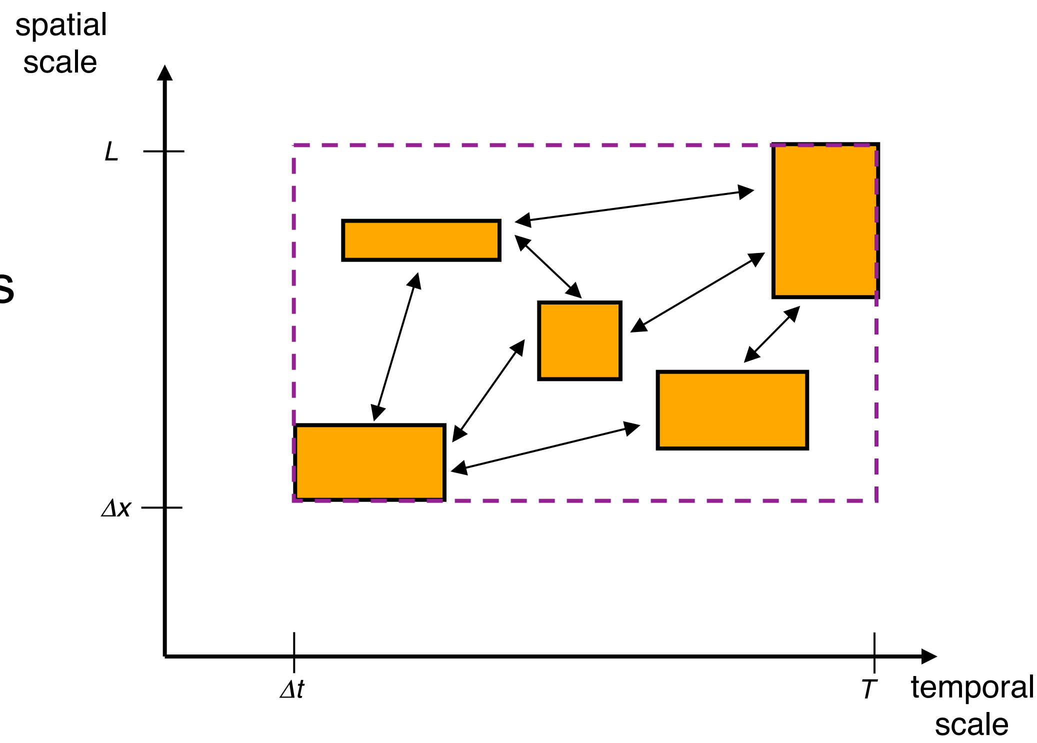


From a multiscale system to many single scale systems

- Model a multiscale system as a collection of coupled single scale models.
  - **Scale Separation Map**
- Single scale models may implement many different numerical methods.
  - But they are all described with the same **submodel execution loop**.
- Submodels should not know about the rest of the system : they are autonomous components
- Only the **smart conduits** know about the properties of the submodels they connect.

Single-Scale Models

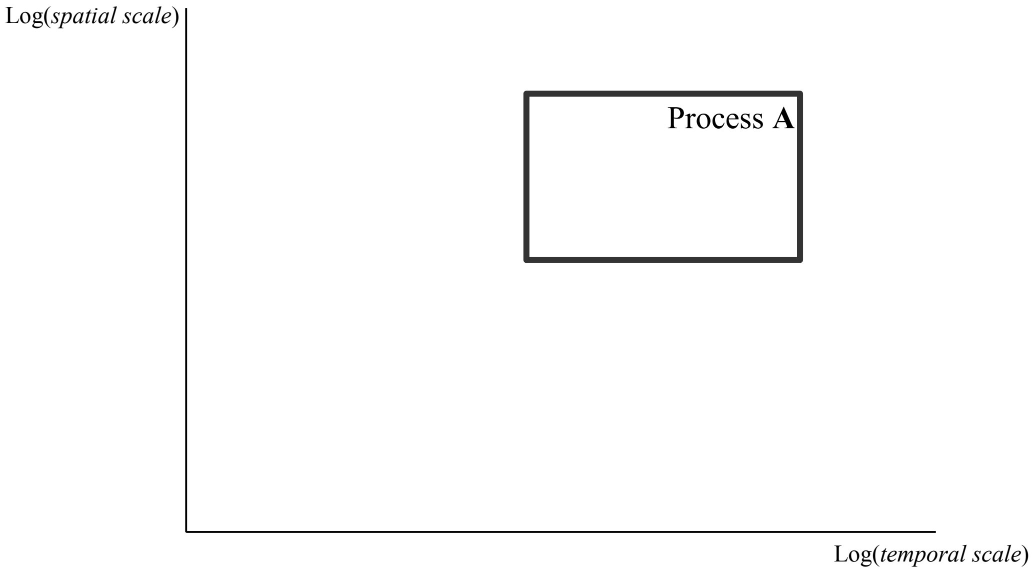
- Cellular Automata (CA) or Agent Based Models (ABM)
  - powerful approaches to describe complex systems
  - CA and ABM can be decribed in a generic way (see below)
  - With less, do more : Complex Automata (CxA)



The Scale Separation Map

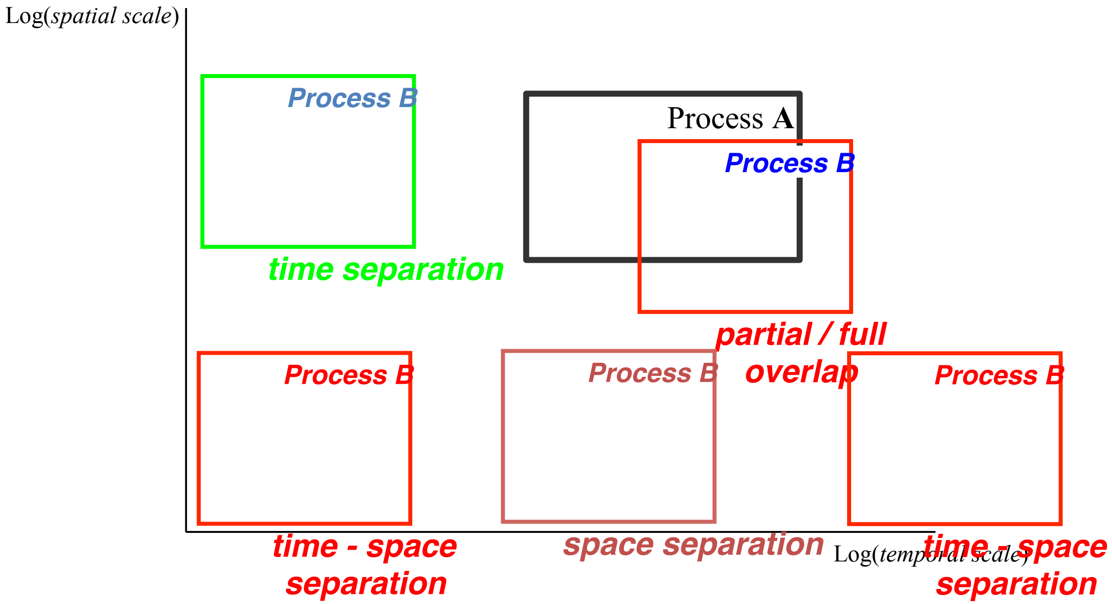
- A powerful methodological way to identify sub-models

- Classify the sub-model interactions as full or partial overlap of scales.



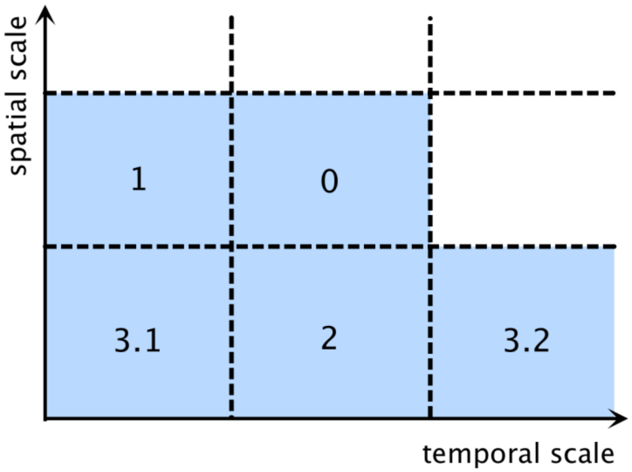
Relation between the scales

- The Scale Separation Map specifies the relation between the sub-models in **five regions**.
- There is more than the standard micro-macro relation and more than than the “bi-scale" modeling.

