

Genode as general-purpose OS progress report and demonstration



Norman Feske
<norman.feske@genode-labs.com>



Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014



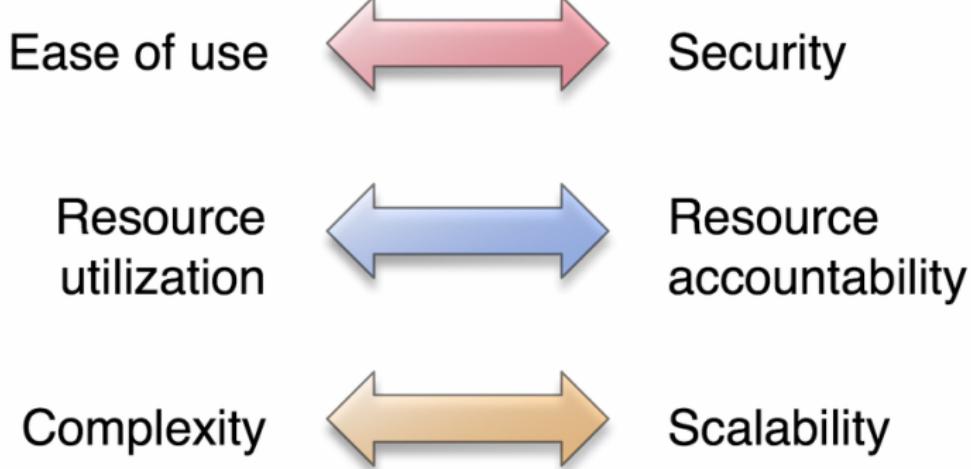
Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Myths





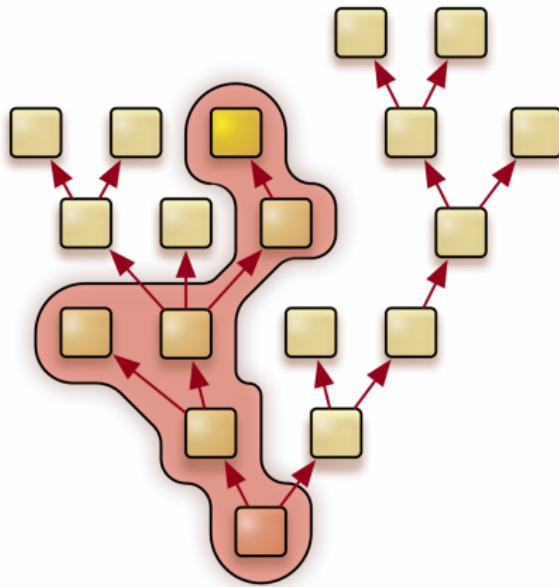
Key technologies

- Microkernels
- Decomponentization, kernelization
- Capability-based security
- Virtualization





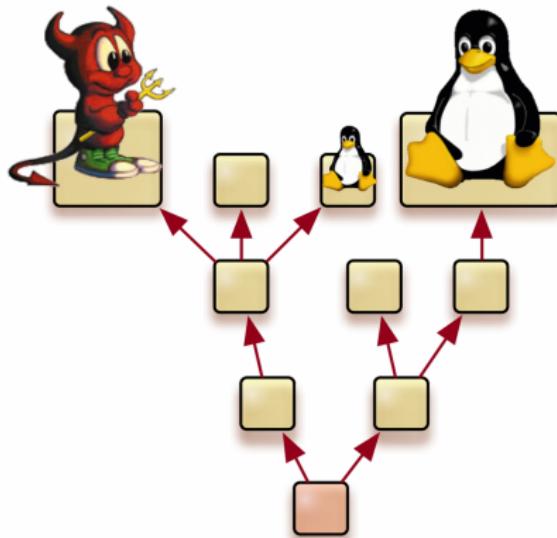
Genode architecture



→ Application-specific TCB



Combined with virtualization



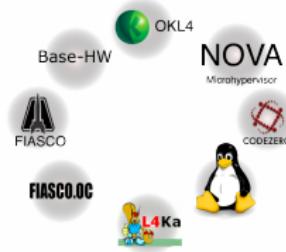


How to get there? Found a company!

