

Concurrent Systems

Nebenläufige Systeme

XIV. Pickings

Wolfgang Schröder-Preikschat

February 3, 2016



Agenda

Recapitulation
Concurrent Systems

Perspectives
Parallel Systems
Computing Equipment
Further Education

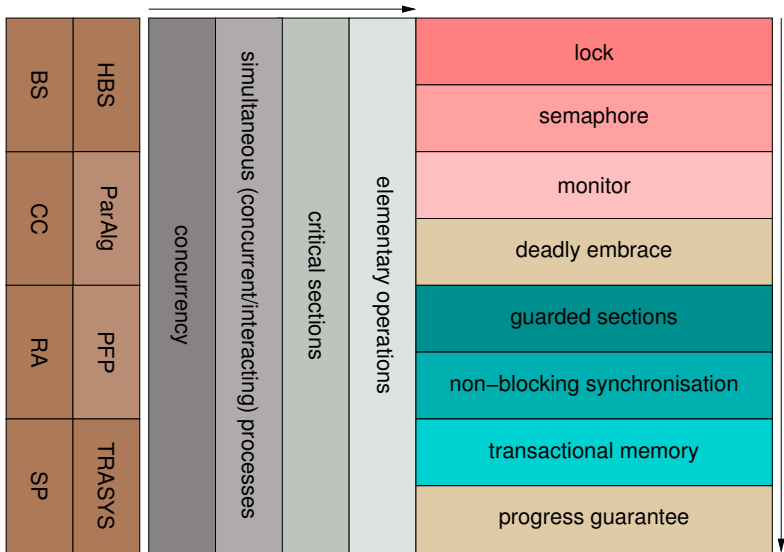


Recapitulation
Concurrent Systems

Perspectives
Parallel Systems
Computing Equipment
Further Education



Content of Teaching and Cross-References



Outline

Recapitulation

Concurrent Systems

Perspectives

Parallel Systems

Computing Equipment

Further Education



- **composability** and **configurability**
 - application-oriented (varying, type-safe) system software
- **specialisation**
 - dedicated operating systems: integrated, adaptive, parallel



- **composability** and **configurability**
 - application-oriented (varying, type-safe) system software
- **specialisation**
 - dedicated operating systems: integrated, adaptive, parallel
- **reliability**
 - gentle fault and intrusion tolerance
- **thriftiness**
 - resource-aware operation of computing systems
- **timeliness**
 - migration paths between time- and event-triggered real-time systems



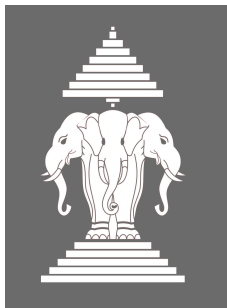
- **composability** and **configurability**
 - application-oriented (varying, type-safe) system software
- **specialisation**
 - dedicated operating systems: integrated, adaptive, parallel
- **reliability**
 - gentle fault and intrusion tolerance
- **thriftiness**
 - resource-aware operation of computing systems
- **timeliness**
 - migration paths between time- and event-triggered real-time systems
- **concurrency**
 - coordination of cooperation and competition between processes



- **composability** and **configurability**
 - application-oriented (varying, type-safe) system software
 - **specialisation**
 - dedicated operating systems: integrated, adaptive, parallel
 - **reliability**
 - gentle fault and intrusion tolerance
 - **thriftiness**
 - resource-aware operation of computing systems
 - **timeliness**
 - migration paths between time- and event-triggered real-time systems
 - **concurrency**
 - coordination of cooperation and competition between processes
- ↳ “concurrent systems” is more or less **cross-cutting** thereto. . .

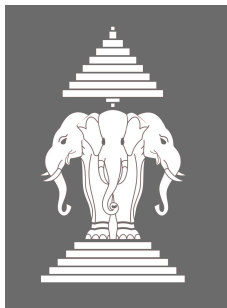


Latency Awareness in Operating Systems



¹<http://univis.uni-erlangen.de> → Research projects → LAOS

- **latency prevention**
 - lock- and wait-free synchronisation
 - integrated generator-based approach
- **latency avoidance**
 - interference protection
 - race-conflict containment
- **latency hiding**
 - operating-system server cores
 - asynchronous remote system operation



¹<http://univis.uni-erlangen.de> → Research projects → LAOS

■ latency prevention

- lock- and wait-free synchronisation
- integrated generator-based approach

■ latency avoidance

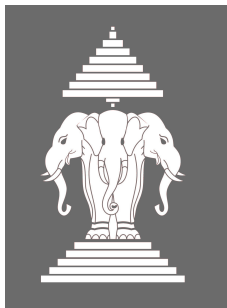
- interference protection
- race-conflict containment

