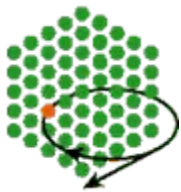# TINE + Windows CE

Andres Pazos
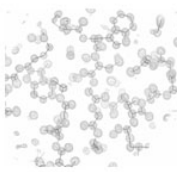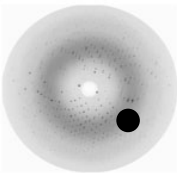## EMBL-Hamburg

Status

- **Present:**
  - Libraries compiled for windows CE 5.0 (works with CE 6.0)
    - tine32
    - Cdi32, TwinCATads, SedPCSimulate
  - Simple tests in a embedded x86 machine
- **Future:**
  - Stress tests
  - Compile for more CPU architectures
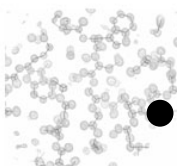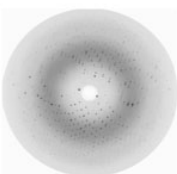 - Migrate to release 4.0

Two Ideas

- ## Make servers
  - Example: TINE server running in a Embedded PC directly connected to the HW
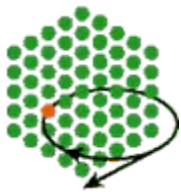- ## Make clients
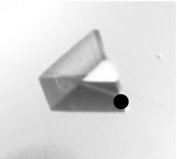 - Example: I have mobile access to my servers from my PDA

- ## Multiplatform compilation: windows Desktop and windows CE
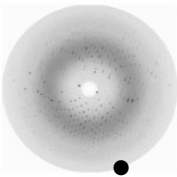
Standard Tools

- For Programming
  - MS eMbedded Visual: C++ & VB
  - MS Visual Studio: C++, VB & C#
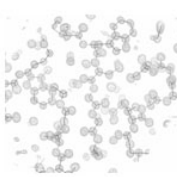 - *you need a SDK for each winCE distribution
- Platform Builder
  - Build your own Windows CE distribution (example: only command line without GUI)
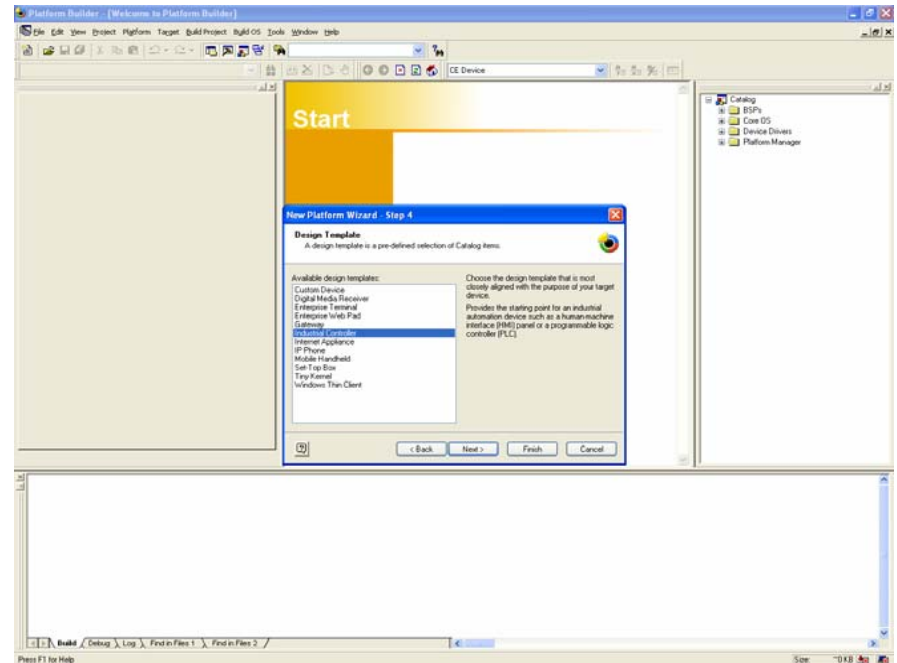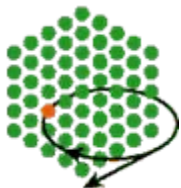 - Build your SDK
- Emulator
  - Advantage: you simulate your hardware
  - Disadvantage: Networking not straight forward
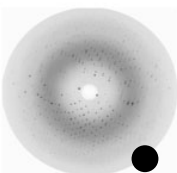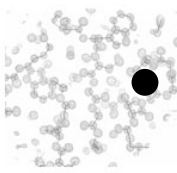- ActiveSync