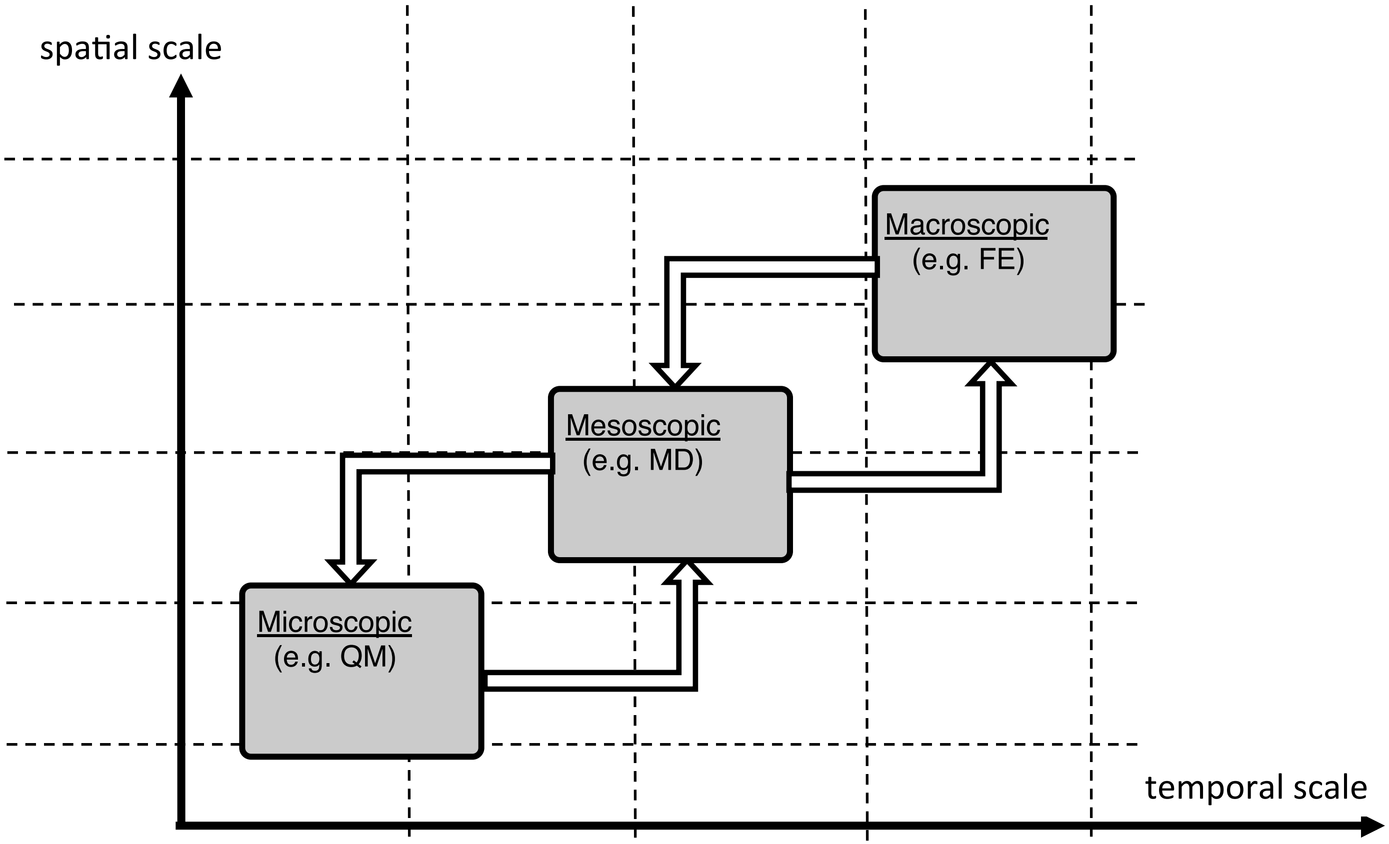


The Scale Separation Map

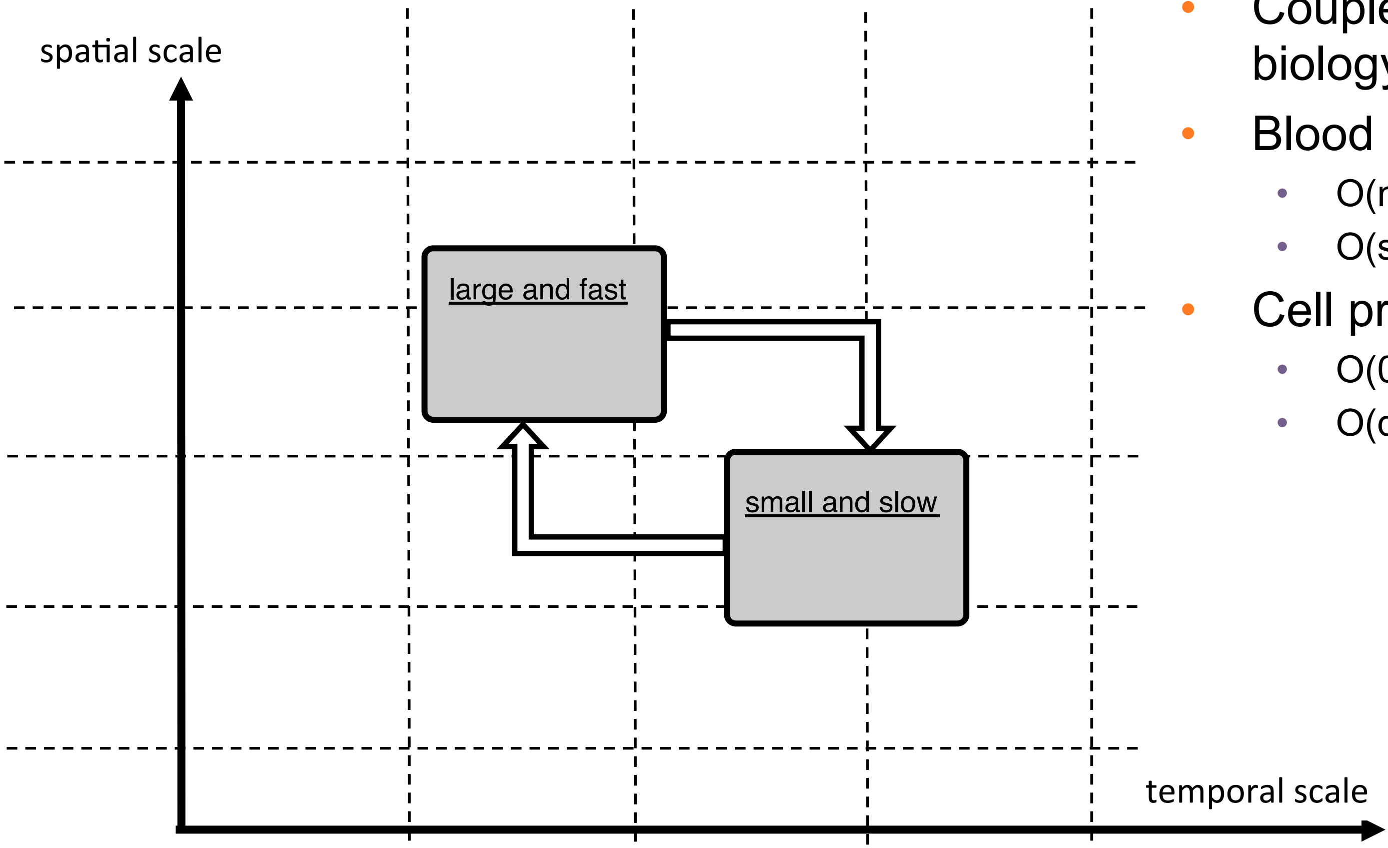
- A powerful methodological way to identify sub-models
- Classify the sub-model interactions as full or partial overlap of scales.
- Specify the relation between the sub models in five interaction regions.



Region 3.1 Micro → Macro

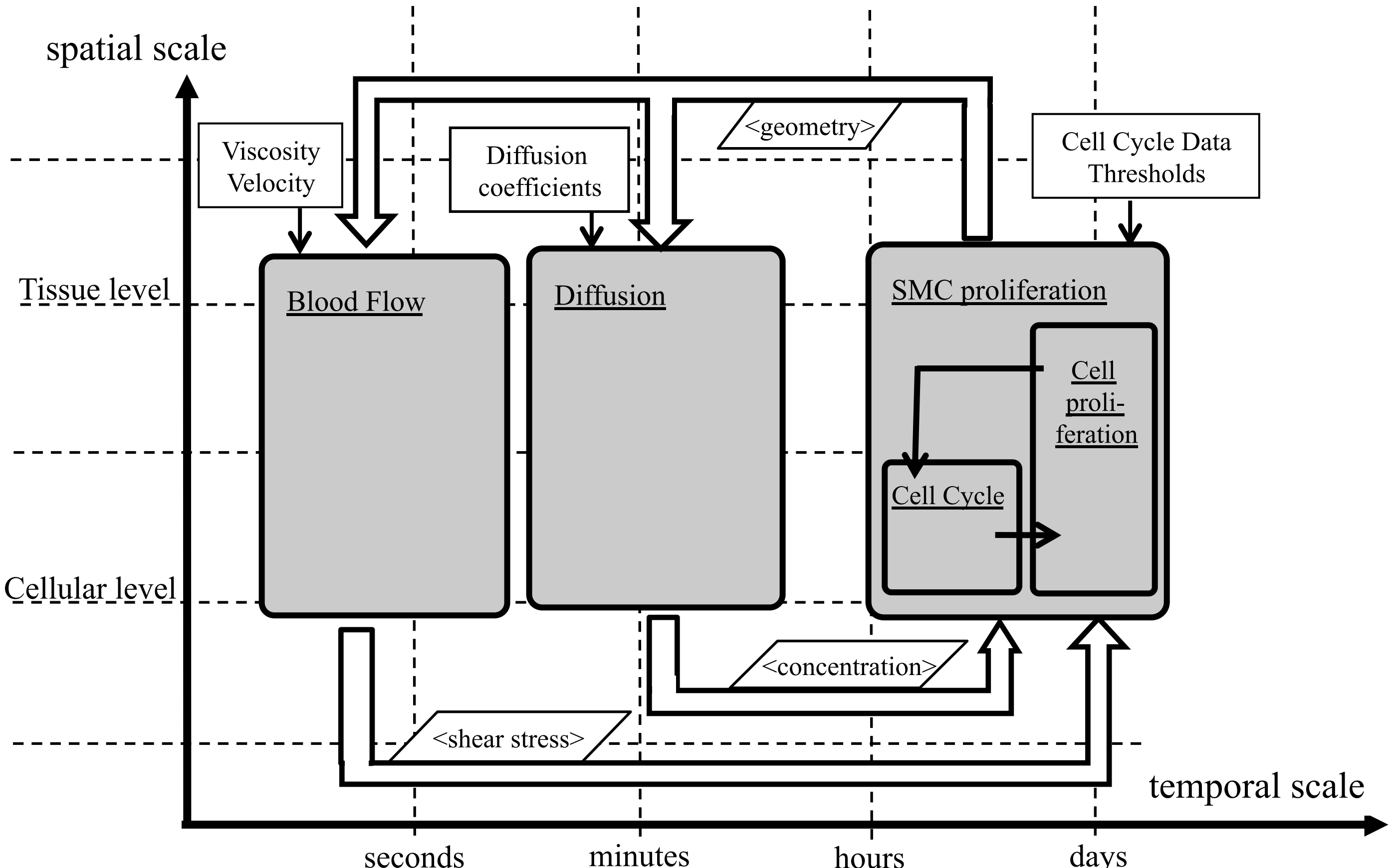


But what about region 3.2?



- Couple physics with biology
- Blood flow in artery
  - O(mm) length scale
  - O(second) time scale
- Cell proliferation
  - O(0.01 mm) length scale
  - O(day) time scale

Simplified Scale Separation Map for ISR



Legend:

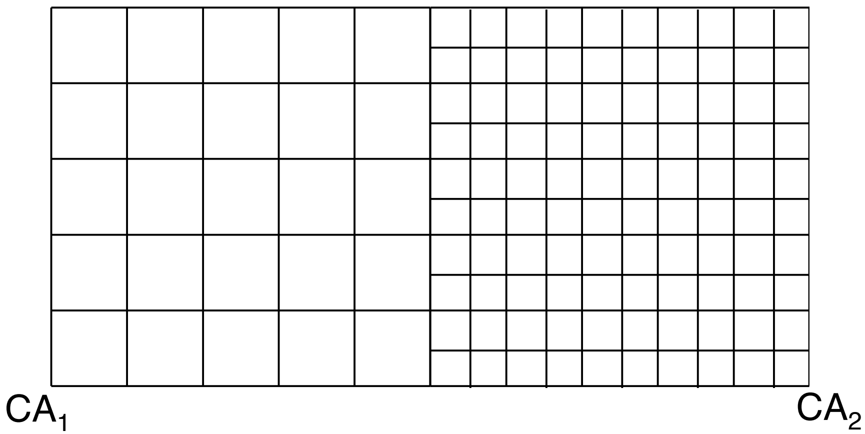
Inputs/outputs to single-scale models

Coupling between different-scale models

< … > Data items passed in coupling templates

Relation between computational domains

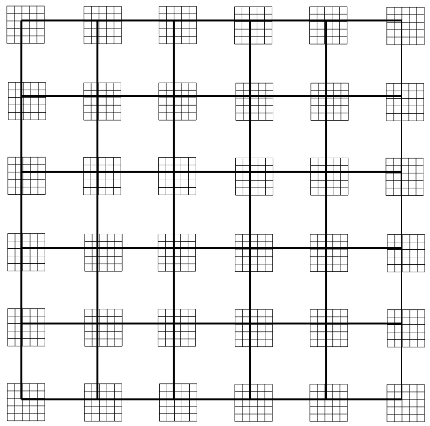
Multi Domain



(scale overlap)

The computational domain is split into
a coarse and a fine sub domain

Single Domain



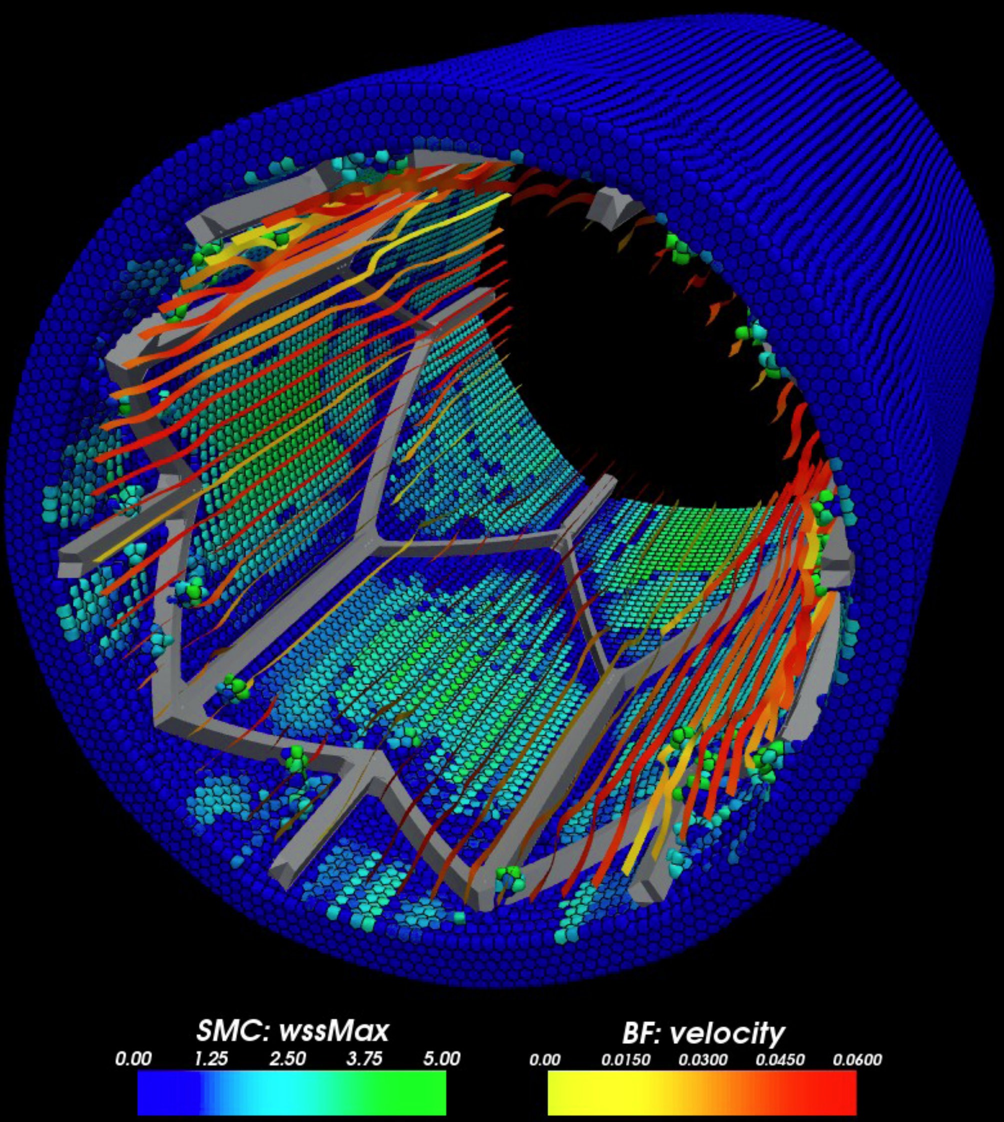
(micro-macro separation)