■ latency hiding

- operating-system server cores
- asynchronous remote system operation

■ experiments with different **operating-system architectures**

- process-/event-based and hardware-centric operating-system kernels
- LAKE, Sloth



¹<http://univis.uni-erlangen.de> → Research projects → LAOS

■ latency prevention

- lock- and wait-free synchronisation
- integrated generator-based approach

■ latency avoidance

- interference protection
- race-conflict containment

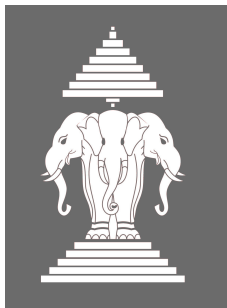
■ latency hiding

- operating-system server cores
- asynchronous remote system operation

■ experiments with different **operating-system architectures**

- process-/event-based and hardware-centric operating-system kernels
- LAKE, Sloth

■ DFG: 2 doctoral researchers, 2 student assistants

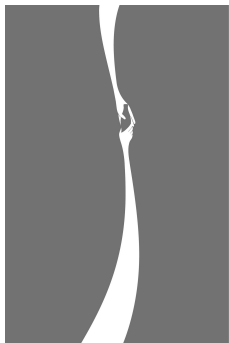


¹<http://univis.uni-erlangen.de> → Research projects → LAOS



²<http://univis.uni-erlangen.de> → Research projects → COKE

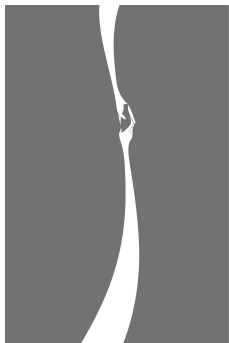
- **event-based minimal kernel**
 - cache-aware main-memory footprint
 - hyper-threading of latent actions



²<http://univis.uni-erlangen.de> → Research projects → COKE

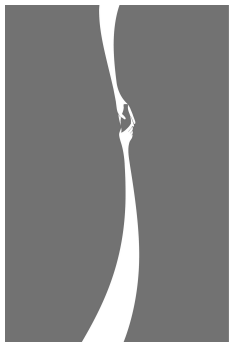


- **event-based minimal kernel**
 - cache-aware main-memory footprint
 - hyper-threading of latent actions
- featherweight **agreement protocols**
 - overall kernel-level synchronisation
 - familie of consistency kernels



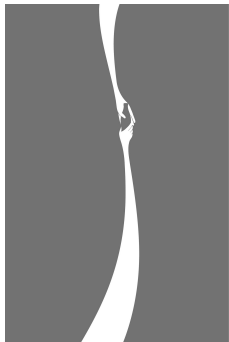
²<http://univis.uni-erlangen.de> → Research projects → COKE

- **event-based minimal kernel**
 - cache-aware main-memory footprint
 - hyper-threading of latent actions
- featherweight **agreement protocols**
 - overall kernel-level synchronisation
 - familie of consistency kernels
- **problem-oriented consistency**
 - sequential, entry, release consistency
 - functional hierarchy of consistency domains
 - memory domains for NUMA architectures



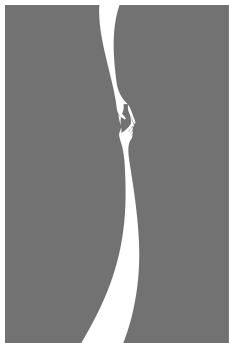
²<http://univis.uni-erlangen.de> → Research projects → COKE

- **event-based minimal kernel**
 - cache-aware main-memory footprint
 - hyper-threading of latent actions
- featherweight **agreement protocols**
 - overall kernel-level synchronisation
 - familie of consistency kernels
- **problem-oriented consistency**
 - sequential, entry, release consistency
 - functional hierarchy of consistency domains
 - memory domains for NUMA architectures
- implementation as to different **processor architectures**
 - partial or total, resp. {in,}coherent shared memory



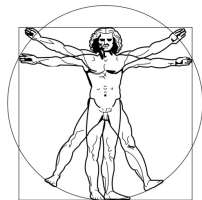
²<http://univis.uni-erlangen.de> → Research projects → COKE

- **event-based minimal kernel**
 - cache-aware main-memory footprint
 - hyper-threading of latent actions
- featherweight **agreement protocols**
 - overall kernel-level synchronisation
 - familie of consistency kernels
- **problem-oriented consistency**
 - sequential, entry, release consistency
 - functional hierarchy of consistency domains
 - memory domains for NUMA architectures
- implementation as to different **processor architectures**
 - partial or total, resp. {in,}coherent shared memory
- DFG: 2 doctoral researchers (1 FAU, 1 BTU)



