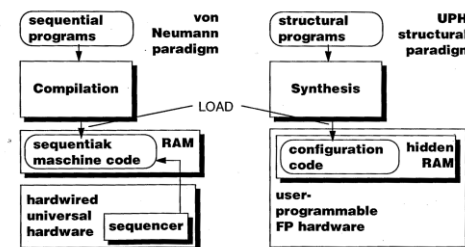


CASHE using a new machine paradigm



CASHE for programmers?

has become feasible:
 (example scenario)

program being implemented on workstation (host) with universal accelerator board (UAB):

- procedure X found to be throughput bottleneck
- accelerator for X compiled into UAB
- interface compiled into UAB and host
- accelerated algorithm X now running on UAB

new paradigm needed

procedural paradigm needed

- von Neumann paradigm
 - deterministic: traceable, debuggable
 - does not support soft hardware
 - very tight sequencer / ALU coupling
 - does not support "soft" hardware
- we need another way of sequencing
 - instruction sequencer not feasible
 - what about data-driven ?



what new paradigm is feasible?

- ❑ data flow machine:
 - driven by "firing" (arbitration)
 - not traceable, nor debuggable
 - no familiarity with data flow languages
- ❑ procedural data-driven paradigm needed:
 - deterministically data-driven
 - traceable, debuggable
 - familiarity with procedural style programming



the Xputer paradigm



not a "data flow machine" !

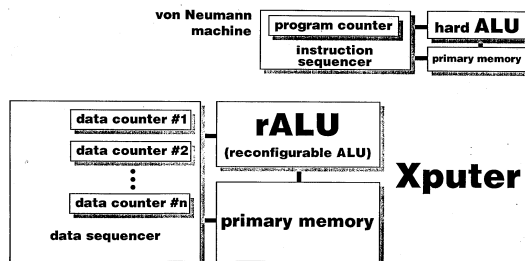
what's an Xputer ?

- ❑ a novel procedural machine paradigm
 - **data counter** instead of instruction counter
 - supports soft ALU (using FPLCs, FPICs etc.)
 - data-procedural execution mechanism
 - deterministic: traceable, debuggable

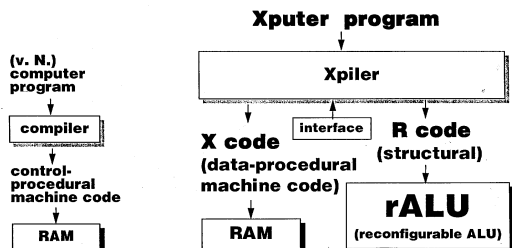
compared to Xputers the data flow machine is rather a Monte Carlo Machine !



data-procedural machine paradigm



data-procedural compilation



Xputer application areas

(high acceleration factors)

- ❑ Image preprocessing & other DSP
- ❑ efficient Neuronet Emulation
- ❑ uniform recurrency equation systems
- ❑ grid-based / multi grid computation problems
- ❑ other scientific computing
- ❑ aero-space,
- ❑ weather forecast / environmental modeling
- ❑ EDA
- ❑ other areas



acceleration factors vs. von Neumann

- Spaghetti / Sauerkraut type software:
 ≈ 1/2 order of magnitude
- Algorithms with regular data dependencies:
 ≈ 2 - 3 orders of magnitude
- why so fast ?
 - overhead pushed to compile time
 - some Xputer architectural features



achieved ...

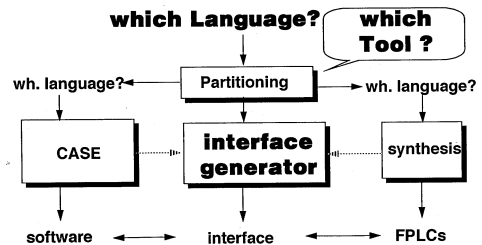
- third data-procedural Language designed
- experimental compiler implemented
- experimental program generator implemented
- third generation MoM Xputer being built
- promising experimental performance results
- best & 2nd best paper awards



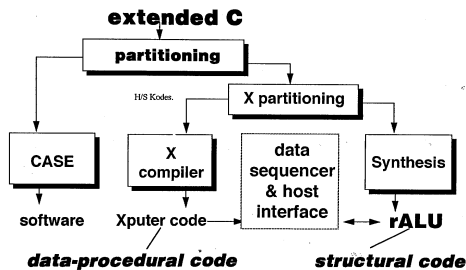
conclusions



why related to CASHE ?



Xputer-based CASHE



Xputer applications

as universal as von Neumann

- high performance:
 - universal accelerator co-processor platform
 - compiled accelerators within CASHE
 - stand alone for embedded systems
 - new ways to quick ASIC implementation
 - new ways to super computing



Xputerimpact

- data-procedural programming languages (new area...)
- new programming styles...
- close to programmers' virtual von Neumann ...
- but much more efficient
- in programming language research

- powerful tool to optimize hardw./ soft. trade-off
- improvements in portability & reusability (TM)
- CASHE: usable also by software-only type people



END