Example multidomain

Bloodflow in the lumen

coupled to

SMC growth in the wall

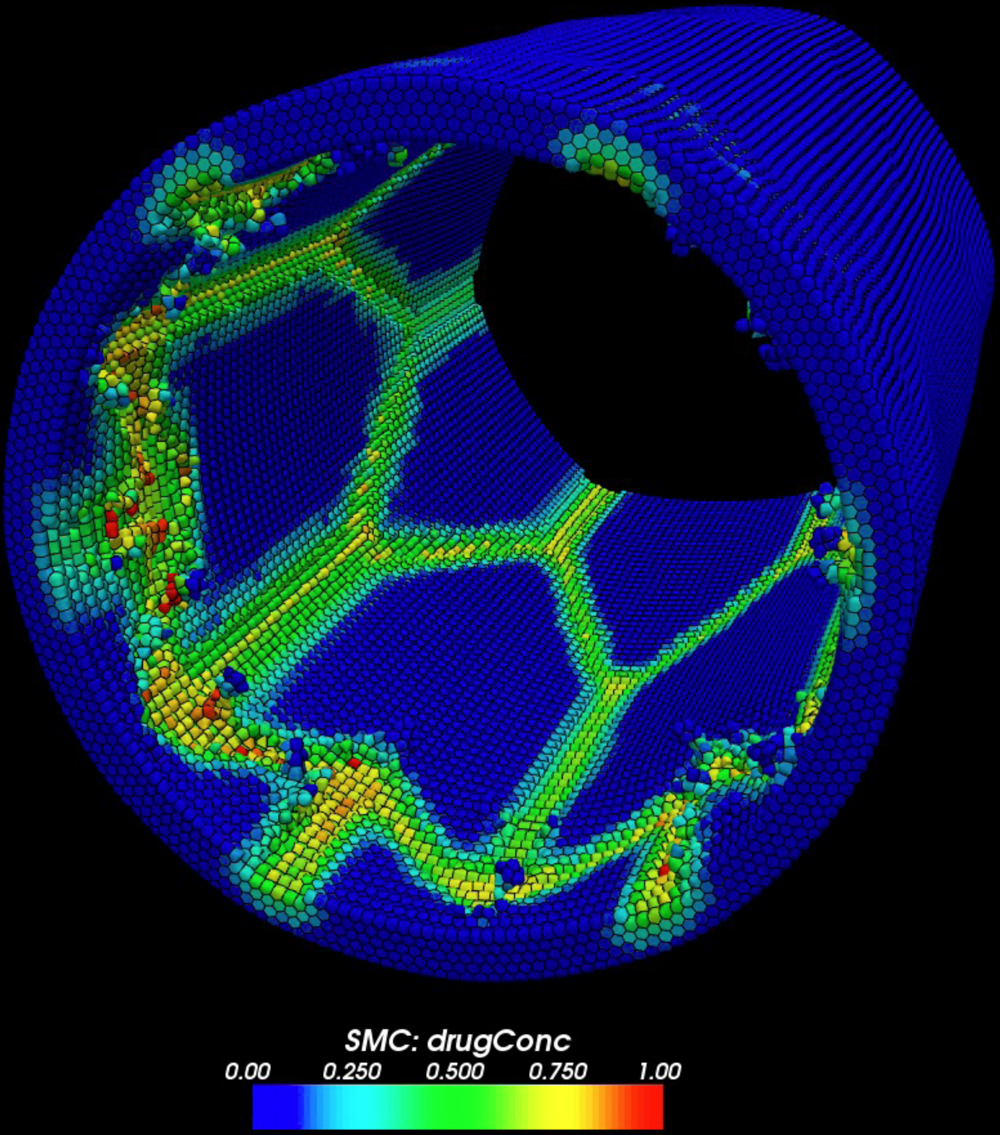


example Single Domain

Diffusion of drug in the wall

coupled to

SMC growth in the wall

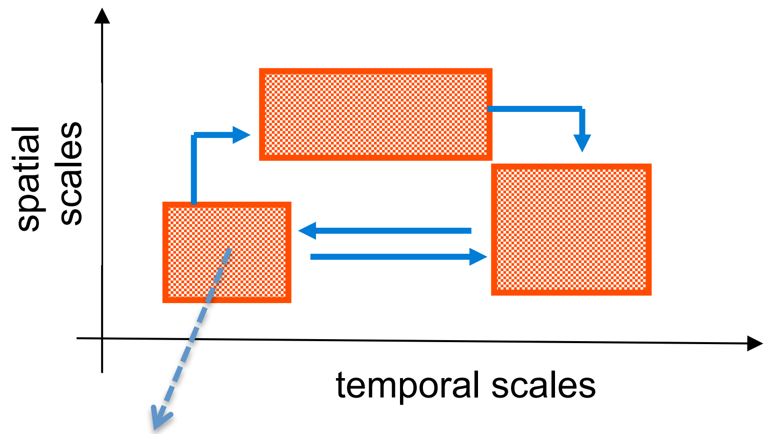


More formalism

- Submodel execution loop and coupling templates
- Taxonomy of multiscale models
- Formalism

Generic Submodel Execution Loop

- $\mathbf{f}_{init}$ is for initialization
- **S** is for one iteration of the Solver
- **B** is to specify the boundaries
- $\mathbf{O}_i$ is for intermediate observation
- $\mathbf{O}_f$ is for final observation

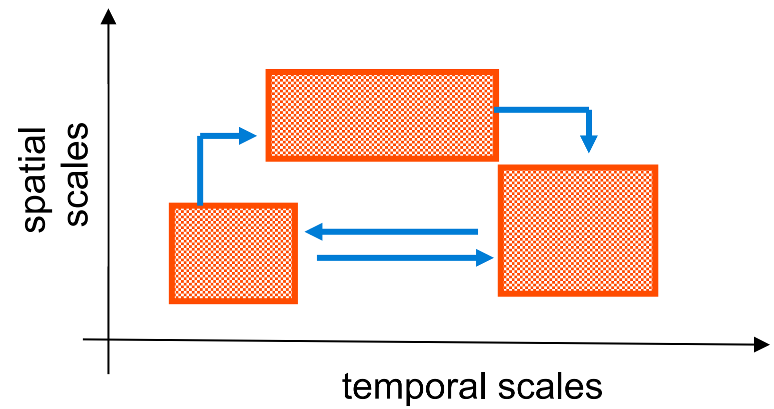


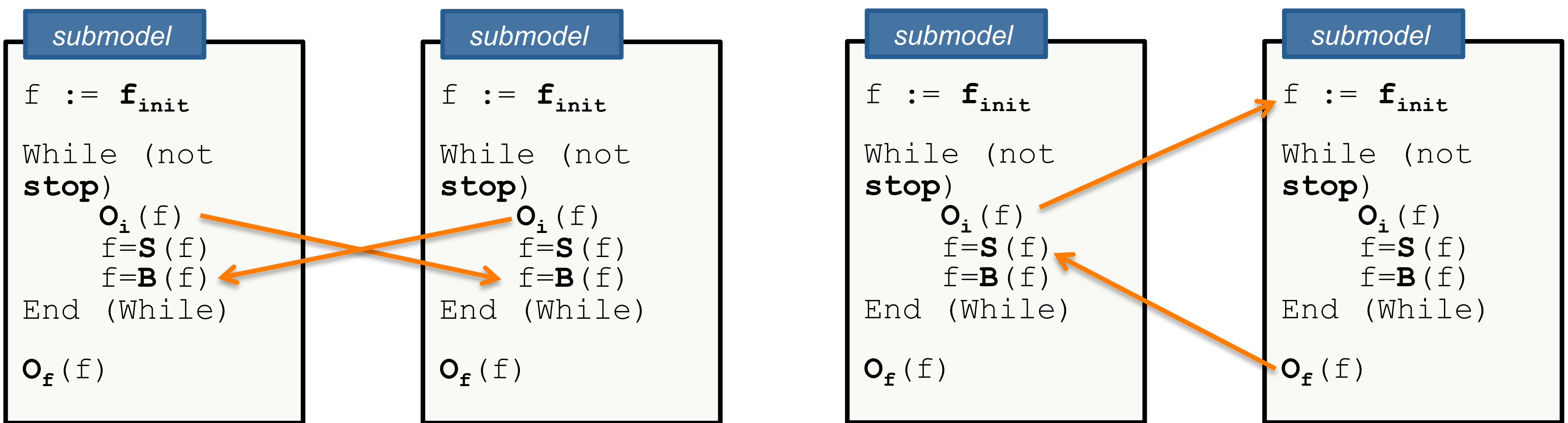
submodel

```
f := finit

While (not
stop)
     Oi(f)
     f=S(f)
     f=B(f)
End (While)

Of(f)
```

Coupling templates

- Coupling operators of the submodel execution loop of different submodels together
- **O**i , **O**f as origin
- finit , **B** and **S** as possible destinations

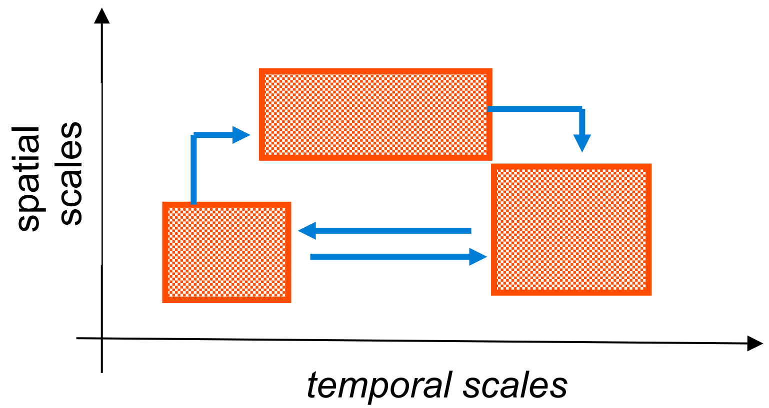


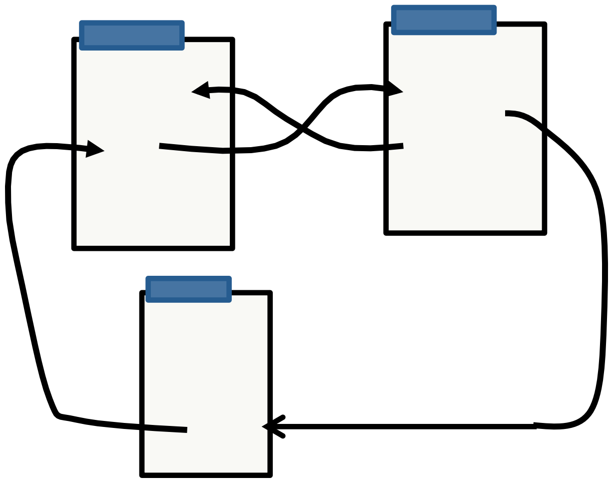


Connection scheme

- Scale separation map
- Submodel Execution Loop
- Coupling templates

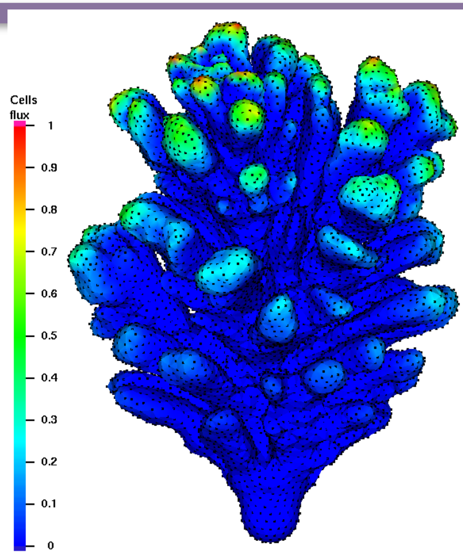
- Putting it all together results in a connection scheme (graph)