- Genode Labs, founded in May 2008, self-funded
- Systems research and development
- Idea: *Start small, build sustainable business, grow organically*
- Team of 8 people
- Small yet diverse customer base
- Main source of income is contracting work



Components



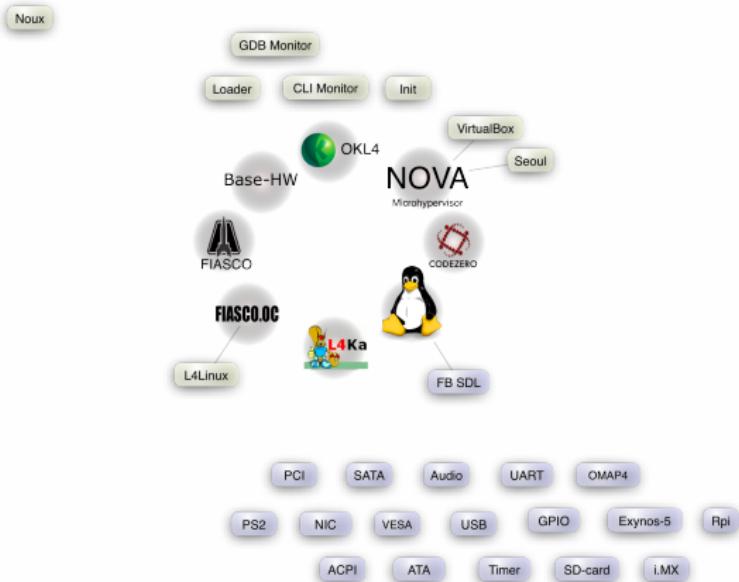


Components



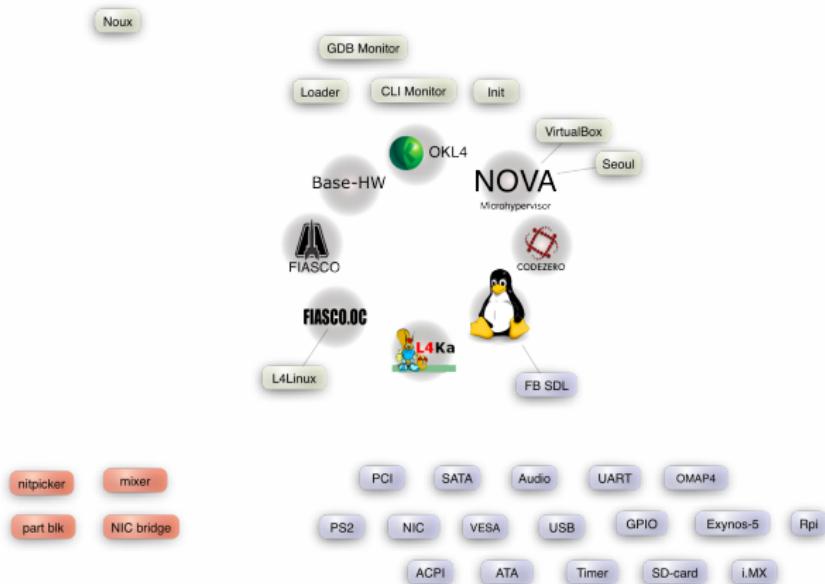


Components



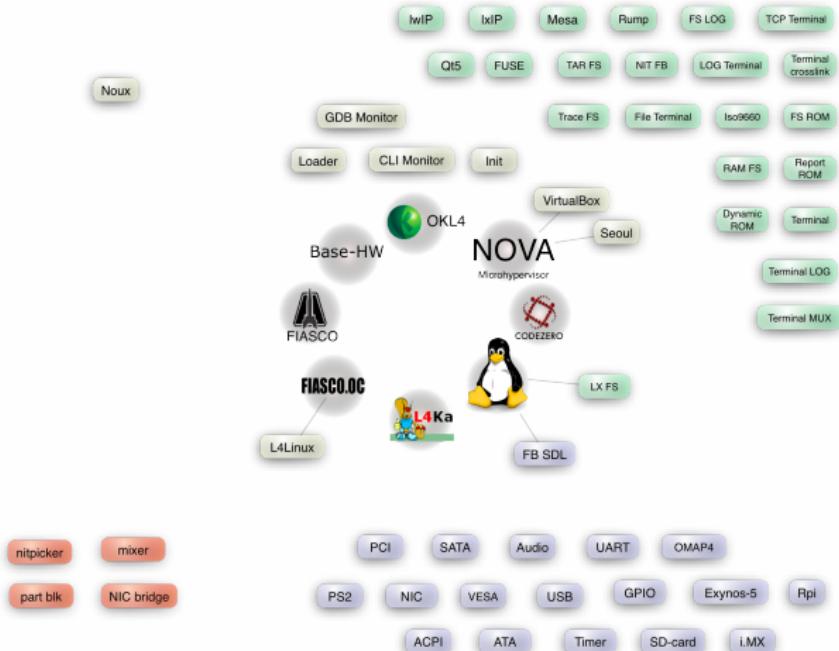


Components



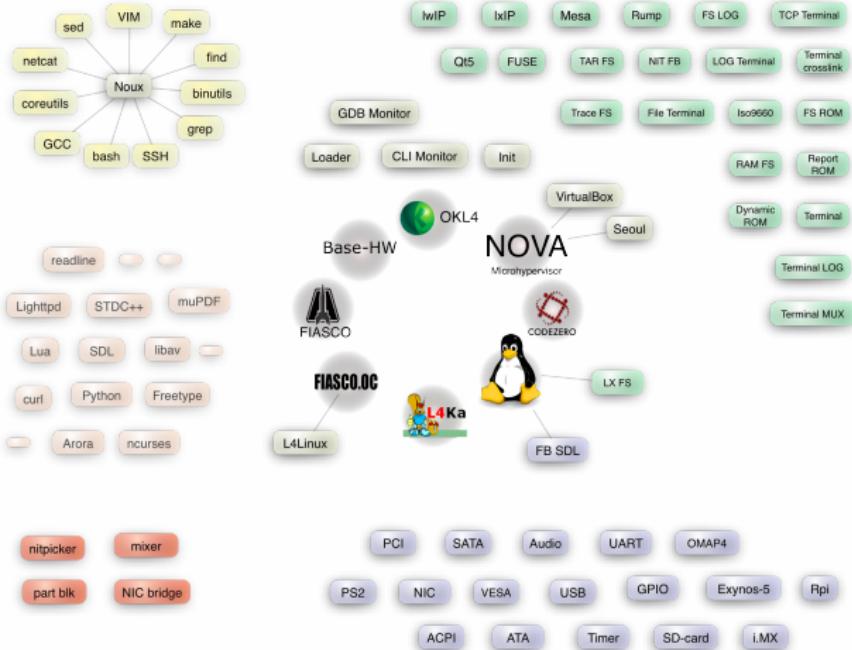


Components





Components





