Real time Ray Tracing of Dynamic Scenes on an FPGA Chip

Jörg Schmittler, Sven Woop, Daniel Wagner,
Wolfgang J. Paul, Philipp Slusallek

Computer Graphics Group,
Saarland University, Germany

„Some argue that in the very long term, rendering may best be solved by some variant of ray tracing, in which huge numbers of rays sample the environment for the eye's view of each frame.

„Real-Time Rendering", 1st edition (page 391)

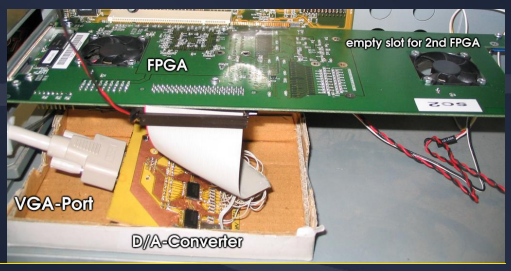
„Some argue that in the very long term, rendering may best be solved by some variant of ray tracing, in which huge numbers of rays sample the environment for the eye's view of each frame.

And there will also be colonies on Mars, underwater cities, and personal jet packs."

„Real-Time Rendering", 1st edition (page 391)

No Jet Packs, but …

- First graphics hardware for real time ray tracing
  - This presentation actually runs on it
- Full ray tracing pipeline
  - With reflections, transparencies, and shadows
  - Phong shading with bilinear-filtered textures
  - Fully supports dynamic scenes



Previous Work

Other Interactive ray tracing solutions:

- CPUs: Supercomputers, PC clusters [Wald'01]

→ It scales well in number of clients

- GPUs [Purcell'02] and streaming processors

→ Promising, but not (yet?) efficient

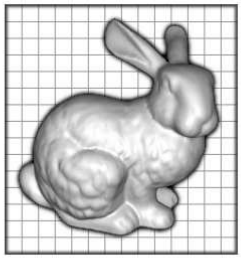
- Special purpose hardware [Green'91, Pfister'99]

→ First as accelerators or limited ray casters

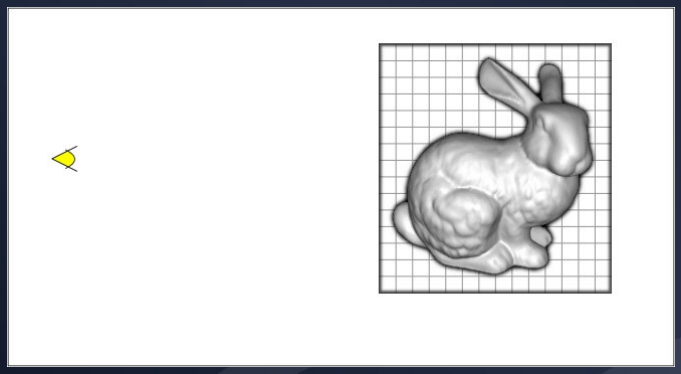
  - No success due to technology reasons
  - Ray tracing unit requires minimum size

But now the time is right for ray tracing in hardware!

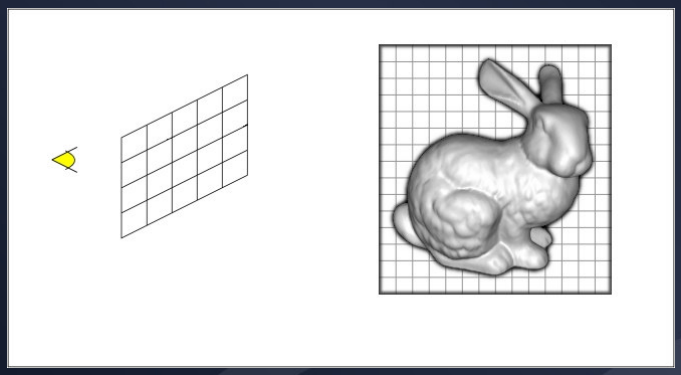
SaarCOR Architecture:
Ray Tracing of Static Scenes



SaarCOR Architecture:
Ray Tracing of Static Scenes

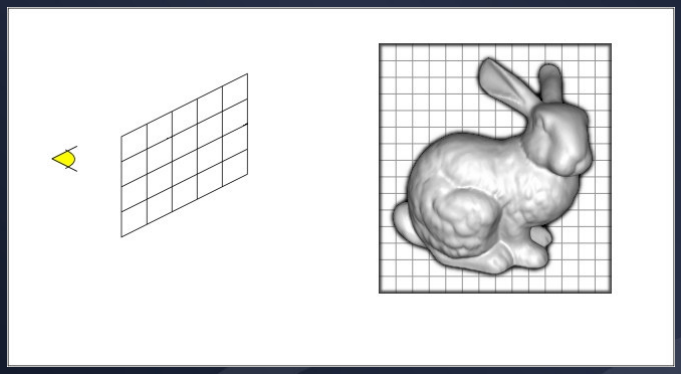


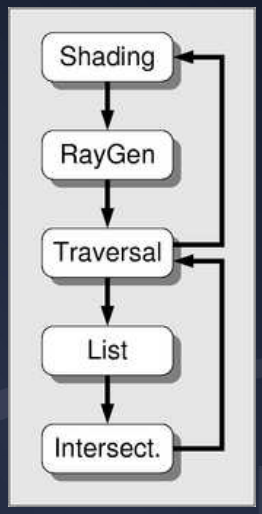
SaarCOR Architecture:
Ray Tracing of Static Scenes



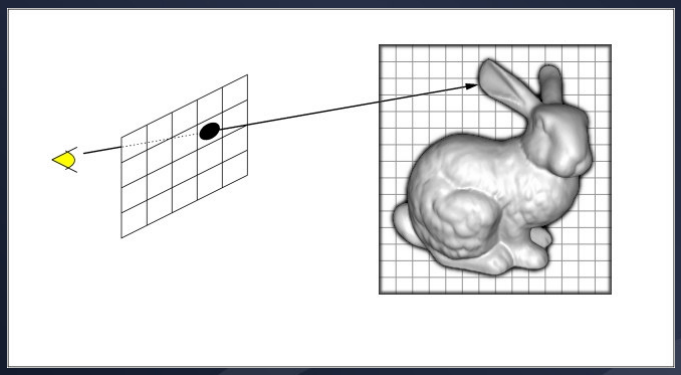
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

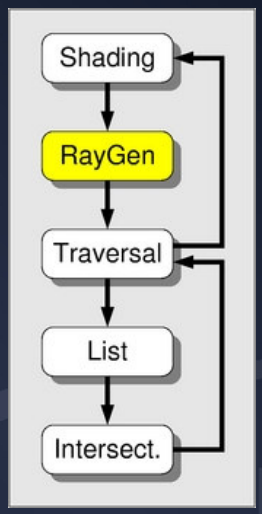
SaarCOR Architecture:
Ray Tracing of Static Scenes





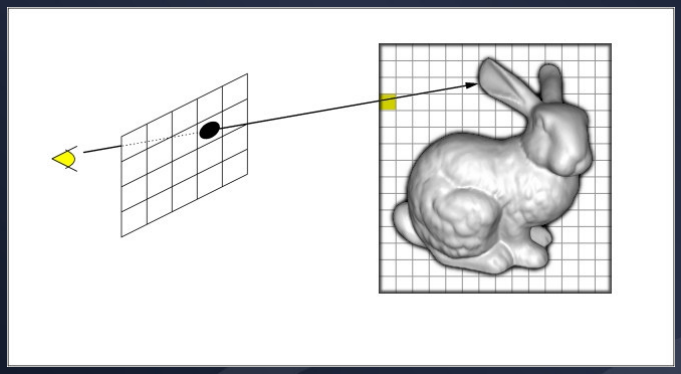
SaarCOR Architecture:
Ray Tracing of Static Scenes