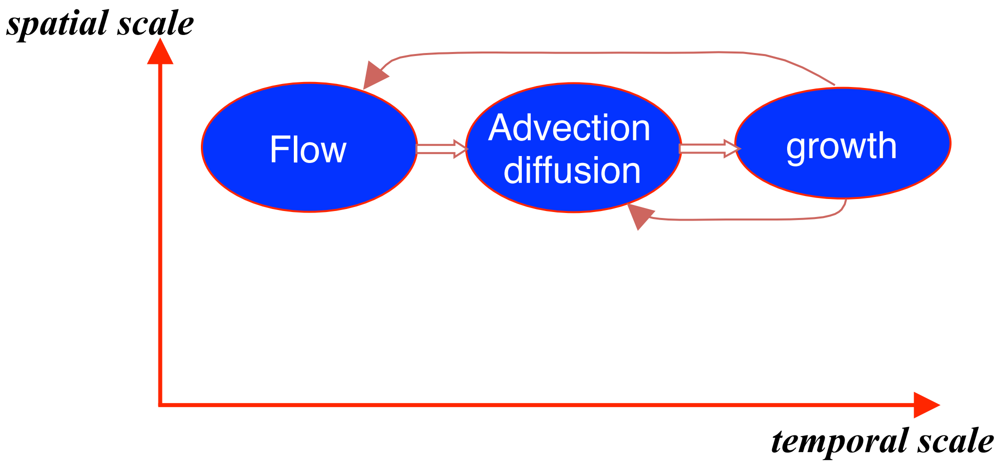




Example, Coral Growth

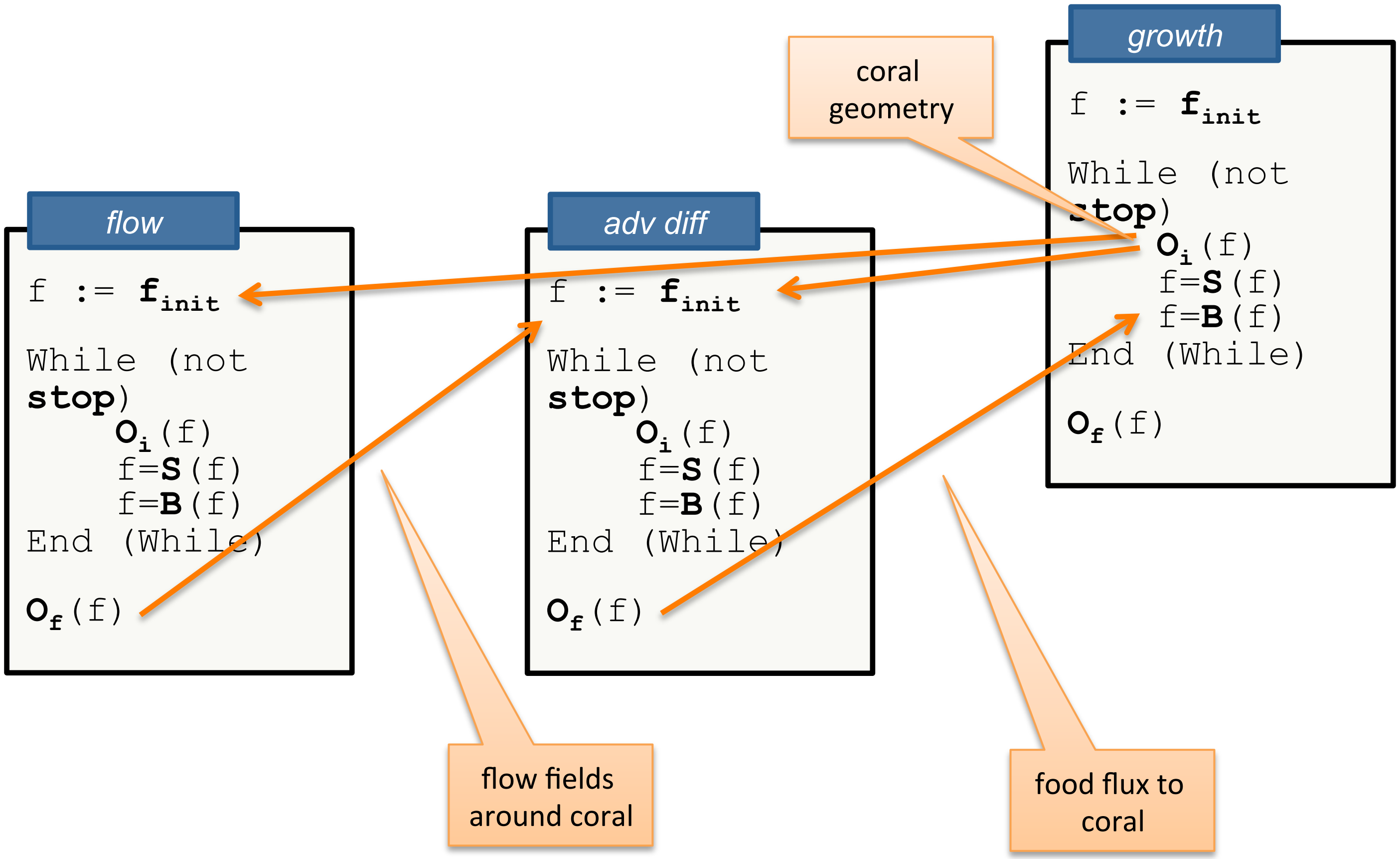
- Simulated Coral Growth
  - Growth dictated by influx of nutrients at coral surface
  - Nutrients are transported by water flow around the coral
- Three subsystems
  - All operating on the same length scale of the coral
  - Flow around the coral
    - Fast process, O(s)
  - Advection-diffusion of nutrients
    - Diffusion time scale O(10 s) to O(1 min)
  - Growth of coral
    - Accretive growth or aggregation
    - Slow process, O(year)

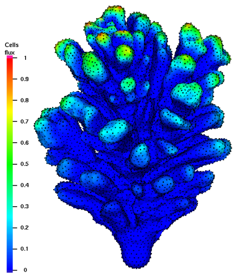




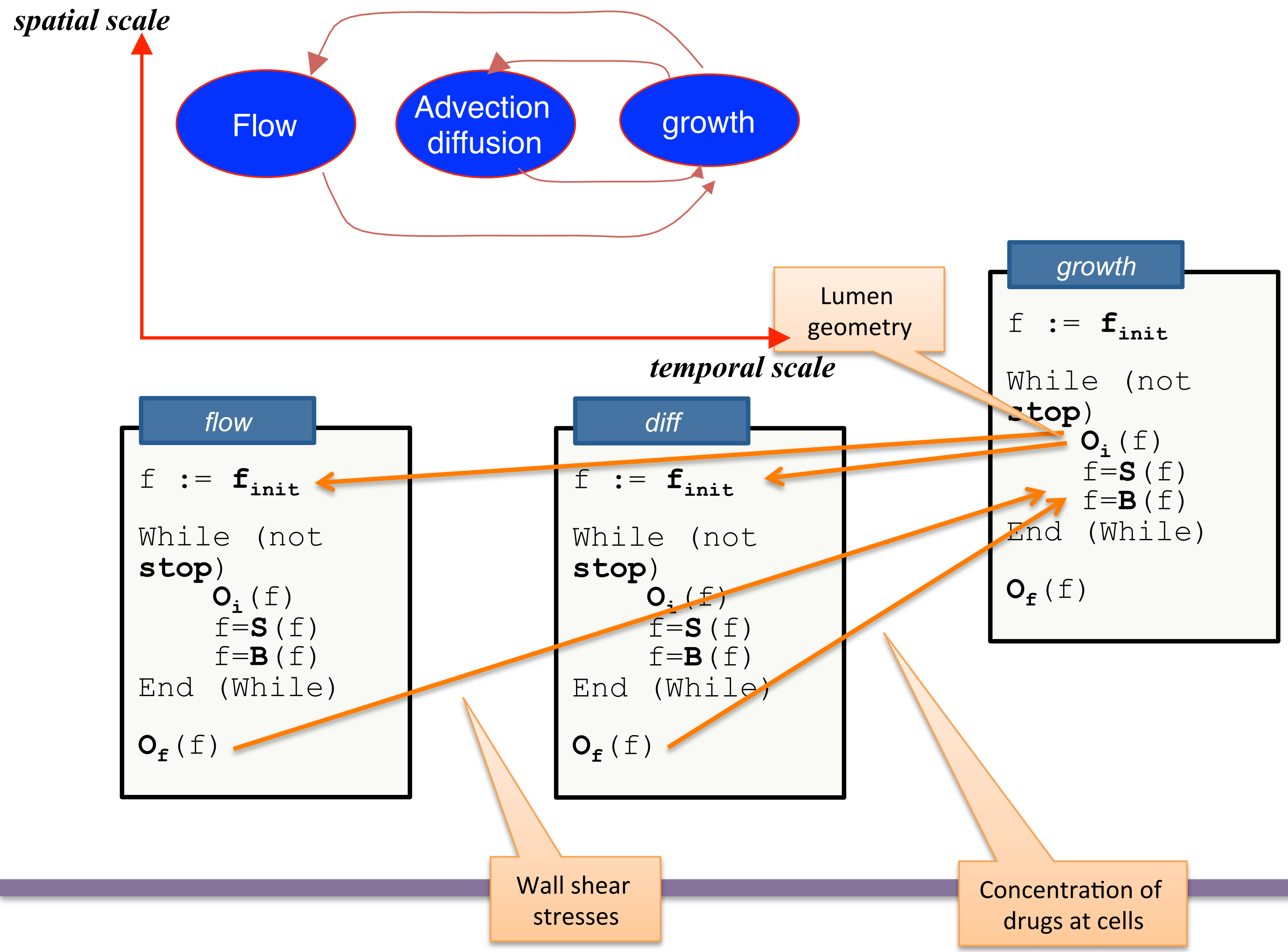
J.A. Kaandorp and J.E. Kübler: The algorithmic beauty of seaweeds, sponges and corals, Springer-Verlag (Heidelberg, New York) 2001.
Merks, Hoekstra, Kaandorp, Sloot, J. Theoret. Biol. 224, 153-166, 2003
Merks, Hoekstra, Kaandorp, Sloot, J. Theoret. Biol. 228, 559-576, 2004

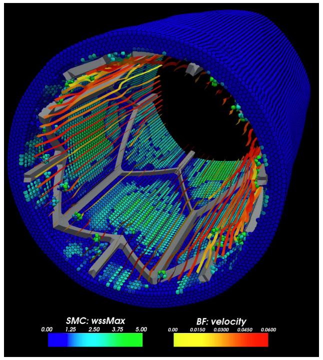
Example, Coral Growth





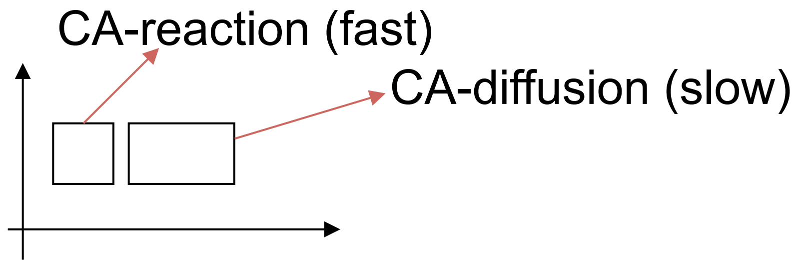
Example, in stent restenosis

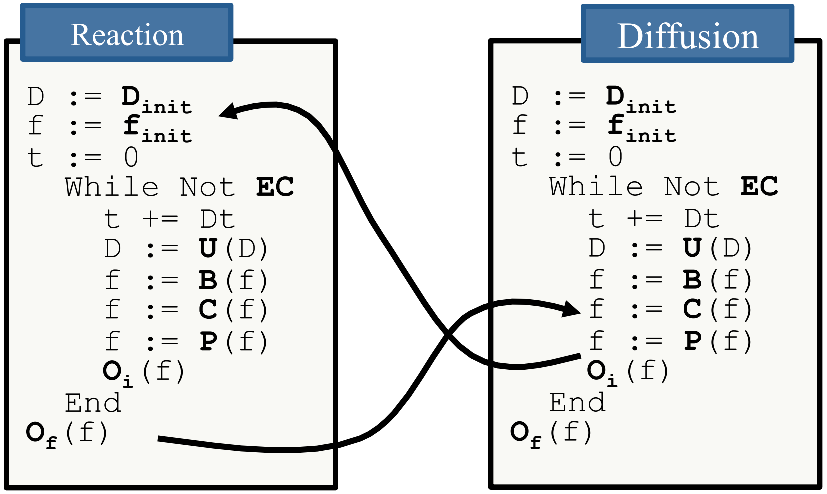




Examples

- **reaction-diffusion**

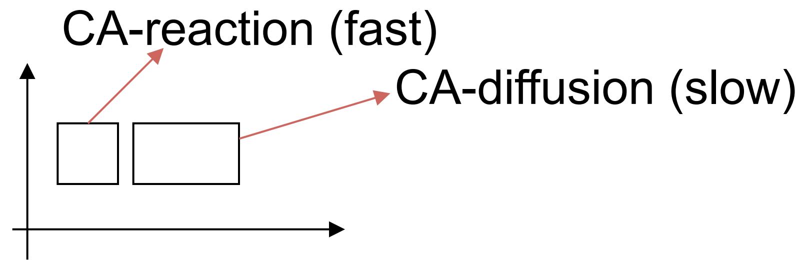




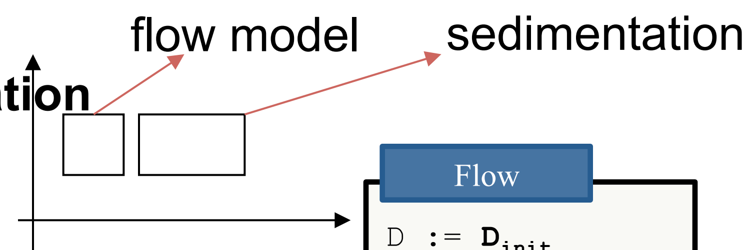
coupling through IC and collision operator