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





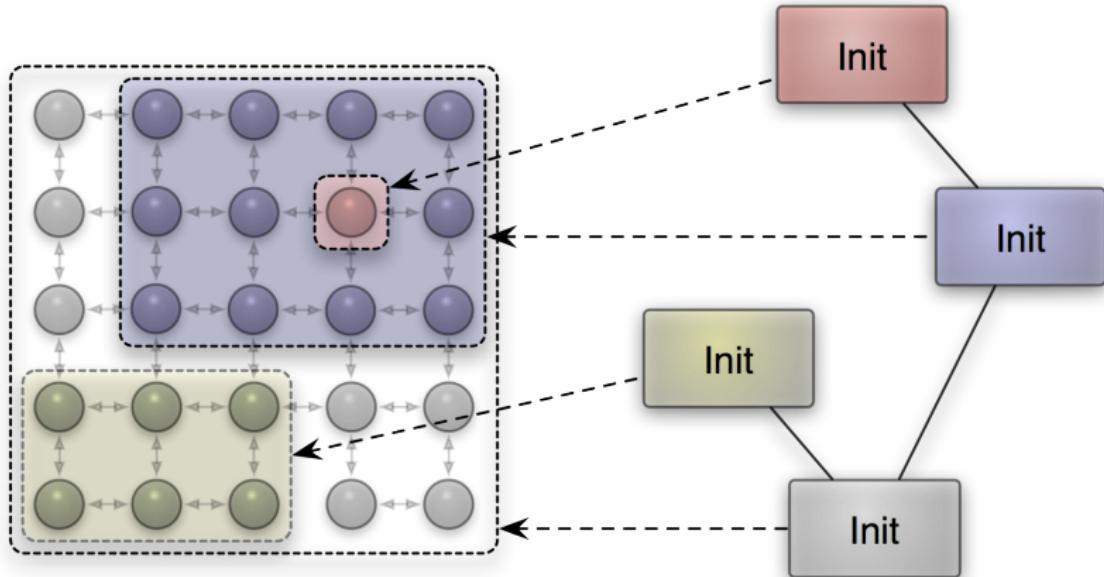
Fundamentals - NOVA kernel

- IOMMU support
- Kernel-memory reuse
- Multi-processor support





Fundamentals - Affinity management





Fundamentals - OS infrastructure

- Device drivers
 - essential drivers are in place
(NIC, graphics, input, USB, audio, SATA)
- File systems
 - ▶ per-process virtual file system
 - ▶ FUSE
 - ▶ Rump
- TCP/IP
 - ▶ lwIP
 - ▶ Linux TCP/IP for gigabit networking





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Functionality