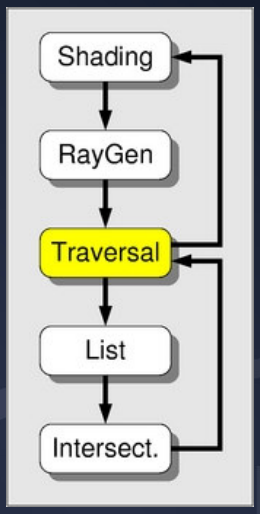




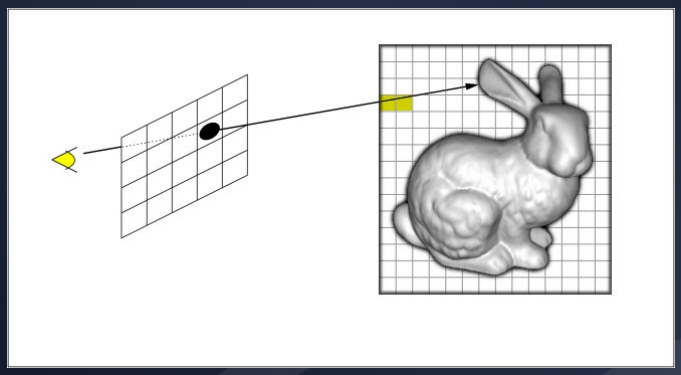
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

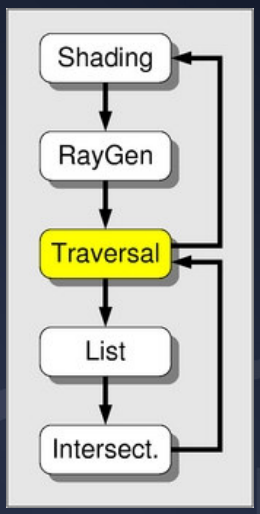
SaarCOR Architecture:
Ray Tracing of Static Scenes



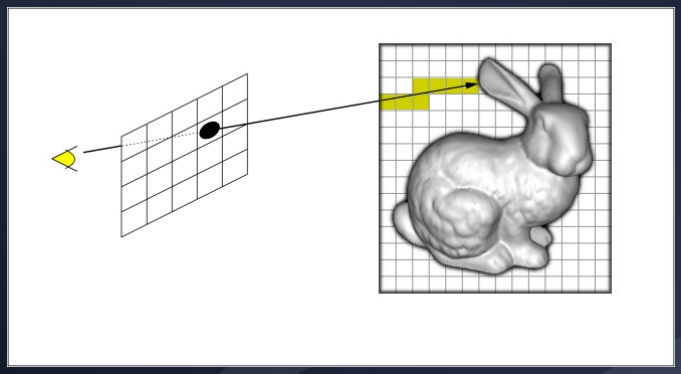


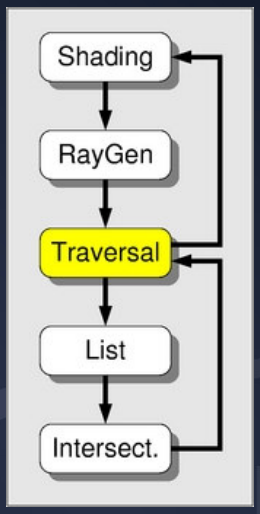
SaarCOR Architecture:
Ray Tracing of Static Scenes



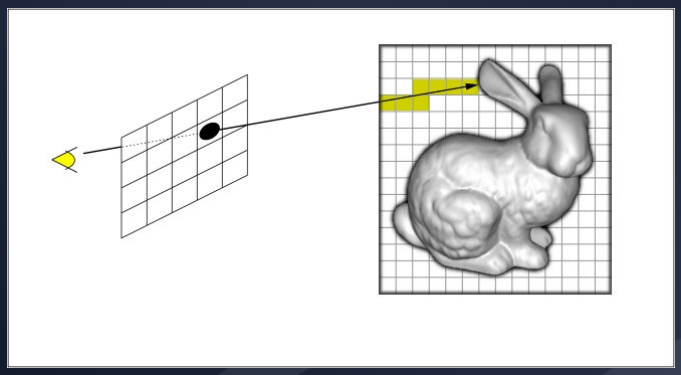


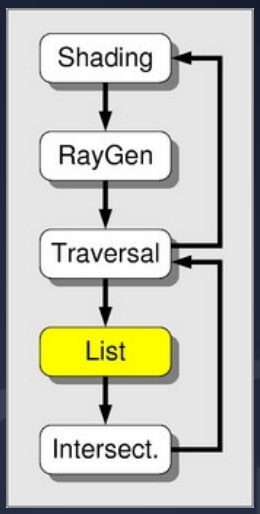
SaarCOR Architecture:
Ray Tracing of Static Scenes



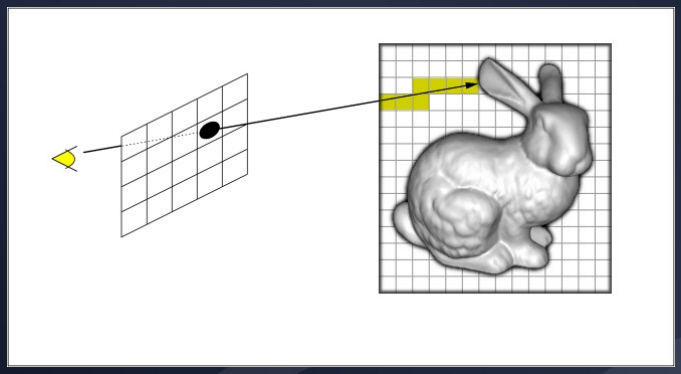


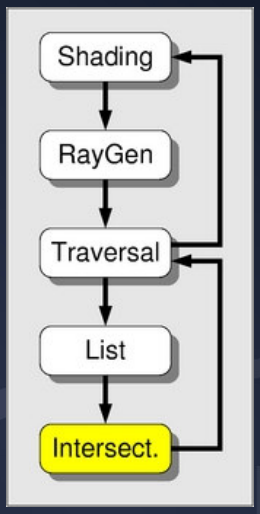
SaarCOR Architecture:
Ray Tracing of Static Scenes



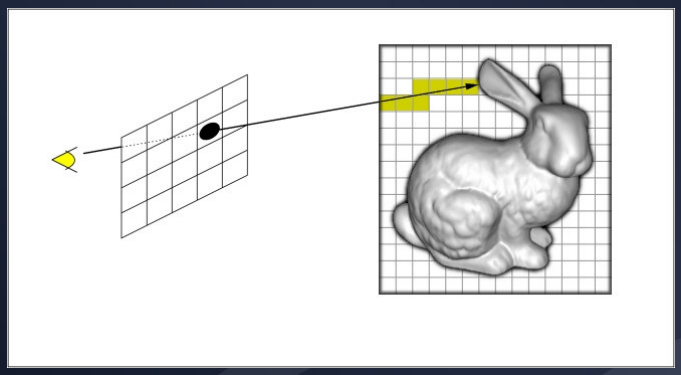


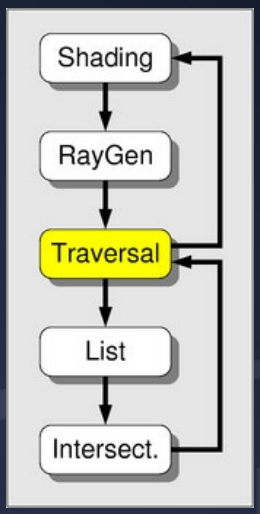
SaarCOR Architecture:
Ray Tracing of Static Scenes