Examples

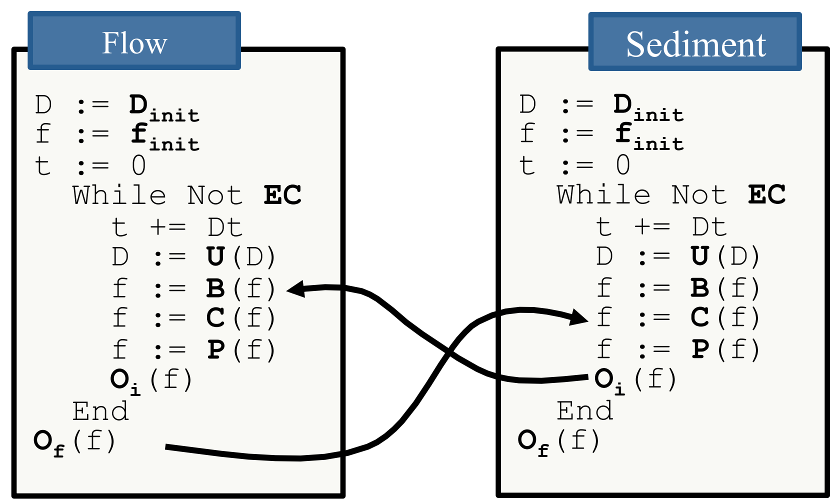
- reaction-diffusion



- **flow - sedimentation**

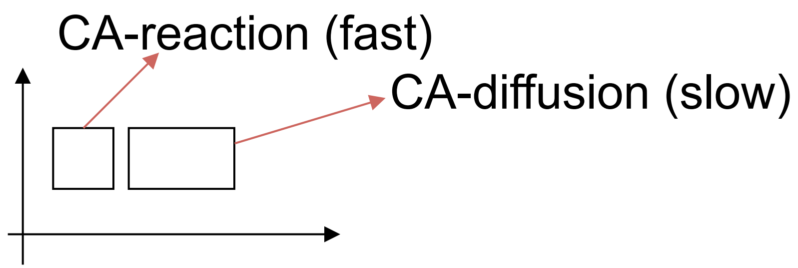


coupling through
collision operator and BC

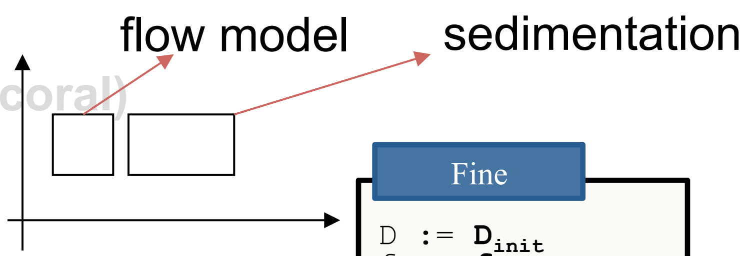


Examples

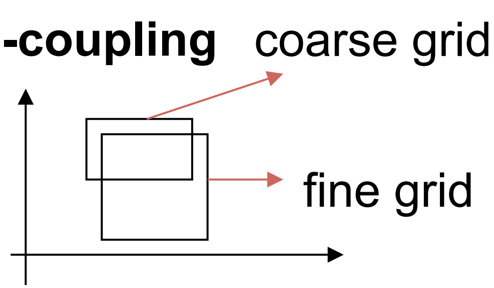
• reaction-diffusion

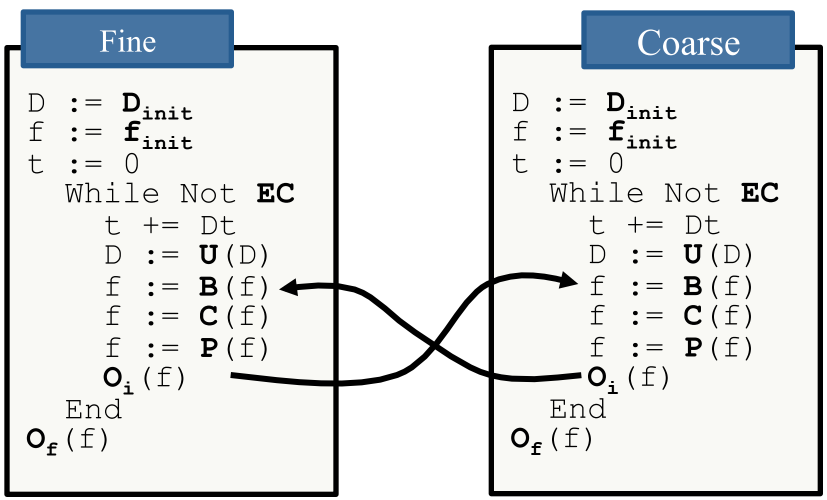


• flow – sedimentation (coral)

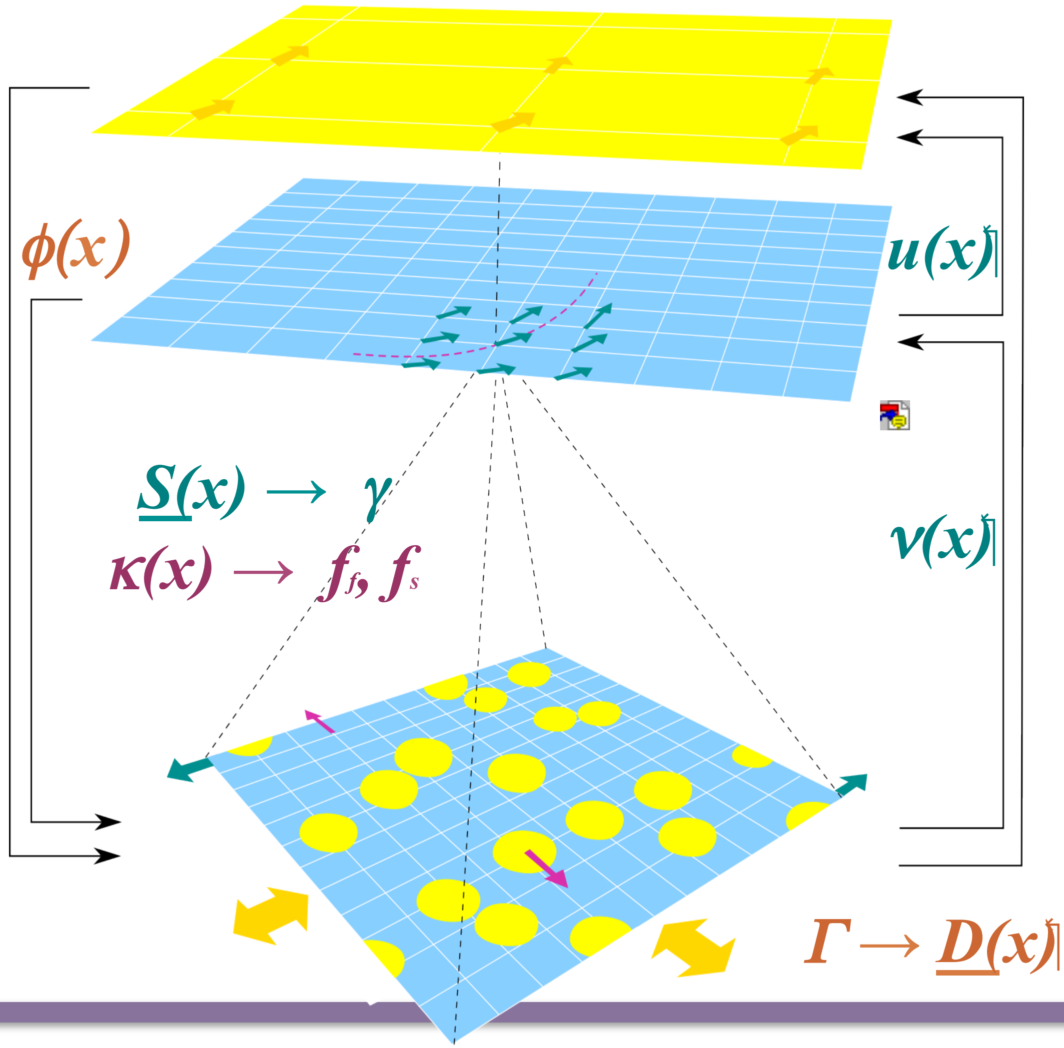


• **grid-coupling**





HMM Suspension



Macro-scale

- LBM fluid with local viscosity ν
- Advection/Diffusion of local particle density ϕ

Micro-scale

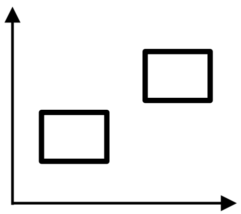
- fully resolved LBM suspension
- sheared by Lees-Edwards BC's

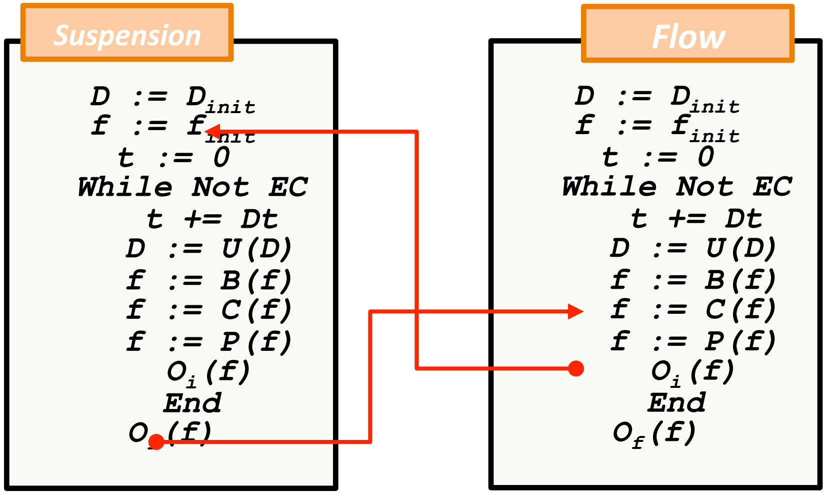
to obtain

- stress → viscosity
- particle flux → diffusivity tensor

Suspension Model

- Configuration
  - Space Scale Separaton, Time Scale Separation,
  - Single-Domain.
- Macroscopic model
  - Lattice Boltzmann fluid model.
- Microscopic model
  - Fully resolved Lattice Boltzmann suspension model.
- Complete micro model on each (or many) macro model lattice point.
- Micro model computes local viscosity.
  - Used in collision operator of macro model.
- The macro model provides the micro-models with boundary conditions
  - shear rate, particle density/distribution





Multiscale Speedup

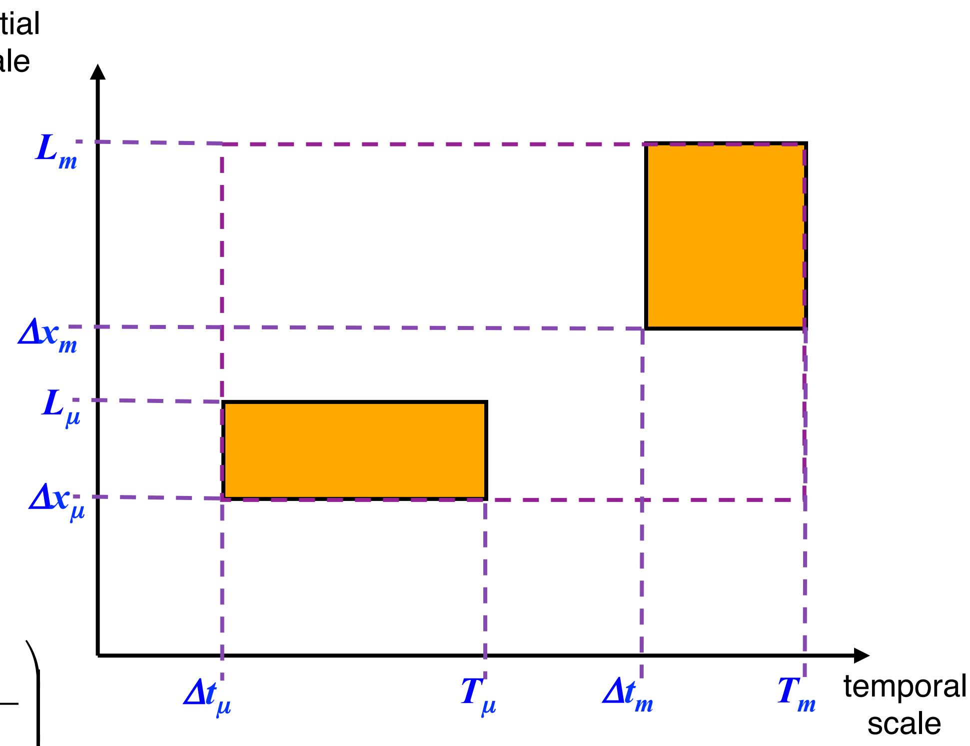
- 1 microscale and one macroscale process
  - At each iteration of the macroscale, the microscale is called
- Execution time full fine scale solver

$$T_{ex}^{full} = \left(\frac{L_M}{\Delta x_\mu}\right)^D \left(\frac{T_M}{\Delta t_\mu}\right)$$

- Execution time for multiscale solver

$$T_{ex}^{multiscale} = \left(\frac{L_M}{\Delta x_M}\right)^D \left(\frac{T_M}{\Delta t_M}\right)\left(\frac{L_\mu}{\Delta x_\mu}\right)^D \left(\frac{T_\mu}{\Delta t_\mu}\right)$$

- Multiscale speedup $S^{multiscale} = \frac{T_{ex}^{full}}{T_{ex}^{multiscale}} = \left(\frac{\Delta x_M}{L_\mu}\right)^D \left(\frac{\Delta t_M}{T_\mu}\right)$

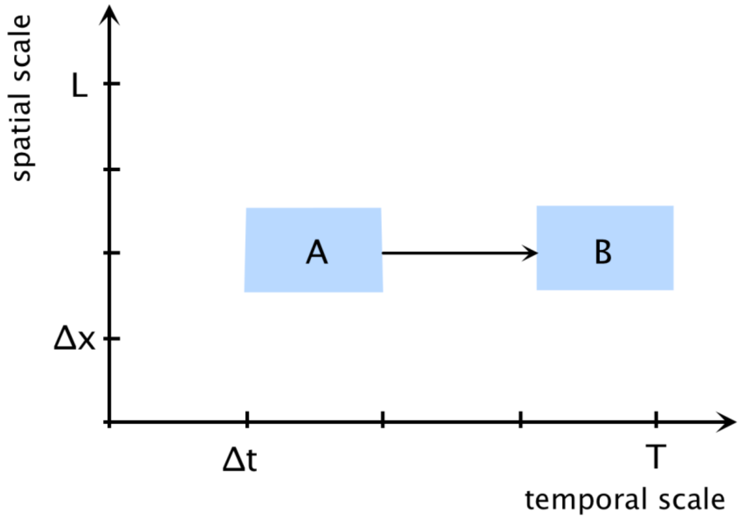


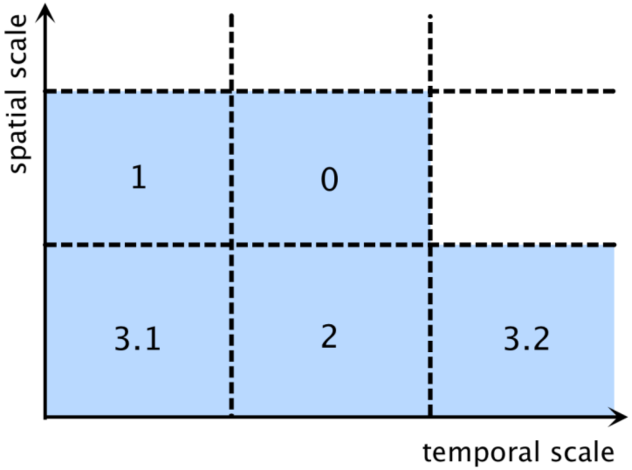
Coupling templates (1)