- Simple CLI
- Virtualization as a stop-gap solution
 - ▶ Vancouver aka Seoul
 - ▶ VirtualBox
- Noux runtime for GNU software
- GNU debugger
- Qt5
 - ▶ Change from QWS to QPA
 - ▶ QML





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





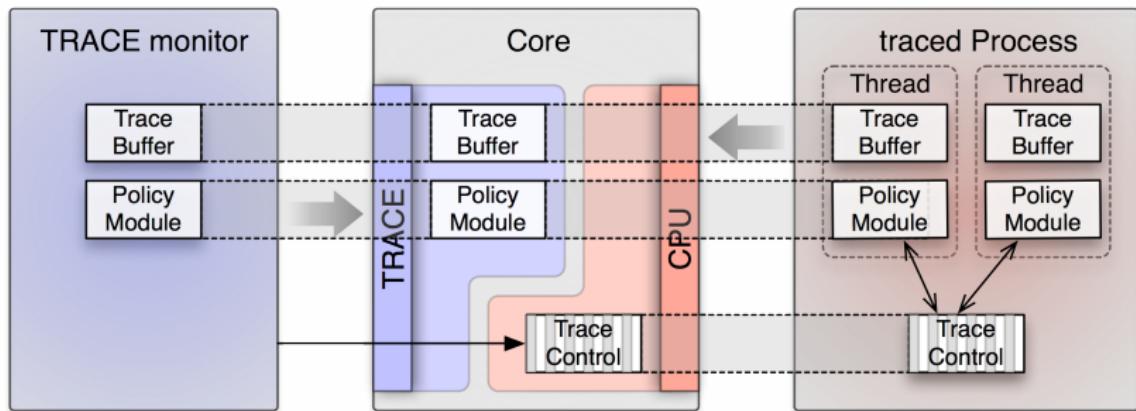
Tracing: Wish list

- Negligible performance overhead
- Kernel independence
- Accountability of used resources
- Useful level of abstraction
- Runtime-defined tracing policies
- Low-complexity implementation
- Online and offline analysis