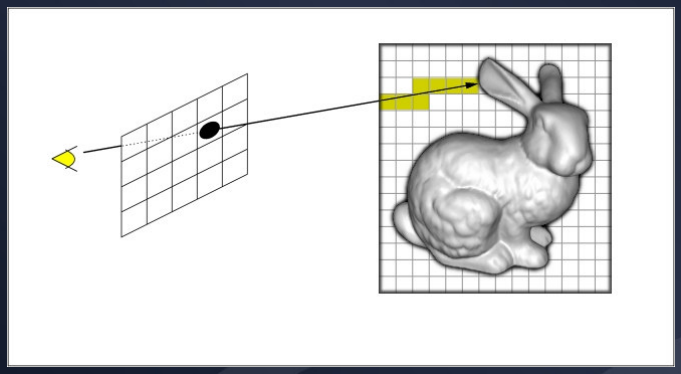


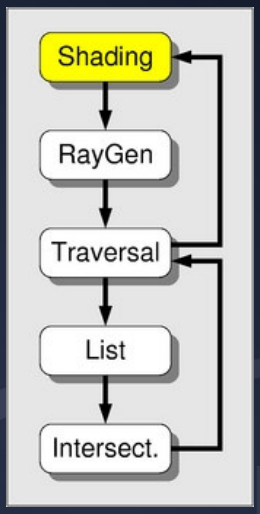
SaarCOR Architecture:
Ray Tracing of Static Scenes



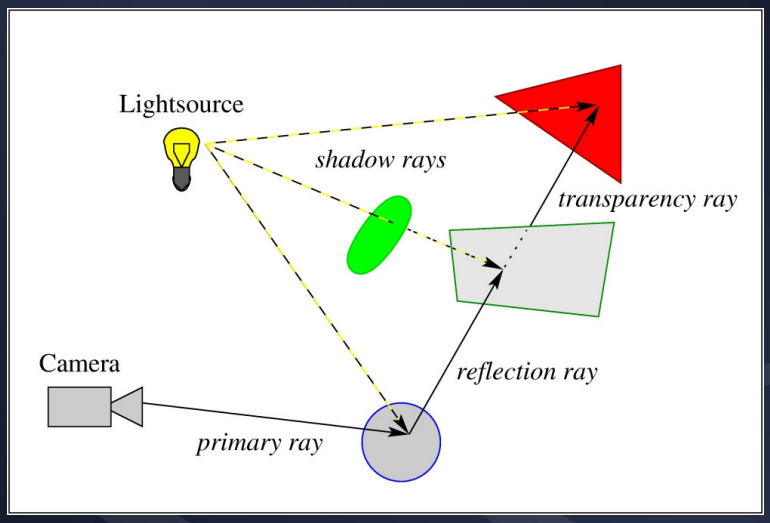


SaarCOR Architecture:
Ray Tracing of Static Scenes

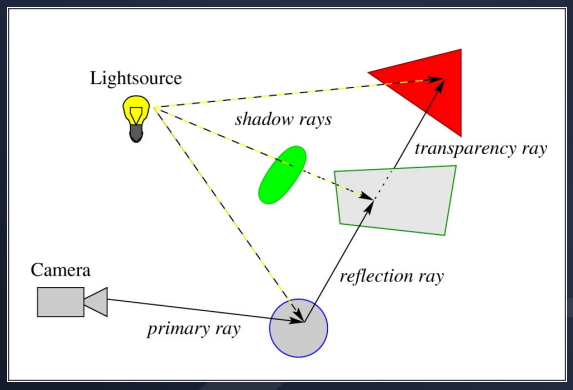




SaarCOR Architecture:
Recursive Ray Tracing

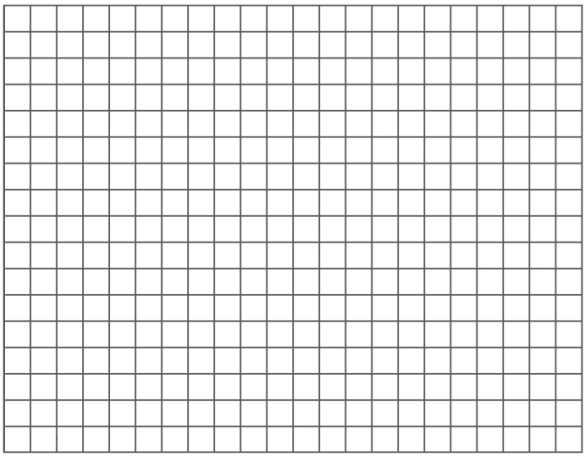


SaarCOR Architecture:
Recursive Ray Tracing

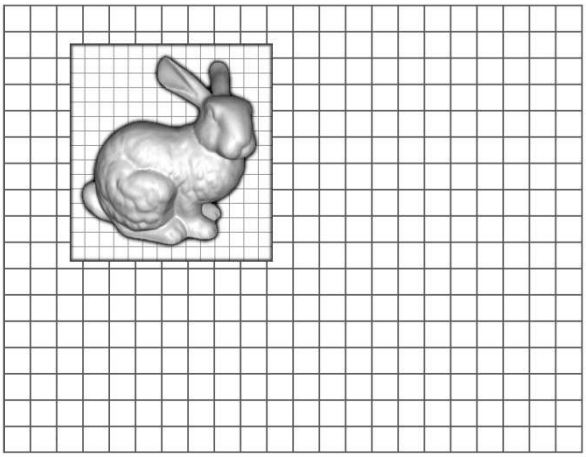


- Demand driven
- Global effects
- Fully automatic
- Per pixel operations
- Highly efficient
- Plug'n'play shading

SaarCOR Architecture:
Ray Tracing of Dynamic Scenes

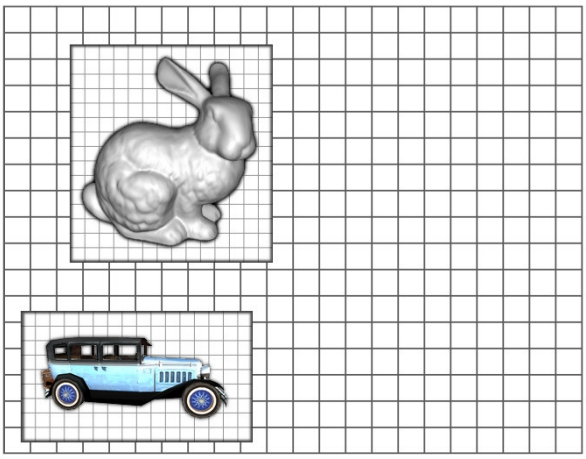


SaarCOR Architecture:
Ray Tracing of Dynamic Scenes



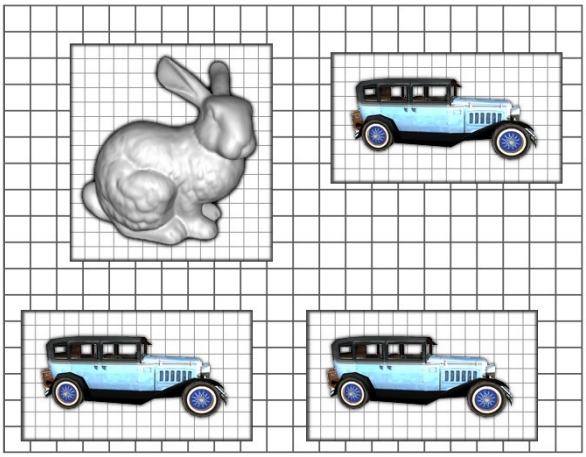
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Ray Tracing of Dynamic Scenes