- Operators O_i and O_f may send observations
- Operators f_{init}, B, and S may receive
- Specify coupling between submodels A and B using operators
- Substitute S with B if interaction is multi-domain

	Name	*Coupling template*	*Temporal scale*
1.	interact	$O_i^A \rightarrow S^B$	overlap
2.	call	$O_i^A \rightarrow f_{init}^B$	*A* larger than *B*
3.	release	$O_f^B \rightarrow S^A$	*A* larger than *B*
4.	dispatch	$O_f^A \rightarrow f_{init}^B$	any

Coupling templates (2)





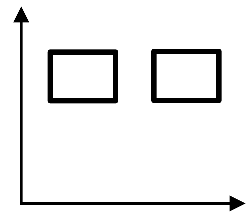
	Name	*Coupling template*	*Temporal scale*
1.	interact	$\mathbf{O}_i^A \rightarrow \mathbf{S}^B$	overlap
2.	call	$\mathbf{O}_i^A \rightarrow \mathbf{f}_{init}^B$	*A* larger than *B*
3.	release	$\mathbf{O}_f^B \rightarrow \mathbf{S}^A$	*A* larger than *B*
4.	dispatch	$\mathbf{O}_f^A \rightarrow \mathbf{f}_{init}^B$	any

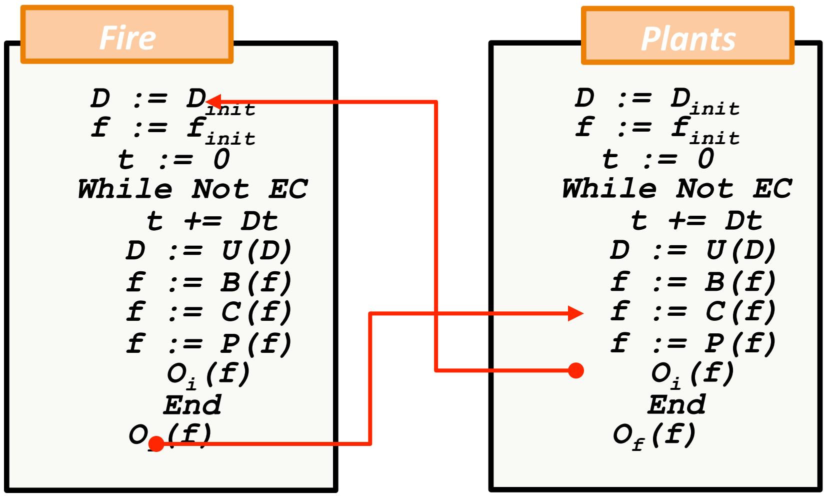
Coupling through SEL operators

- The Submodel-Execution-Loop (SEL) gives a generic way of implementing the different couplings.
- Results in Coupling Templates

Example 1
Savannah-Forest-Fire interactions

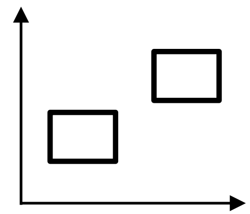
- Configuration
  - Space Scale Overlap, Time Scale Separation,
  - Single-Domain.
- Based on botanical studies of a Ivory Coast natural reserve.
- Plant model: Ecological model leading to a forest climax
  - $\Delta t_P = O(y)$.
- Fire model: Each year (human cause) fire propagate and burn
  - plants $\Delta t_F = O(h)$.
- Fire burns young trees but is diminished by dense vegetation.

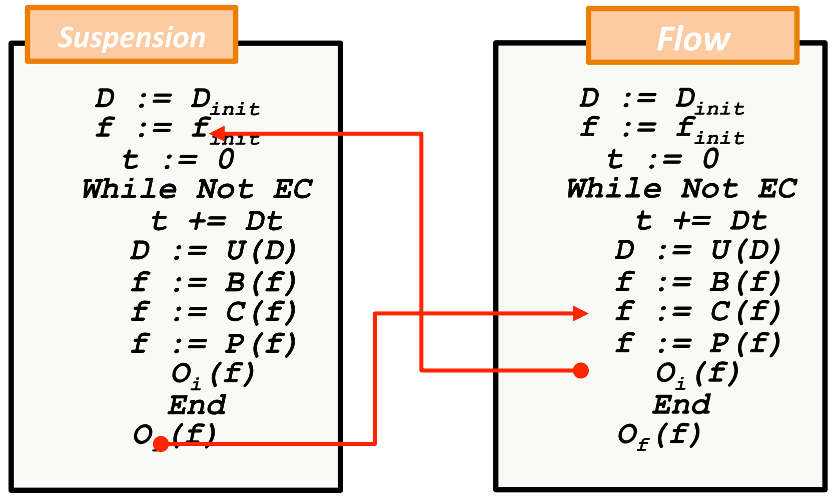




Example 2
Suspension Model

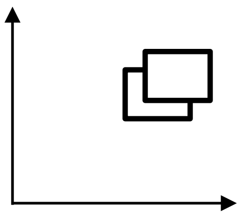
- Configuration
  - Space Scale Separaton, Time Scale Separation,
  - Single-Domain.
- Macroscopic model
  - Lattice Boltzmann fluid model.
- Microscopic model
  - Fully resolved Lattice Boltzmann suspension model.
- Complete micro model on each (or many) macro model lattice point.
- Micro model computes local viscosity.
  - Used in collision operator of macro model.
- The macro model provides the micro-models with boundary conditions
  - shear rate, particle density/distribution

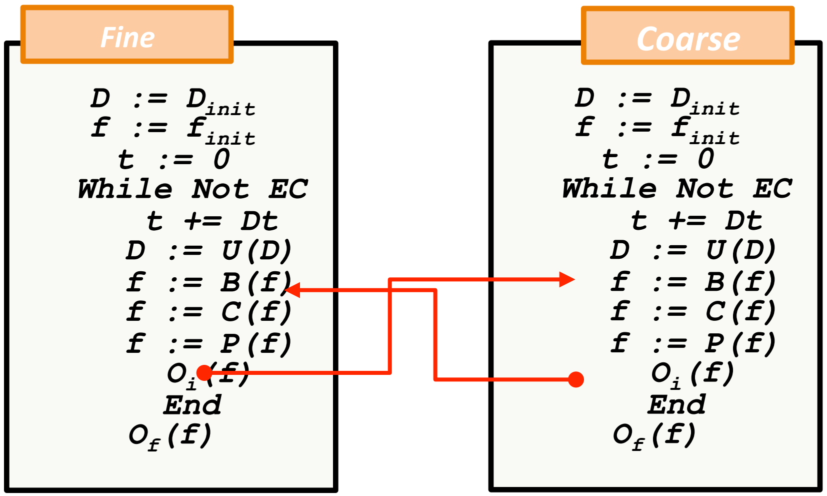




Example 3
Grid Refinement

- Configuration
  - Space Scale partial overlap, Time Scale partial overlap,
  - Multi-Domain.
- Classical Grid Refinement Problem
- A single model is split in two computation domains: coarse and fine
- Usually time steps and grid spacing are connected via
  - $\Delta t_C = n\Delta t_F$ but $\Delta t_C << T_F$.
  - $\Delta x_C = n\Delta x_F$.
- This implies an interpolation/ normalisation of exchanged data.





Classification of systems

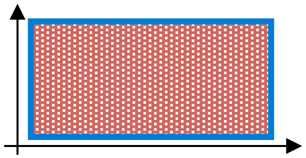
- single-Domain (sM) or multi-Domain (mD)
- Relation on the Scale Separation Map

	Time Overlap		Time Separation	
	Single Domain	**Multi Domain**	**Single Domain**	**Multi Domain**
Space Overlap	Coupling through collision operator. *Snow transport, diffusion/ advection, ...*	Coupling through boundary condition. *Fluid structure, grid refinement, ...*	Coupling through collision operator. *Forest-Savannah-Fire interactions*	Coupling through boundary, initial conditions. *Coral Growth, ...*
	Single Domain	**Multi Domain**	**Hierarchical Coupling**	
Space Separation	Coupling through collision operator. *Algae-Water ecological model, ...*	Coupling through boundary condition. *Wave propagation in two media, ...*	Coupling through collision operator and initialization. *Suspension Fluid, ...*	
			"Physics-Biology Coupling" Coupling through boundary conditions and initialization. *Oscillating blood flow and endothelial cells, ..*	

CXA Formalism

algorithm

- **mathematical description:** $A(\Delta x, \Delta t, X, T, \mathbf{F}, \Phi, \mathbf{u})$

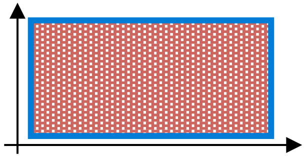


CXA Formalism

algorithm

- **mathematical description:** $A(\Delta x, \Delta t, X, T, \mathbf{F}, \Phi, \mathbf{u})$

update rule: $f^{n+1} = \Phi(\mathbf{u}) f^n \quad \Phi : \mathbf{F} \rightarrow \mathbf{F}$



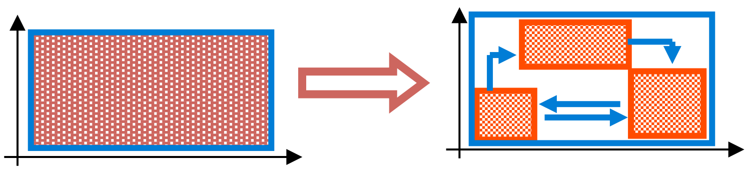
CXA Formalism

algorithm

- **mathematical description:** $A(\Delta x, \Delta t, X, T, \mathbf{F}, \Phi, \mathbf{u})$

update rule: $f^{n+1} = \Phi(\mathbf{u}) f^n \quad \Phi : \mathbf{F} \rightarrow \mathbf{F}$

- **scale splitting:**



$\mathbf{F}_1 \times \mathbf{F}_2 \longrightarrow$

$f_1^{n+1} = \Phi_1(\mathbf{u}_1) f_1^n$

$f_2^{n+1} = \Phi_2(\mathbf{u}_2) f_2^n$

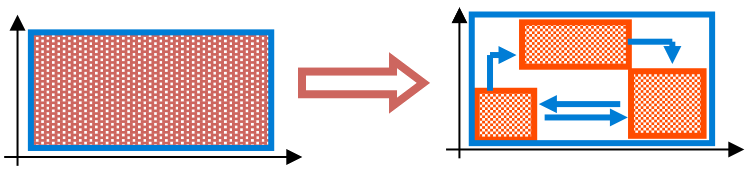
CXA Formalism

algorithm

- **mathematical description:** $A(\Delta x, \Delta t, X, T, \mathbf{F}, \Phi, \mathbf{u})$

update rule: $f^{n+1} = \Phi(\mathbf{u}) f^n \qquad \Phi : \mathbf{F} \rightarrow \mathbf{F}$

- **scale splitting:**



$$\mathbf{F}_1 \times \mathbf{F}_2 \longrightarrow \begin{cases} f_1^{n+1} = \Phi_1(\mathbf{u}_1) f_1^n \\ f_2^{n+1} = \Phi_2(\mathbf{u}_2) f_2^n \end{cases}$$

- **coupling:** $\Phi_1(\mathbf{u}_1) = P \circ C(\mathbf{u}_1) \circ B(\mathbf{u}_1) \qquad \mathbf{u}_1 = \mathbf{u}_1(f_2)$

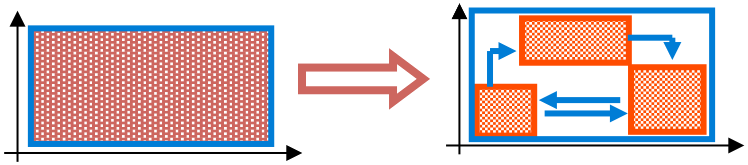
CXA Formalism

algorithm

- **mathematical description:** $A(\Delta x, \Delta t, X, T, \mathbf{F}, \Phi, \mathbf{u})$

update rule: $f^{n+1} = \Phi(\mathbf{u}) f^n$ $\quad \Phi : \mathbf{F} \rightarrow \mathbf{F}$

- **scale splitting:**



$$\mathbf{F}_1 \times \mathbf{F}_2 \longrightarrow \begin{cases} f_1^{n+1} = \Phi_1(\mathbf{u}_1) f_1^n \\ f_2^{n+1} = \Phi_2(\mathbf{u}_2) f_2^n \end{cases}$$

- **coupling:** $\Phi_1(\mathbf{u}_1) = P \circ C(\mathbf{u}_1) \circ B(\mathbf{u}_1)$ $\quad \mathbf{u}_1 = \mathbf{u}_1(f_2)$

- **multiscale technique:**
  - time splitting
  - coarsening
  - amplification

$$\Phi \longrightarrow (\Phi_1, \Phi_2)$$

Time Splitting

- Assume we have a sD problem with the following SEL

$$S_{\Delta t} = S_{\Delta t}^{(1)} S_{\Delta t}^{(2)}$$

- Then if $S^{(1)}$ *acts on a longer time than* $S^{(2)}$ *we may want to approximate*

$$[S_{\Delta t}]^M = S_{M\Delta t}^{(1)} [S_{\Delta t}^{(2)}]^M$$

Coarse graining

- This strategy consists in expressing a sD problem as

$$[S_{\Delta x}]^n = \Gamma^{-1}[S_{2\Delta x}]^{n/2}\Gamma$$

- Where Γ is a projection operator

Amplification

- Here we consider a process acting at low intenstity but for a long time, in a time periodic environment (e.g. growth processes in a pulsatile flow). We have two coupled processes which are iterated n >>1 time

$$[S^{(1)}]^n \text{ and } [S^{(2)}(k)]^n$$

- Where k expresses the intensity of process $C^{(2)}$. If the period of process $C^{(2)}$ is $m << n$, we can approximate the above evolution as

$$[S^{(1)}]^m \text{ and } [S(k')]^m$$

- with $k' = (n/m)k$, for a linear process.