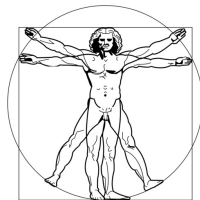
²<http://univis.uni-erlangen.de> → Research projects → COKE

Heterogeneous Resource-Aware Multi-Processing



³<http://univis.uni-erlangen.de> → Research projects → RAMP

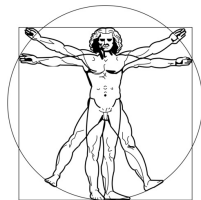
- GPU-centric **resource management**
 - timely predictable run-time system
 - run-to-completion kernel
 - prioritisation and isolation of GPU tasks
 - scheduling according to execution costs
 - trade-off handling as to throughput and response time



³<http://univis.uni-erlangen.de> → Research projects → RAMP

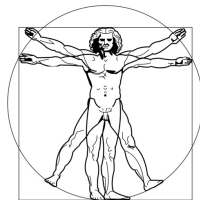
Heterogeneous Resource-Aware Multi-Processing

- GPU-centric **resource management**
 - timely predictable run-time system
 - run-to-completion kernel
 - prioritisation and isolation of GPU tasks
 - scheduling according to execution costs
 - trade-off handling as to throughput and response time
- RAM-centric **run-time executive** for heterogeneous processors
 - application-specific and problem-oriented memory management
 - run-time adaptation and relocation of dynamic data structures



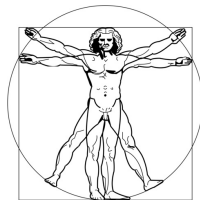
³<http://univis.uni-erlangen.de> → Research projects → RAMP

- GPU-centric **resource management**
 - timely predictable run-time system
 - run-to-completion kernel
 - prioritisation and isolation of GPU tasks
 - scheduling according to execution costs
 - trade-off handling as to throughput and response time
- RAM-centric **run-time executive** for heterogeneous processors
 - application-specific and problem-oriented memory management
 - run-time adaptation and relocation of dynamic data structures
- **tailor-made system software** for heterogeneous image systems
 - support of an incremental improvement of visual quality
 - patterns for adaptive detail adaptation of geometry or textures



³<http://univis.uni-erlangen.de> → Research projects → RAMP

- GPU-centric **resource management**
 - timely predictable run-time system
 - run-to-completion kernel
 - prioritisation and isolation of GPU tasks
 - scheduling according to execution costs
 - trade-off handling as to throughput and response time
- RAM-centric **run-time executive** for heterogeneous processors
 - application-specific and problem-oriented memory management
 - run-time adaptation and relocation of dynamic data structures
- **tailor-made system software** for heterogeneous image systems
 - support of an incremental improvement of visual quality
 - patterns for adaptive detail adaptation of geometry or textures
- DFG: 1 doctoral researcher, 1 student assistant



³<http://univis.uni-erlangen.de> → Research projects → RAMP

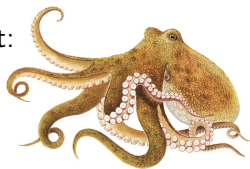
Run-Time Support System for Invasive Computing



⁴<http://univis.uni-erlangen.de> → Research projects → iRTSS

Octo

- borrowed from the designation of a creature that:
 - i is highly parallel in its actions and
 - ii excellently can adapt oneself to its environment
- the kraken (species *Octopoda*)
 - can operate in parallel by virtue of its eight tentacle
 - is able to do customisation through camouflage and deimatic displays and
 - comes with a highly developed nervous system
 - in order to attune to dynamic ambient conditions and effects



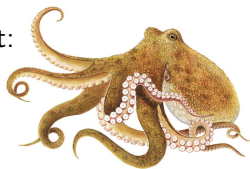
POS

- abbrev. for *parallel operating system*
 - an operating system that not only supports parallel processes
 - but that also functions **inherently parallel** thereby

⁴<http://univis.uni-erlangen.de> → Research projects → iRTSS

Octo

- borrowed from the designation of a creature that:
 - i is highly parallel in its actions and
 - ii excellently can adapt oneself to its environment
- the kraken (species *Octopoda*)
 - can operate in parallel by virtue of its eight tentacle
 - is able to do customisation through camouflage and deimatic displays and
 - comes with a highly developed nervous system
 - in order to attune to dynamic ambient conditions and effects



POS

- abbrev. for *parallel operating system*
 - an operating system that not only supports parallel processes
 - but that also functions **inherently parallel** thereby
- DFG: 2.5 doctoral researchers, 1 research/3 student assistants