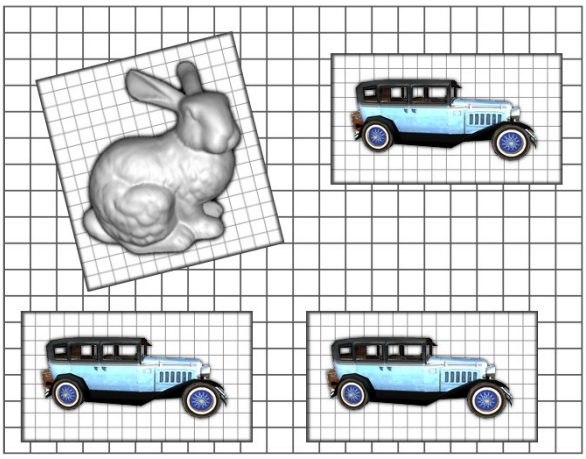
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Ray Tracing of Dynamic Scenes



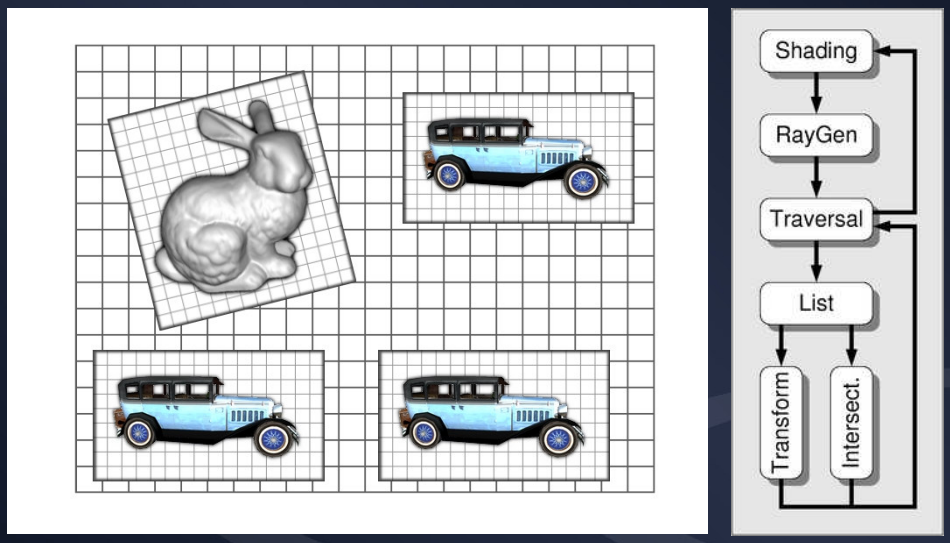
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Ray Tracing of Dynamic Scenes

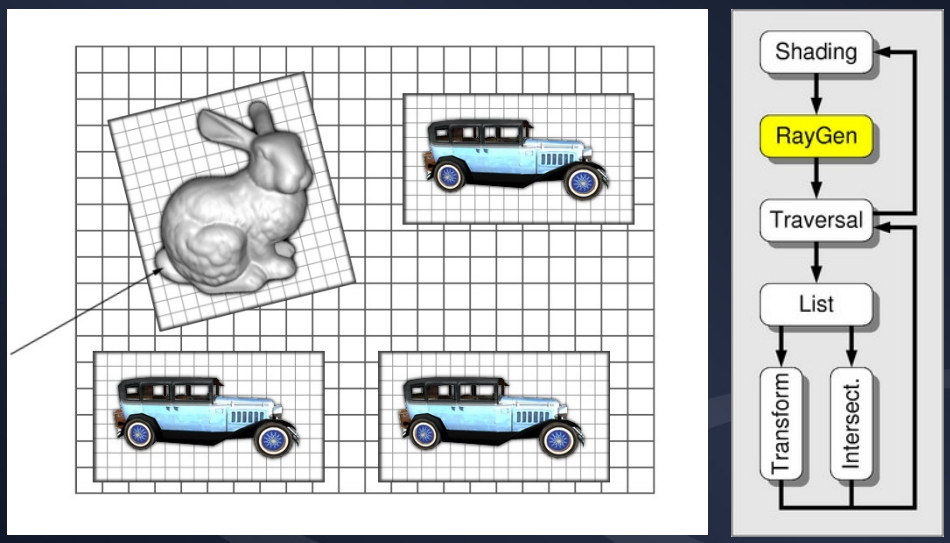


COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

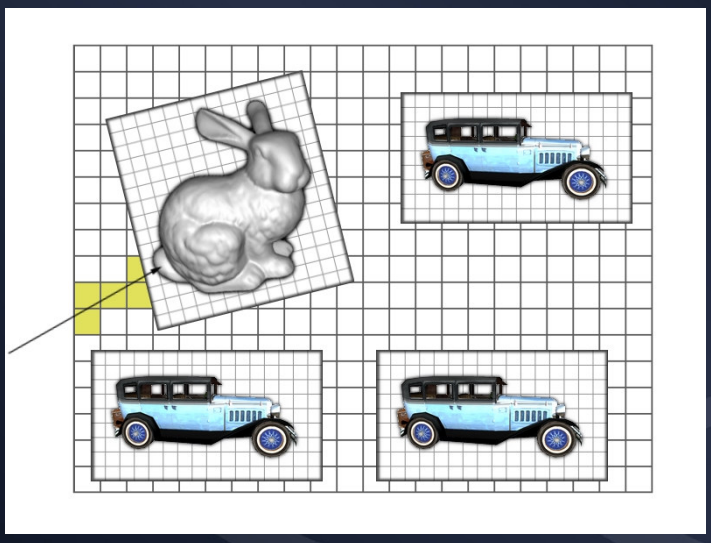
SaarCOR Architecture:
Ray Tracing of Dynamic Scenes

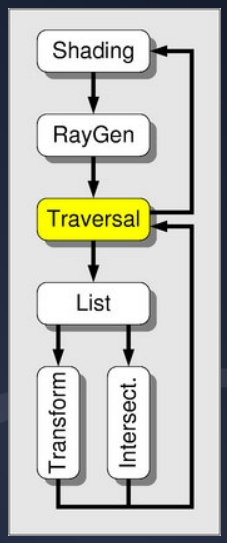


SaarCOR Architecture:
Ray Tracing of Dynamic Scenes

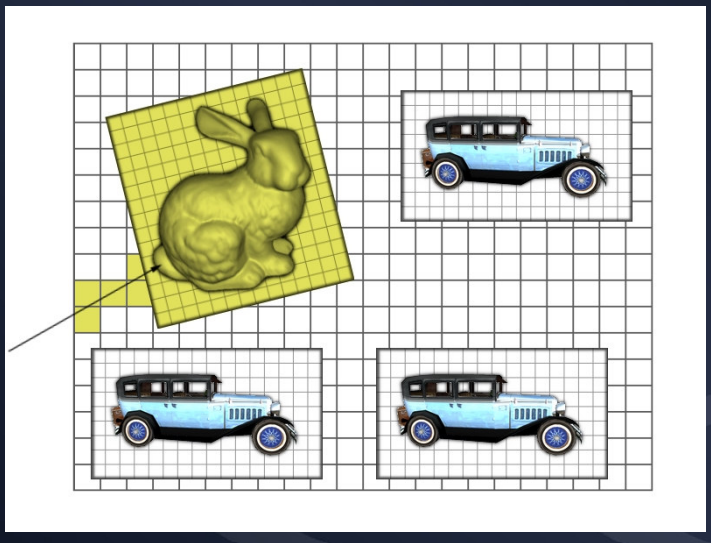


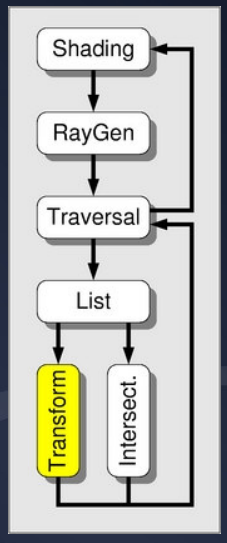
SaarCOR Architecture:
Ray Tracing of Dynamic Scenes