Scale Splitting Error

$$f^{n+1} = \Phi(\mathbf{u}) f^n$$

$$f_1^{n+1} = \Phi_1(\mathbf{u}_1) f_1^n$$
$$f_2^{n+1} = \Phi_2(\mathbf{u}_2) f_2^n$$

$$\begin{array}{ccc} f \in \mathbf{F} & \xrightarrow{\Pi = (\Pi_1, \Pi_2)} & (f_1, f_2) \in \mathbf{F}_1 \times \mathbf{F}_2 \\ \Phi \downarrow & & \Phi_1, \Phi_2 \downarrow \\ \mathbf{F} & & \mathbf{F}_1 \times \mathbf{F}_2 \end{array}$$

Scale Splitting Error

$$f^{n+1} = \Phi(\mathbf{u}) f^n$$

$$f_1^{n+1} = \Phi_1(\mathbf{u}_1) f_1^n$$
$$f_2^{n+1} = \Phi_2(\mathbf{u}_2) f_2^n$$

$$f \in \mathbf{F} \xrightarrow{\Pi = (\Pi_1, \Pi_2)} (f_1, f_2) \in \mathbf{F}_1 \times \mathbf{F}_2$$

$\Phi \downarrow$ $\qquad$ $\Phi_1, \Phi_2 \downarrow$

$\mathbf{F} \xrightarrow{\mathbf{O}(f)} \mathbf{\Omega} \xleftarrow{\mathbf{O}_{\mathrm{CxA}}(f_1, f_2)} \mathbf{F}_1 \times \mathbf{F}_2$

observable
(project on quantities of interest)

- **formal scale splitting error**
  - **difference between observed results (in opportune norm)**
  - **case by case, use the properties of problem and algorithms**

We have detailed results, but not discussed here, see e.g. A. Caiazzo, J.-L. Falcone, B. Chopard, and A.G. Hoekstra, Asymptotic analysis of Complex Automata models for reaction-diffusion systems, Applied Numerical Mathematics 59, 2023-2034, 2009.

MULTISCALE COMPUTING

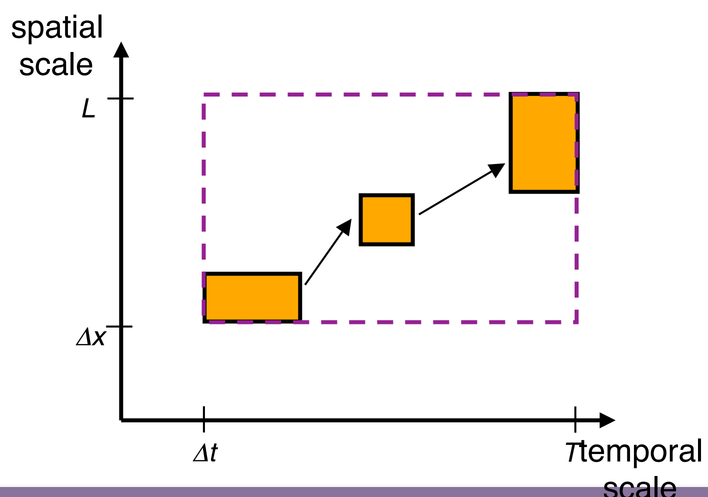
So, what about multiscale computing?

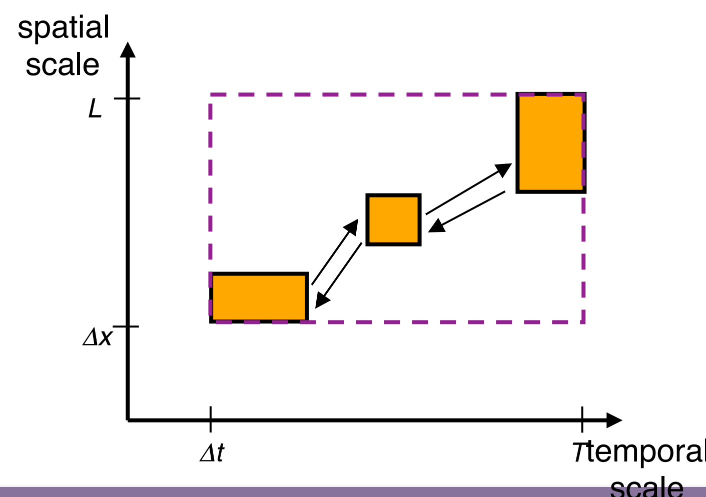
- Inherently hybrid models are best serviced by different types of computing environments
- When simulated in three dimensions, they usually require large scale computing capabilities.
- Such large scale hybrid models require a distributed computing ecosystem, where parts of the multiscale model are executed on the most appropriate computing resource.
- (Distributed) Multiscale Computing

Two Multiscale Computing paradigms

- Loosely Coupled
  - One single scale model provides input to another
  - Single scale models are executed once
  - workflows

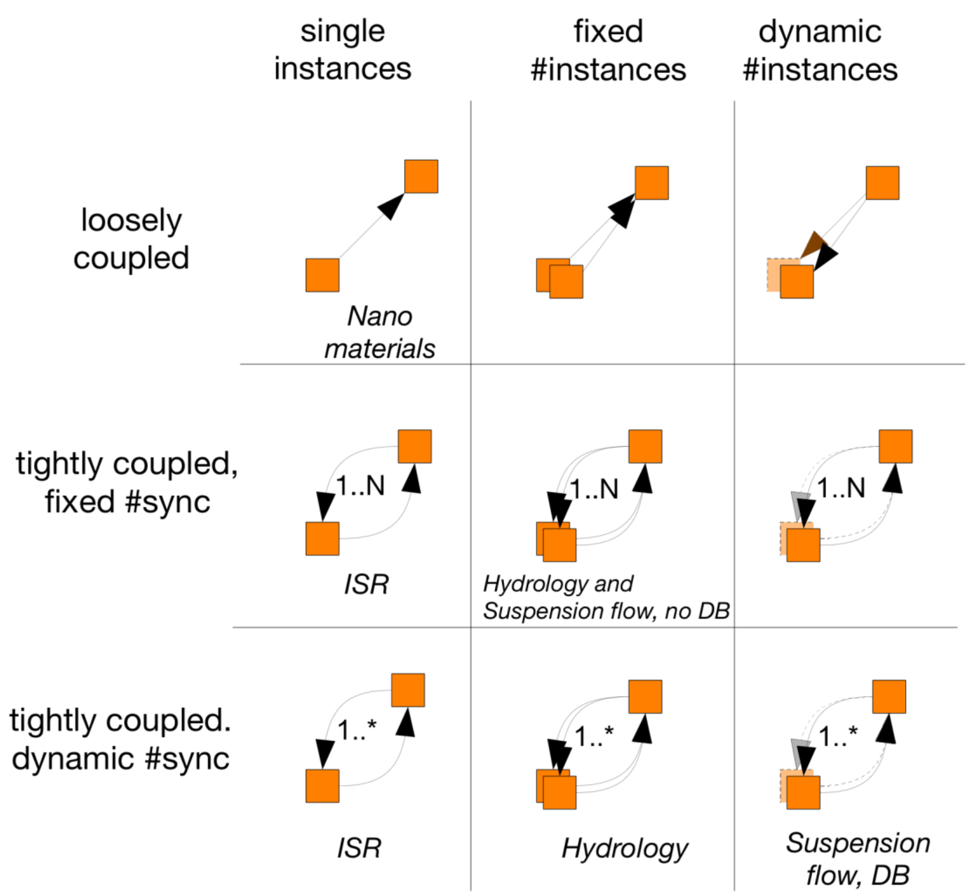
- Tightly Coupled
  - Single scale models call each other in an iterative loop
  - Single scale models may execute many times
  - Dedicated coupling libraries are needed





Coupling topology

- Loosely coupled model: acyclic coupling topology
- Dynamic or static:
  - Number of synchronization points
  - Number of submodel instances



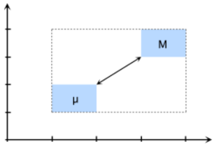
Model description

- Submodels
- Couplings
- Full network
  - how many submodels are instantiated?
  - which instantances are they coupled to?
    - coupling topology

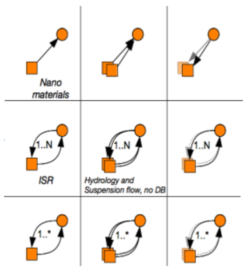
Overview

- ✓ Modeling
  - ✓ Functional decomposition
  - ✓ Coupling topology
- Automation
  - Specification
  - Analysis
- Distributed computing

SSM



↓



Coupling topology

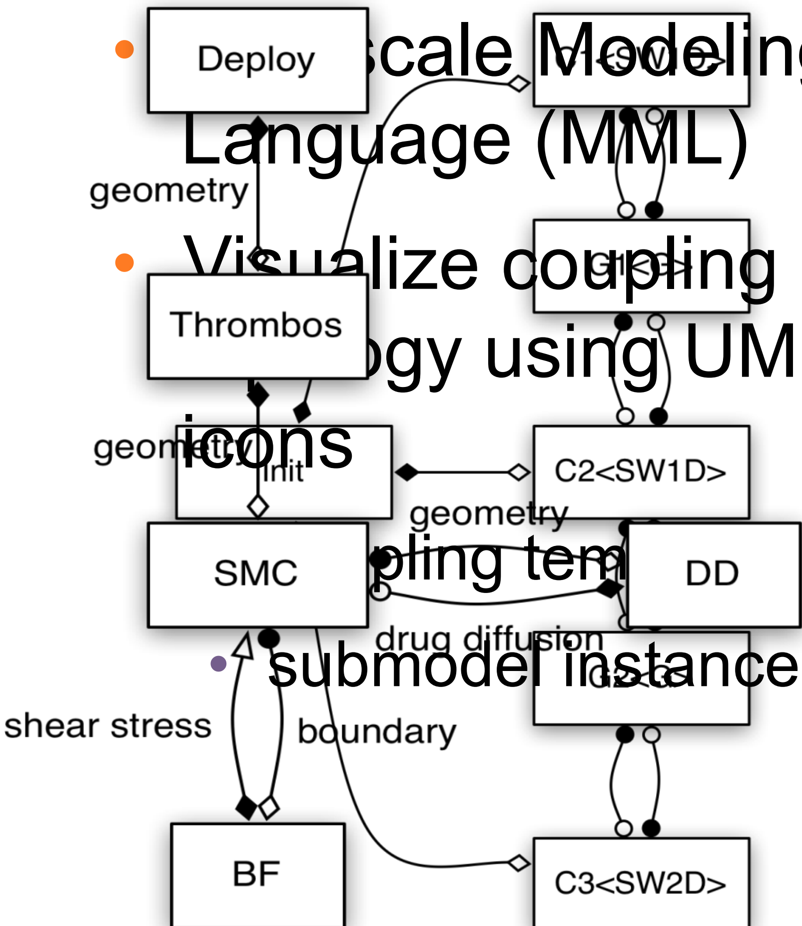
↓

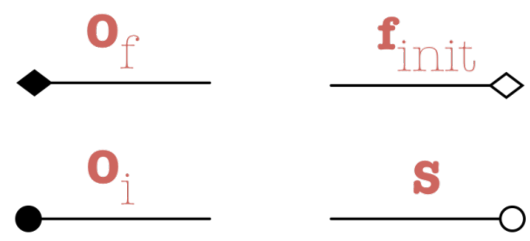
Automation

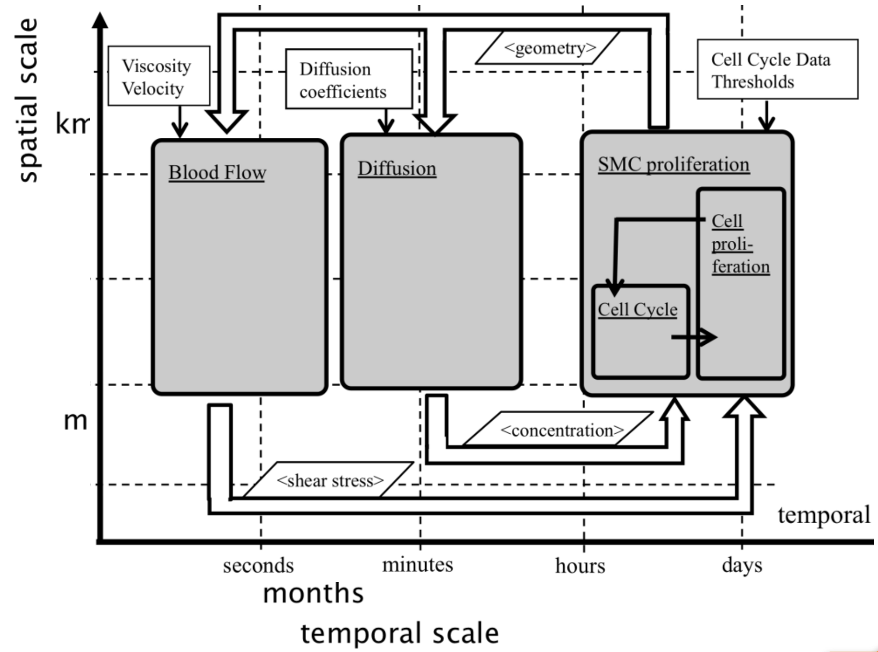
- Use the multiscale model description to do computing
- Machine-readable specification
- Interpretation and analysis of the specification

