# Concurrency: Languages, Programming and Theory

# – Proofs in $\pi$-Calculus –

# Session 14 – February 4, 2004

Uwe Nestmann

EPFL-LAMP



**Joint Foundations for Lambda & Pi**

- $\alpha$-conversion & substitution
- equivalences
- contexts & congruence

**Lambda**

- semantics using evaluation contexts
- reduction strategies
- various equivalences
- Y recursion



## (Exam) Questions?

**CCS / Pi**

- concurrency primitives
- read & write labeled transition semantics rules
- derivation of transitions
- simulation, mutual & bi- simulation, strong vs. weak
  - definitions
  - distinguishing examples
- modeling exercises/examples w/ and w/out mobility

**Scala**

- transforming calculus into “language”
- transforming “language” into calculus
- representation of “high-level” concurrency primitives



## Exercise: Semaphores

- recall the examples from Session 8
- now, think different !

- informal description of specification and implementation
- model in Pi
- run in Scala
- verify using the ABC