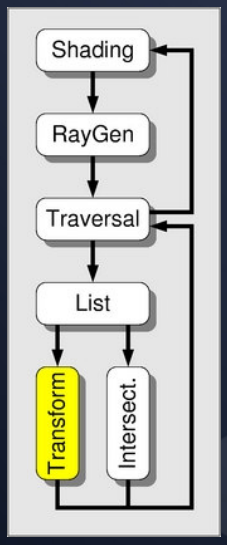
SaarCOR Architecture:
Ray Tracing of Dynamic Scenes



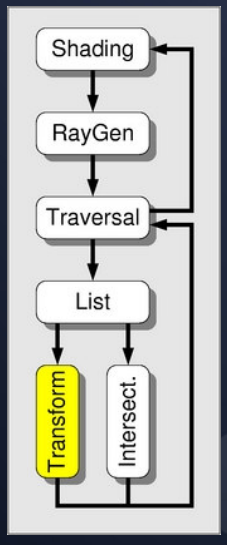


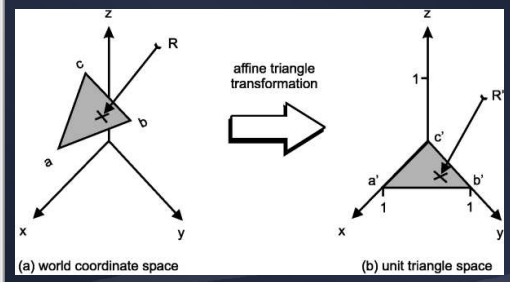
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Improving Hardware Efficiency



SaarCOR Architecture:
Improving Hardware Efficiency

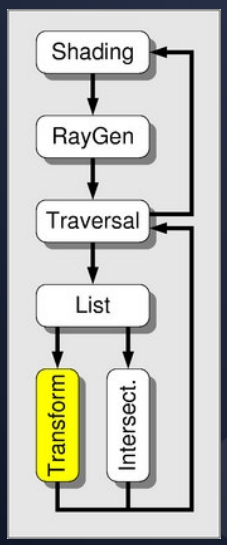


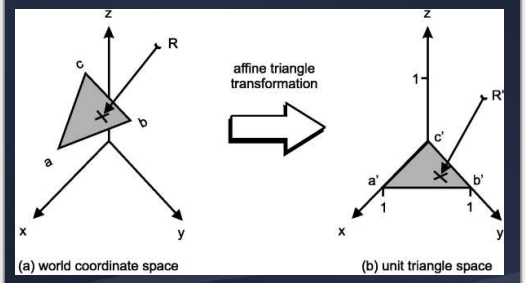


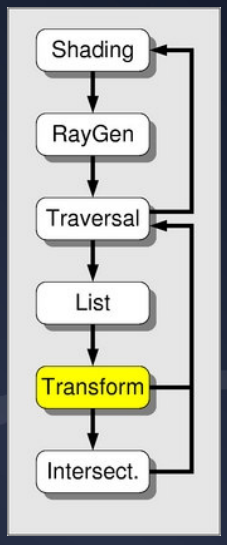
[Arenberg'88]

COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Improving Hardware Efficiency

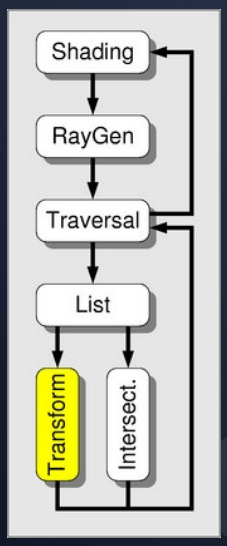




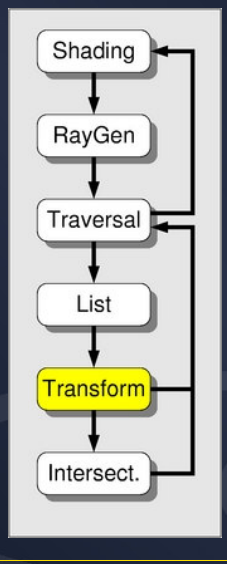


COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Improving Hardware Efficiency

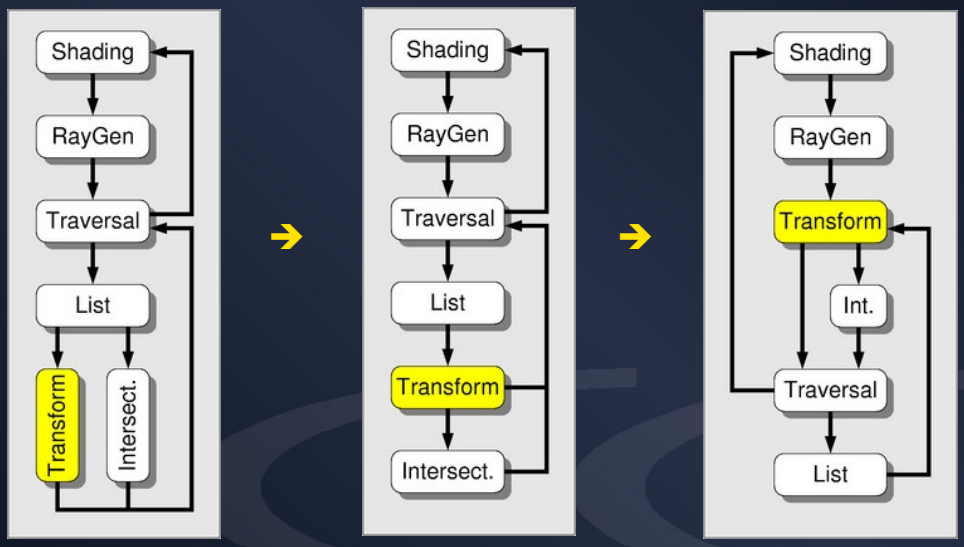


➔

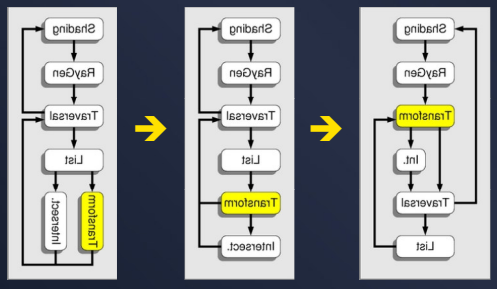


COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES

SaarCOR Architecture:
Improving Hardware Efficiency



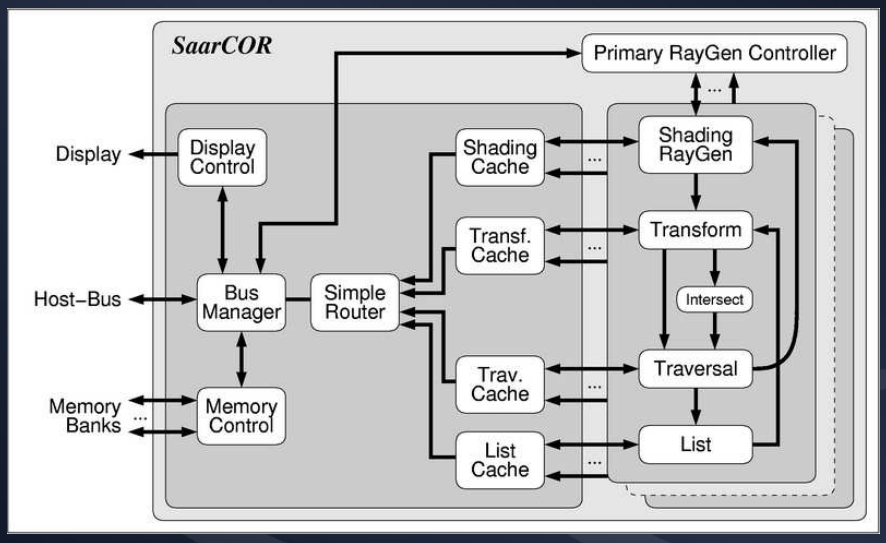
SaarCOR Architecture:
Improving Hardware Efficiency