Specification

- …scale Modeling Language (MML)
- Visualize coupling …ogy using UML icons
- …pling tem…
  - submodel instances







J.-L. Falcone, B. Chopard, and A. G. Hoekstra. MML: towards a multiscale modeling language. Procedia Computer Science. 1(1):819–826, 2010

Machine readable format

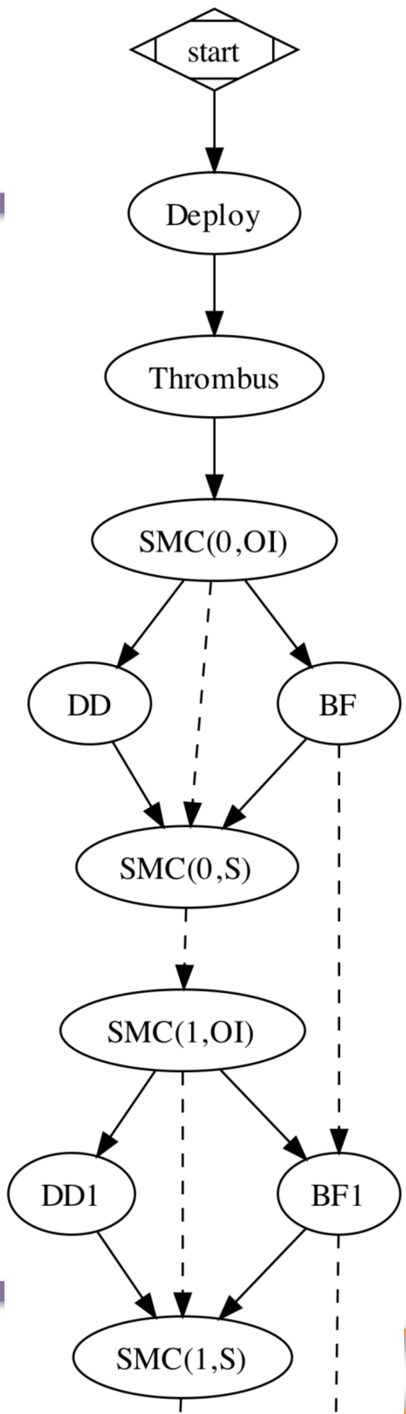
- XML format of MML: xMML
- Automation on the basis of model specification
  - for instance: generate skeleton code

xMML example

```
<model id="ISR2D" name="In-stent restenosis 2D” xmml_version="0.1” xmlns:xi=
http://www.w3.org/2001/Xinclude>
    <description>A model of the process that occurs in the artery after stenting.</description>
    <definitions>
        <xi:include href="isr_meta.xml#xpointer(/metadata/*)"/>
        <submodel id="BF" name="Blood flow" stateful="optional”>
            <timescale delta="1E-7" max="1"/>
            <spacescale delta=”10 um" max=”1 mm"/>
            <spacescale delta="1E-5" max="1E-3"/>
            <ports>
                <in id="state_start" operator="finit" type="state"/>
                <in id="boundary" operator="S" datatype="latticeInt"/>
                <out id="shear_stress" operator="Of” datatype="latticeDouble"/>
                <out id="state_end" operator="Of" type="state"/>
            </ports>
        </submodel> ... </definitions>
    <topology>
        <instance id="ic" submodel="INIT" domain="artery"/>
        <instance id="bf" submodel="BF" domain="artery.blood"/>
        <instance id="smc" submodel="SMC" domain="artery.tissue"/>

        <coupling name="geometry" from="ic.cells" to="smc.initial_positions"/>
        <coupling name="geometry" from="smc.cell_positions" to="bf.boundary"/>
        <coupling name="shear_stress" from="bf.shear_stress"to="smc.shear_stress"/>
    </topology>
```

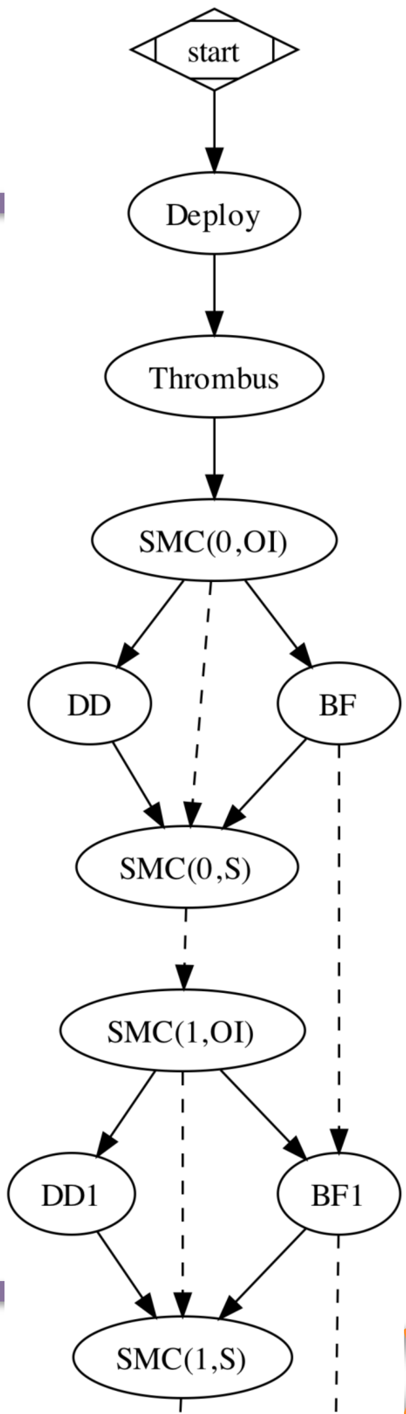
Analysis

- Determine execution order based on xMML and the SEL: task graph
  - Directed acyclic graph
  - Schedule submodels based on dependencies
  - Estimate run time and communication costs
  - Detect deadlocks
  - SEL crucial to the ordering
- Possible to map to workflow or Petri nets or process calculus



Task graph

- Operators O_i and S/B and each iteration in a separate node
  - Transformation to reduce nodes possible and performed already
- Edges based on communication
  - thus control, so they indicate dependencies
  - dashed edges are stateful transitions
    - same instance could be scheduled on a different machine



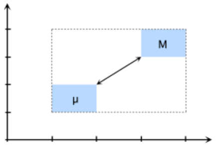
Implementation

- Kernels: single scale model unaware of other submodels
  - Facilitates substitution with other submodels
- Couplings
  - Transfer data according to the SEL
  - Write filters to manage scale differences of sending and receiving submodel

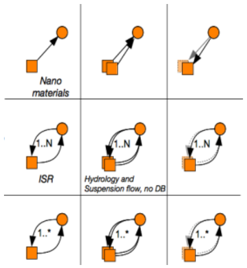
Overview

- ✓ Modeling
  - ✓ Functional decomposition
  - ✓ Coupling topology
- ✓ Automation
  - ✓ Specification
  - ✓ Analysis
- Distributed computing

SSM



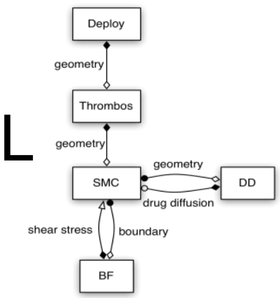
↓



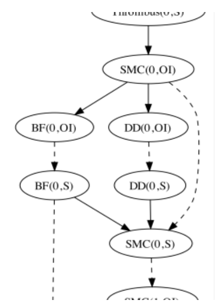
Coupling topology

↓

(x)MML



↓



Task graph

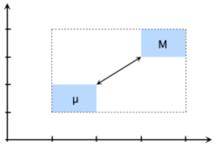
Distributed computing

- Handle resources that must be scheduled
- Minimize communicational overhead
- Minimize run-time dependencies (cross-scheduling)
- Utilize heterogeneous hardware
- If a task graph is available then educated guesses on schedules can be made
- If the coupling topology is dynamic
  - scheduling only with a limited lookahead
  - model–schedule interaction is needed

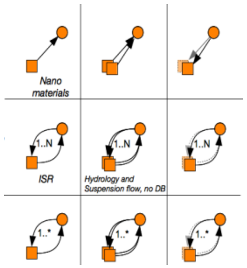
Overview

- ✓ Modeling
  - ✓ Functional decomposition
  - ✓ Coupling topology
- ✓ Automation
  - ✓ Specification
  - ✓ Analysis
- ✓ Distributed computing

SSM



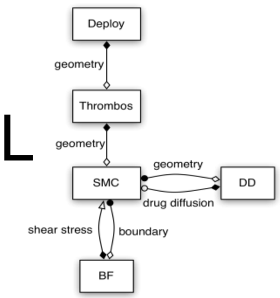
↓



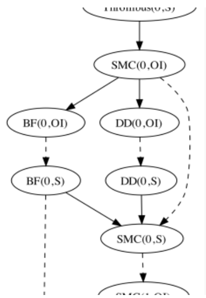
Coupling topology

↓

(x)MML



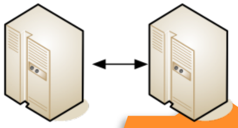
↓



Task graph

↓

Scheduling



Bibliography

- Borgdorff, J., J.-L. Falcone, E. Lorenz, C. Bona-Casas, B. Chopard, and A.G. Hoekstra, *Foundations of distributed multiscale computing: Formalization, specification, and analysis.* Journal of Parallel and Distributed Computing, 2013. **73**(4): p. 465-483.
- Chopard, B., J.-L. Falcone, A. Hoekstra, and J. Borgdorff, *A Framework for Multiscale and Multiscience Modeling and Numerical Simulations*, in *Unconventional Computation*, C. Calude, et al., Editors. 2011, Springer Berlin / Heidelberg. p. 2-8.
- Caiazzo, A., D. Evans, J.-L. Falcone, J. Hegewald, E. Lorenz, B. Stahl, D. Wang, J. Bernsdorf, B. Chopard, J. Gunn, R. Hose, M. Krafczyk, P. Lawford, R. Smallwood, D. Walker, and A. Hoekstra, *A Complex Automata approach for in-stent restenosis: Two-dimensional multiscale modelling and simulations.* Journal of Computational Science, 2011. **2**(1): p. 9-17.
- Borgdorff, J., C. Bona-Casas, M. Mamonski, K. Kurowski, T. Piontek, B. Bosak, K. Rycerz, E. Ciepiela, T. Gubala, D. Harezlak, M. Bubak, E. Lorenz, and A.G. Hoekstra, *A Distributed Multiscale Computation of a Tightly Coupled Model Using the Multiscale Modeling Language.* Procedia Computer Science, 2012. **9**(0): p. 596-605.
- Groen, D., J. Borgdorff, C. Bona-Casas, J. Hetherington, R.W. Nash, S.J. Zasada, I. Saverchenko, M. Mamonski, K. Kurowski, M.O. Bernabeu, A.G. Hoekstra, and P.V. Coveney, *Flexible composition and execution of high performance, high fidelity multiscale biomedical simulations.* Interface Focus, 2013. **3**(2).
- Tahir, H., A.G. Hoekstra, E. Lorenz, P.V. Lawford, D.R. Hose, J. Gunn, and D.J.W. Evans, *Multi-scale simulations of the dynamics of in-stent restenosis: impact of stent deployment and design.* Interface Focus, 2011. **1**(3): p. 365-373.
- Lorenz, E. and A.G. Hoekstra, *Heterogeneous Multiscale Simulations of Suspension Flow.* Multiscale Modeling & Simulation, 2011. **9**(4): p. 1301-1326.
- Sloot, P.M.A. and A.G. Hoekstra, *Multi-scale modelling in computational biomedicine.* Brief Bioinform, 2010. **11**(1): p. 142-152.
- Hoekstra, A., A. Caiazzo, E. Lorenz, J.-L. Falcone, and B. Chopard, *Complex Automata: Multi-scale Modeling with Coupled Cellular Automata*, in *Simulating Complex Systems by Cellular Automata*, A.G. Hoekstra, J. Kroc, and P.M.A. Sloot, Editors. 2010, Springer Berlin / Heidelberg. p. 29-57.
- Falcone, J.-L., B. Chopard, and A. Hoekstra, *MML: towards a Multiscale Modeling Language.* Procedia Computer Science, 2010. **1**(1): p. 819-826.
- Caiazzo, A., J.-L. Falcone, B. Chopard, and A.G. Hoekstra, *Asymptotic analysis of Complex Automata models for reaction-diffusion systems.* Applied Numerical Mathematics, 2009. **59**(8): p. 2023-2034.
- Hegewald, J., M. Krafczyk, J. Tölke, A. Hoekstra, and B. Chopard, *An Agent-Based Coupling Platform for Complex Automata*, in *Computational Science – ICCS 2008*, M.T. Bubak, et al., Editors. 2008, Springer: Berlin, Heidelberg. p. 227-233.
- Evans, D.J.W., P.V. Lawford, J. Gunn, D. Walker, D.R. Hose, R.H. Smallwood, B. Chopard, M. Krafczyk, J. Bernsdorf, and A. Hoekstra, *The application of multiscale modelling to the process of development and prevention of stenosis in a stented coronary artery.* Phil.Trans. R. Soc. A, 2008. **366**: p. 3343-3360.
- Hoekstra, A.G., E. Lorenz, J.-L. Falcone, and B. Chopard, *Towards a Complex Automata Framework for Multi-scale Modeling.* International Journal for Multiscale Computational Engineering, 2007. **5**(6): p. 491-502.