Other Tools

- Write Java programs for winCE
  - ewe Java Virtual Machine
  - Sun PersonalJava migrates to Java2 MicroEdition (J2ME)
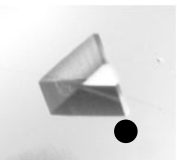 - Waba & SuperWaba
- PythonCE
- From Linux
  - Possible to link GCC with winCE libraries
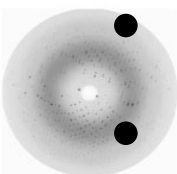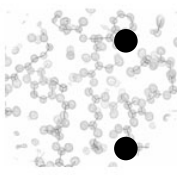
David Michael Ehlen © 2001

Remote tools

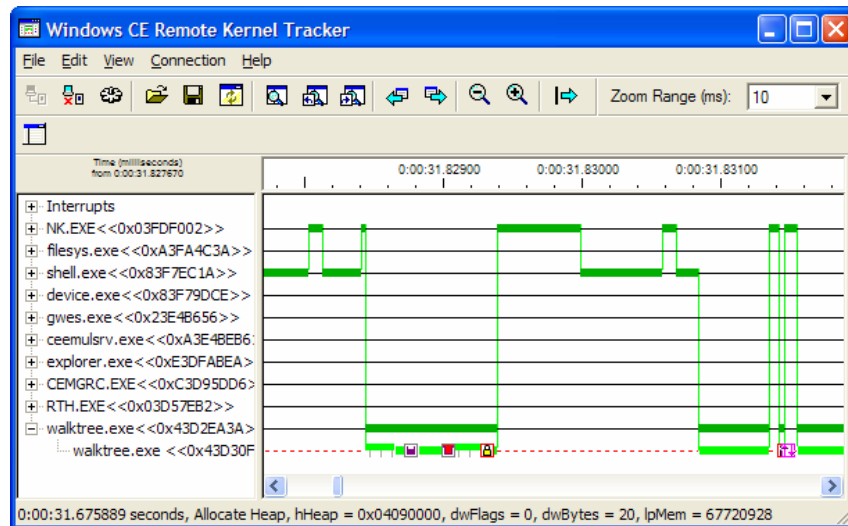- Kernel tracker
- Heap walker
- Call profiler
- Registry editor
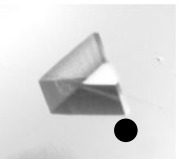- Process viewer
- Performance monitor
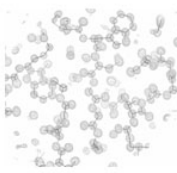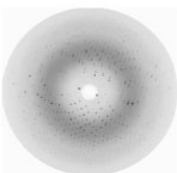- Spy client
- File viewer

Coming back to RTOS topic

- Hard real time possibilities
- Benchmarking from www.dedicated-systems.com

Interrupt period	#interrupts generated	#interrupts serviced	#interrupts lost
20 µs	100 000	99 972	28
23 µs	100 000	99 997	3
25 µs	100 000	100 000	0
28 µs	100 000	100 000	0
26 µs	100 000 000	99 999 999	1
28 µs	100 000 000	100 000 000	0
30 µs	100 000 000	100 000 000	0
27 µs	1 000 000 000	1 000 000 000	0

TINE

David Michael Ehlen © 2001