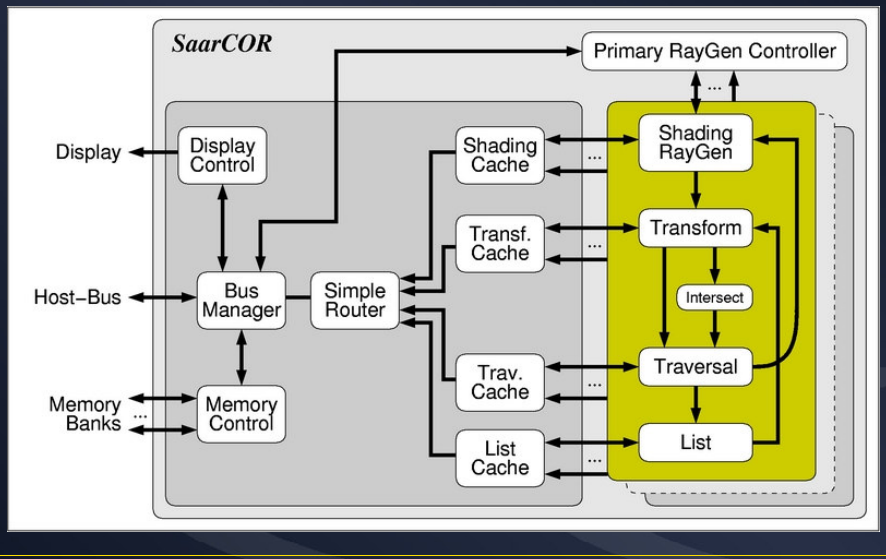
Optimized *dynamic* SaarCOR is 25% cheaper than simple *static* SaarCOR!

Optimizations allow for implementation on a single Xilinx Virtex-II-6000 FPGA chip!

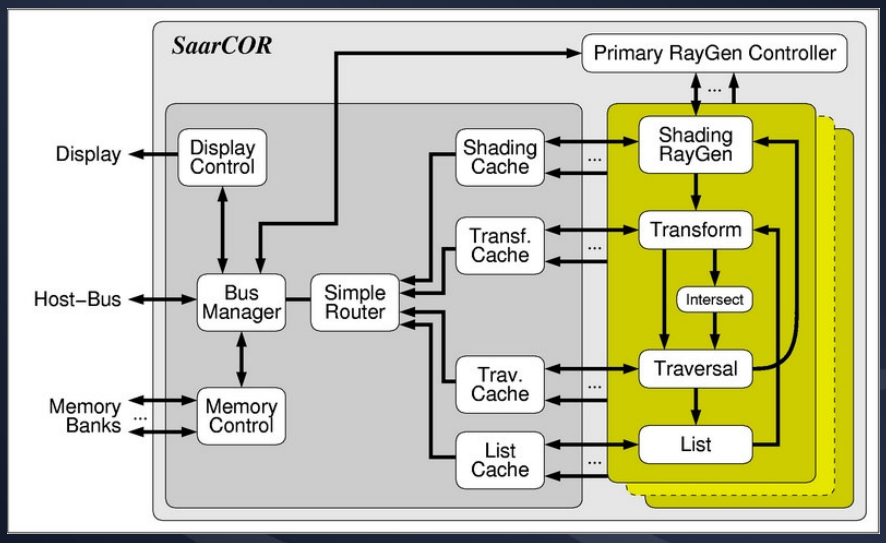
SaarCOR Architecture:
Parameterized Hardware Architecture



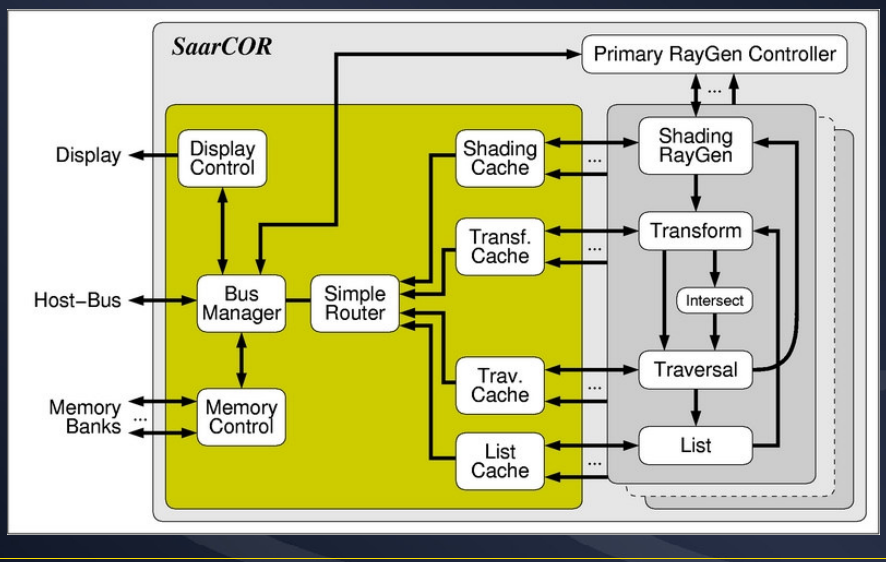
SaarCOR Architecture:
Parameterized Hardware Architecture



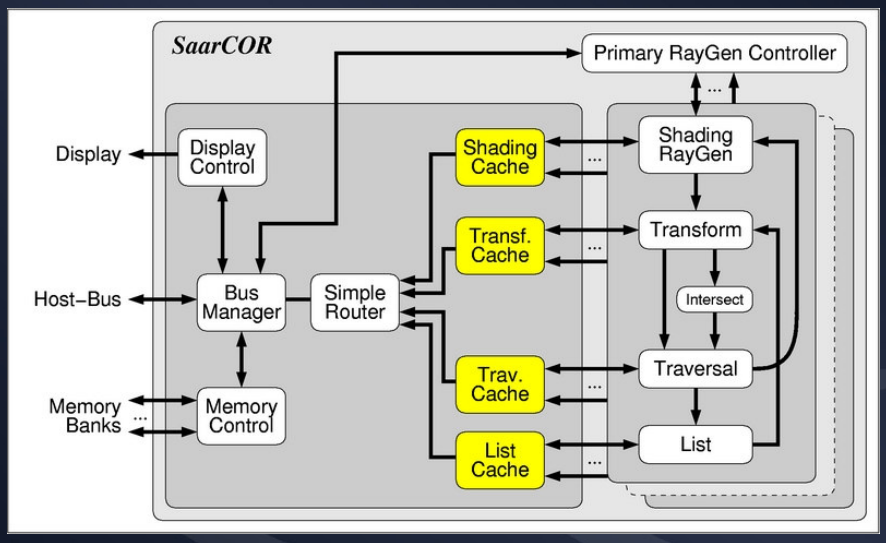
SaarCOR Architecture:
Parameterized Hardware Architecture