⁴<http://univis.uni-erlangen.de> → Research projects → iRTSS

PAX

⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Power-Aware Critical Sections

- scalable synchronisation on the basis of **agile critical sections**
 - **infrastructure** ■ load-dependent and self-organised change of protection against race conditions
 - **linguistic support** ■ preparation, characterisation, and capturing of declared critical sections

PAX

⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Power-Aware Critical Sections

- scalable synchronisation on the basis of **agile critical sections infrastructure**
 - load-dependent and self-organised change of protection against race conditions
- **linguistic support**
 - preparation, characterisation, and capturing of declared critical sections
- automated extraction of critical sections
 - notation language for critical sections
 - program analysis and LLVM integration/adaptation



⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Power-Aware Critical Sections

- scalable synchronisation on the basis of **agile critical sections infrastructure**
 - load-dependent and self-organised change of protection against race conditions
- **linguistic support**
 - preparation, characterisation, and capturing of declared critical sections
- automated extraction of critical sections
 - notation language for critical sections
 - program analysis and LLVM integration/adaptation
- power-aware system programming
 - mutual exclusion, guarded sections, transactions
 - dynamic dispatch of synchronisation protocols or critical sections, resp.

The logo for PAX consists of the letters 'PAX' in a bold, black, sans-serif font. The letter 'X' is replaced by a large, green, stylized cross symbol.

⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Power-Aware Critical Sections

- scalable synchronisation on the basis of **agile critical sections infrastructure**
 - load-dependent and self-organised change of protection against race conditions
- **linguistic support**
 - preparation, characterisation, and capturing of declared critical sections
- automated extraction of critical sections
 - notation language for critical sections
 - program analysis and LLVM integration/adaptation
- power-aware system programming
 - mutual exclusion, guarded sections, transactions
 - dynamic dispatch of synchronisation protocols or critical sections, resp.
- tamper-proof power-consumption measuring
 - instruction survey and statistics based on real and virtual machines
 - energy-consumption prediction or estimation, resp.

The logo for PAX consists of the letters 'PAX' in a bold, black, sans-serif font. The letter 'X' is replaced by a large, green, stylized 'X' shape.

⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Power-Aware Critical Sections

- scalable synchronisation on the basis of **agile critical sections infrastructure**
 - load-dependent and self-organised change of protection against race conditions
- **linguistic support**
 - preparation, characterisation, and capturing of declared critical sections
- automated extraction of critical sections
 - notation language for critical sections
 - program analysis and LLVM integration/adaptation
- power-aware system programming
 - mutual exclusion, guarded sections, transactions
 - dynamic dispatch of synchronisation protocols or critical sections, resp.
- tamper-proof power-consumption measuring
 - instruction survey and statistics based on real and virtual machines
 - energy-consumption prediction or estimation, resp.
- DFG: 2 doctoral researchers, 2 student assistants

The logo for PAX consists of the letters 'PAX' in a bold, black, sans-serif font. The letter 'X' is replaced by a large, green, stylized 'X' shape.

⁵<http://univis.uni-erlangen.de> → Research projects → PAX

Multi/Many-Core Processor Pool

fau_i4*	clock	cores per domain		domain		
		physical	logical	NUMA	tile	
8e 8f	2.9 GHz	8	16	2	–	Xeon
9big01	2.5 GHz	6	–	8	–	Opteron
9big02	2.2 GHz	10	20	4	–	Xeon
9phi01	1.2 GHz	6	12	2	–	Xeon
	1.1 GHz	57	228	2	–	Xeon Phi
scc	1.5 GHz	4	2	1	–	Xeon
	800 MHz	2	–	–	24	Pentium
InvasIC	3.5 GHz	8	16	2	–	Xeon
	25 MHz	4	–	6		LEON/SPARC



Multi/Many-Core Processor Pool

fau4*	clock	cores per domain		domain		
		physical	logical	NUMA	tile	
8e 8f	2.9 GHz	8	16	2	–	Xeon
9big01	2.5 GHz	6	–	8	–	Opteron
9big02	2.2 GHz	10	20	4	–	Xeon
9phi01	1.2 GHz	6	12	2	–	Xeon
	1.1 GHz	57	228	2	–	Xeon Phi
scc	1.5 GHz	4	2	1	–	Xeon
	800 MHz	2	–	–	24	Pentium
InvasIC	3.5 GHz	8	16	2	–	Xeon
	25 MHz	4	–	6		LEON/SPARC

- budgeted acquisition: further n -core systems, transactional memory

OctoPOS ■ $n \geq 64$, in 2015

PAX ■ $n \geq 16$, in 2016, plus several multi-core micro-controllers



Bachelor, Master, or Doctoral Thesis

