Assure – Summary

- **Assure** finds the location (source & sink) of race conditions, but
- the **programmer** must detect the reasons!

Assure – Practical (on crossi)

- on crossi.hww.de
- cd ~/OpenMP/#nr/pitfalls
  - /opt/KAI/assure40/bin/assureifc → myprog myprog.f90 or
  - assurec → myprog myprog.c or
  - assure++ → myprog myprog.cc
  - /myprog (executing the parallel program on one processor by Assure, detecting all race conditions)
  - assureview (or assureview −v)
- with myprog = **conflict**, **race1**, and **race2**
- Tasks:
  - find the reasons in all 3 examples
  - correct the source (without modifying the numerical semantics)
  - verify again with Assure
  - until Assure tells “no errors”
Assure – Practical (on azusa)

- on azusa.hww.de
- cd ~/OpenMP/#nr/pitfalls
  - assureefc –o myprog myprog.f90 or
  - assurec –o myprog myprog.c or
  - assurec++ –o myprog myprog.cc
  - ./myprog (executing the parallel program on one processor
    by Assure, detecting all race conditions)
  - assureview (or assureview –v)
- with myprog = conflict, race1, and race2
- Tasks:
  - find the reasons in all 3 examples
  - correct the source (without modifying the numerical semantics)
  - verify again with Assure
  - until Assure tells “no errors”

Assure – Practical (on hwwhpn)

- on hwwhpn.hww.de
- cd ~/OpenMP/#nr/pitfalls
  - assuref90 –o myprog myprog.f90 or
  - assurecc –o myprog myprog.c
  - ./myprog (executing the parallel program on one processor
    by Assure, detecting all race conditions)
  - assureview
- with myprog = conflict, race1, and race2
- Tasks:
  - find the reasons in all 3 examples
  - correct the source (without modifying the numerical semantics)
  - verify again with Assure
  - until Assure tells “no errors”