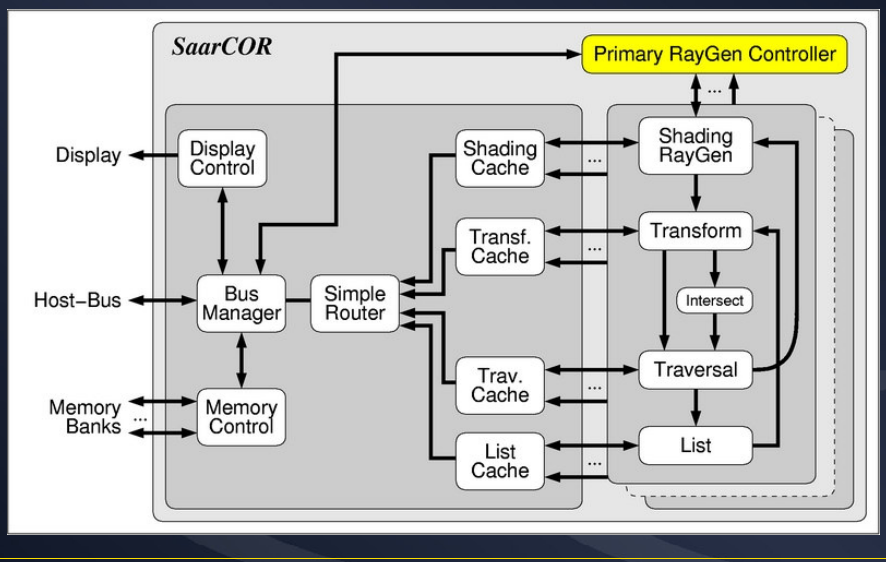
SaarCOR Architecture:
Parameterized Hardware Architecture



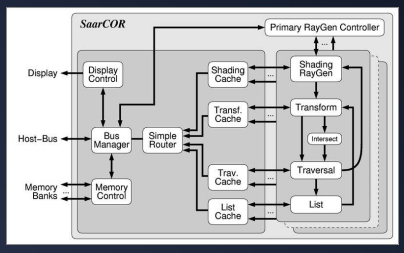
SaarCOR Architecture:
Parameterized Hardware Architecture



SaarCOR Architecture:
Parameterized Hardware Architecture



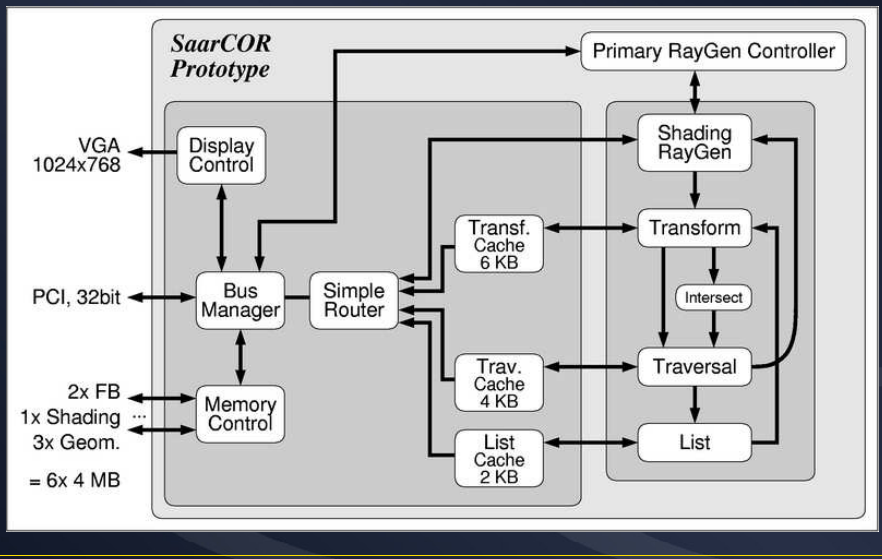
SaarCOR Architecture:
Parameterized Hardware Architecture



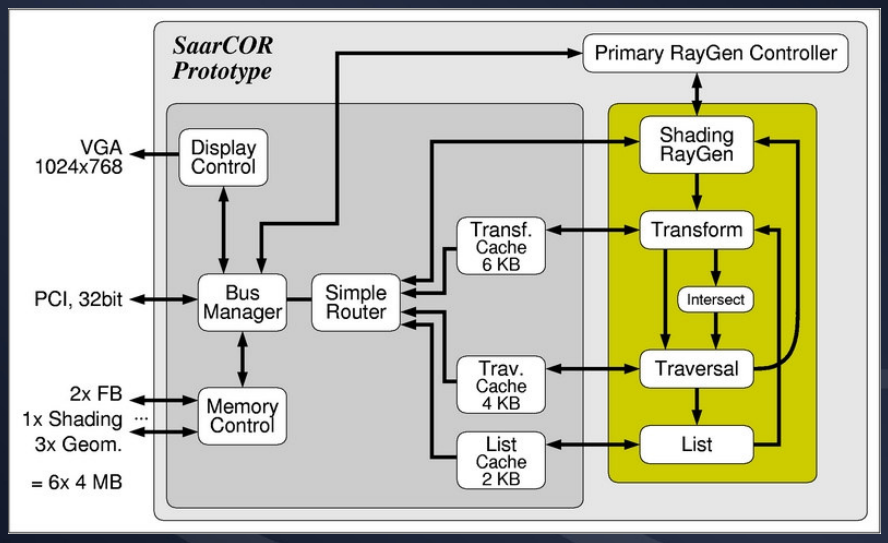
Features:

- Packets of rays to exploit coherence
- Multi-threading for latency hiding
- Scalability through
  - Multiple pipelines on chips
  - Multiple chips on a board
  - Multiple boards in a PC
- Application specific scaling inside a pipeline
- Low memory bandwidth requirements
- Fully automatic virtual memory management [Schmittler'03]

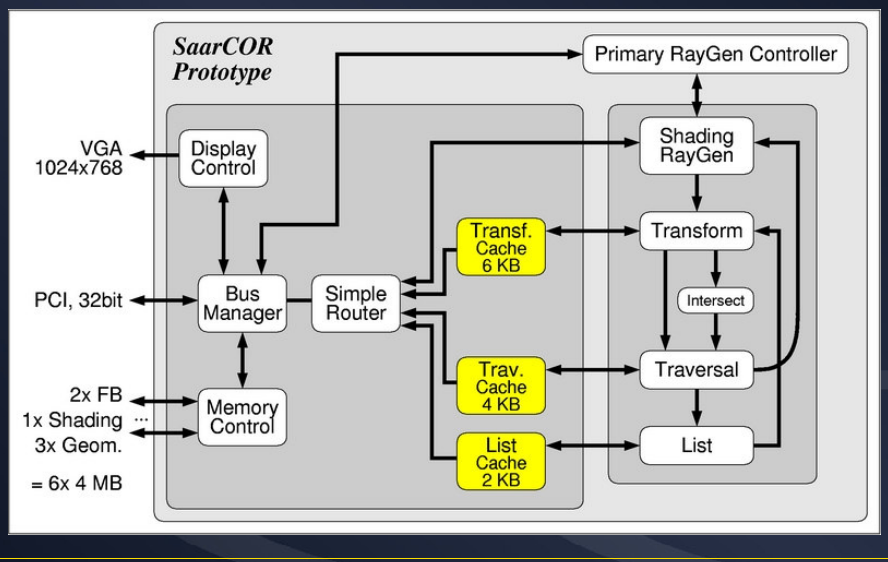
SaarCOR Architecture:
Prototype Architecture



