

Toward a Multi-Level Calculus for Cellular Modeling and Simulation

Rayus Kuplicki

12/6/08

Process Calculus

1010110101011010101110000010110101100101010111111010010101011110101011100101011010010000101101

- Formal model for concurrent systems
- Processes communicate over named channels
- Supports modular specification, composition, and proof mechanisms
- Governed by reduction rules

$$a.P \mid a.Q \mid b.R \xrightarrow{a} P \mid Q \mid b.R \quad (\text{Parallel})$$

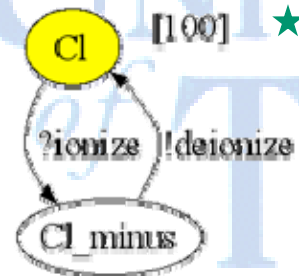
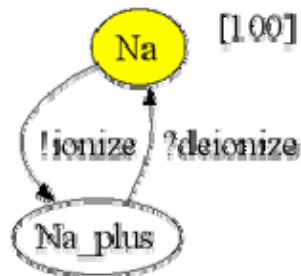
$$(a.P + b.Q + c.R) \mid a.S \xrightarrow{a} P \mid S \quad (\text{Choice})$$

$$a(b).P \mid \bar{a}\langle c \rangle.Q \rightarrow \{c/b\}P \mid Q \quad (\text{Comm.})$$

SPiM

1010110101011010101110000010110101100101010111111010010101011110101011100101011010010000101101

- Stochastic Pi-Calculus: affixes reaction rates to communication channels
- Stochastic Pi Machine



```
new ionize@100.0 : chan      ★
new deionize@10.0 : chan
```

```
let Na() = !ionize; Na_plus()
and Na_plus() = ?deionize; Na()
```

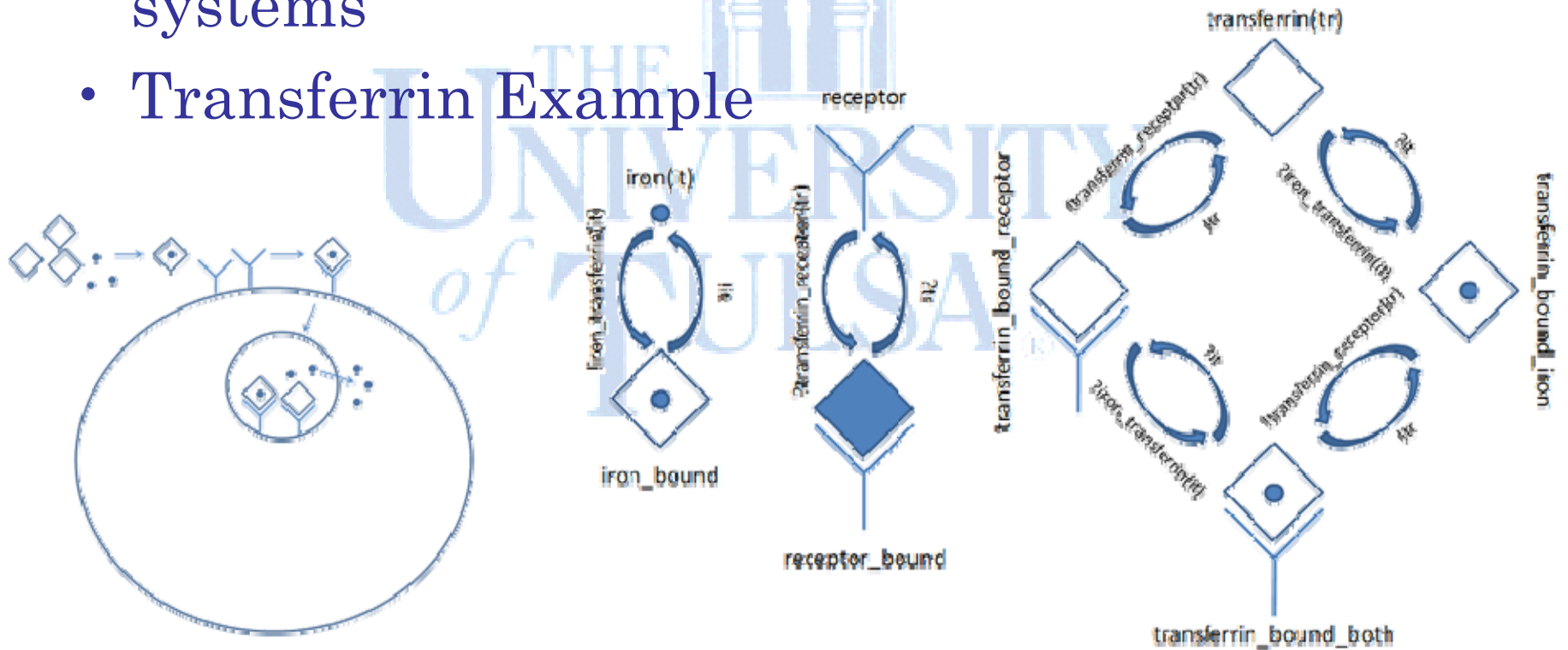
```
let Cl() = ?ionize; Cl_minus()
and Cl_minus() = !deionize; Cl()
```

★ From <http://research.microsoft.com/~aphillip/spim/Examples.pdf>

Locality and Containment

1010110101011010101110000010110101100101010111111010010101011110101011100101011010010000101101

- SPiM lacks support for locality and containment—critical aspects of biological systems
- Transferrin Example

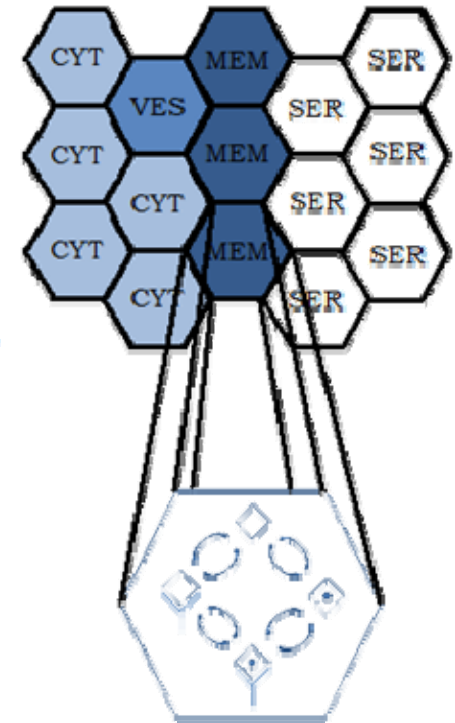




GridSPiM

1010110101011010101110000010110101100101010111111010010101011110101011100101011010010000101101

- First attempt to add locality and containment
- Place SPiM instances in a grid and allow diffusion between grid spaces
- Custom simulations and stochastic rates in each space
- Custom diffusion rates between spaces



Multi-Level Calculus

1010110101011010101110000010110101100101010111111010010101011110101011100101011010010000101101

- Lower level: SPiM specifications and execution
- Upper level: calculus for dynamic containers and localities
- Changes triggered between layers
- Needs scalable, distributable architecture and intuitive specification language