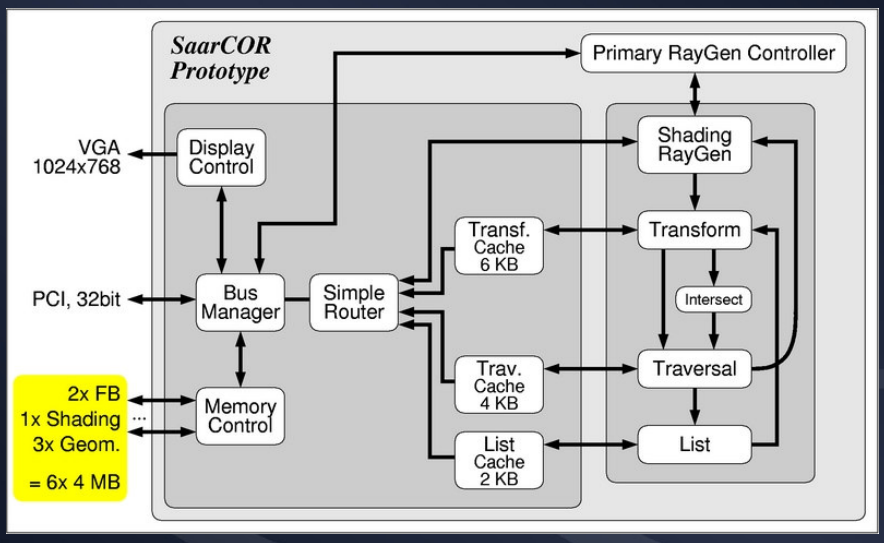
SaarCOR Architecture:
Prototype Architecture



SaarCOR Architecture:
Prototype Architecture



SaarCOR Architecture:
Prototype Architecture



SaarCOR Architecture:
Prototype Features

- Single pipe at only 90 MHz = performance of Intel P4 @ 8-12 GHz
- Only 4 GFLOPs (GeForce FX5900 has 50 times more)
- Low external bandwidth requirements
  - Mostly < 300 MB/s (GF has 100 times more available)
  - Even for complex scenes with un-cached shading
- Already up to 12 million rays per second resulting in 20-60 fps!

Conclusion

- Real time ray tracing in hardware is possible
- Highly scalable architecture
- Allows for advanced rendering applications
  - Global illumination [Benthin'03]
  - Simplified content creation [Schmittler'04]
  - Massive models [Dietrich'04]


Future Work

- Programmable shading
- Better support for fully dynamic scenes
- OpenRT-API [Dietrich'03]
- ASIC prototype



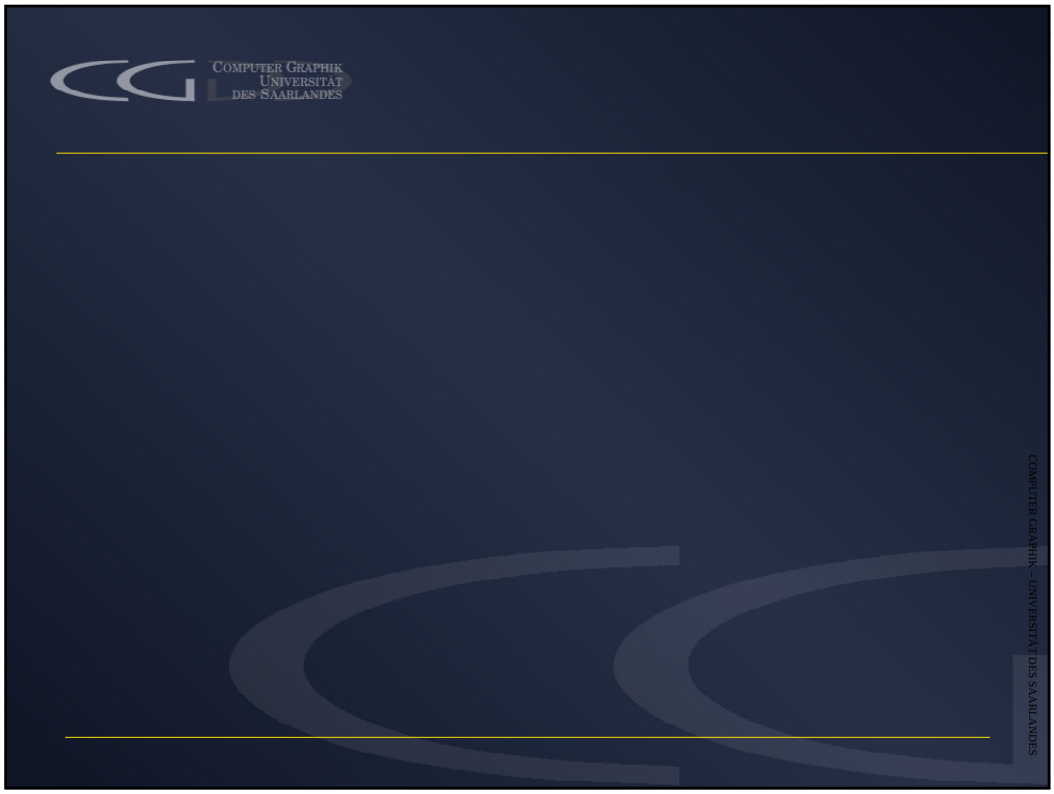
SaarCOR Architecture:
Prototype Demonstration

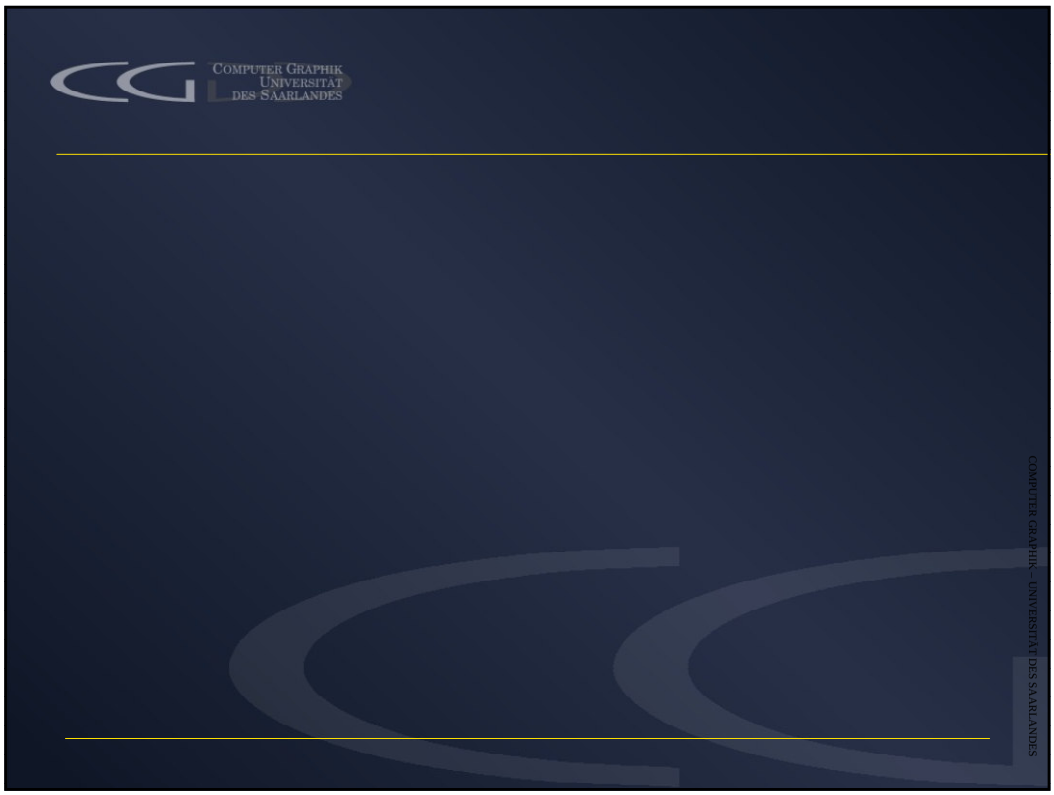
DEMO

Thanks for your attention!

Questions?

graphics.cs.uni-sb.de
OpenRT.de
SaarCOR.de


COMPUTER GRAPHIK
UNIVERSITÄT
DES SAARLANDES
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES


COMPUTER GRAPHIK
UNIVERSITÄT
DES SAARLANDES
COMPUTER GRAPHIK – UNIVERSITÄT DES SAARLANDES