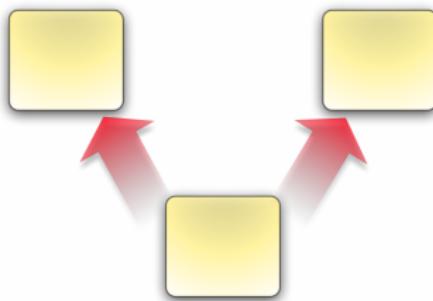
Tracing: Mechanism





Resource management

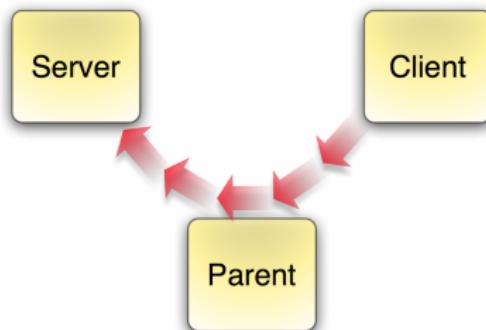
Explicit assignment of physical resources to processes





Resource management (II)

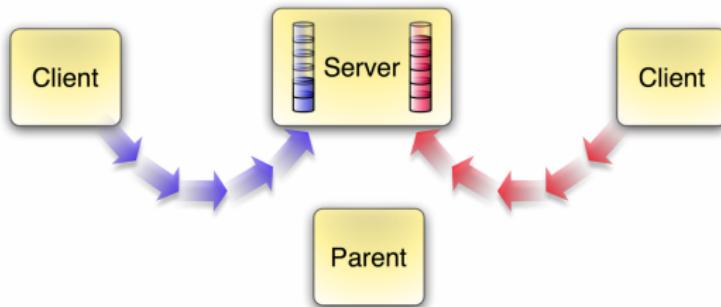
Resources can be attached to sessions





Resource management (III)

Server-side heap partitioning





Dynamic resource balancing

Not all use cases could be covered.

- Caches (i. e., block cache)
- Ballooning

→ *refined parent interface*





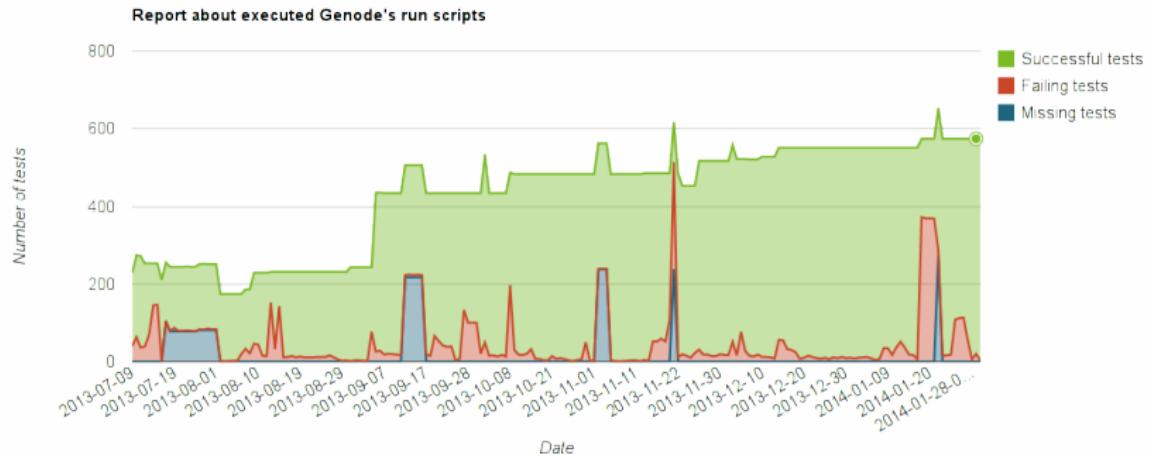
Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Automated tests





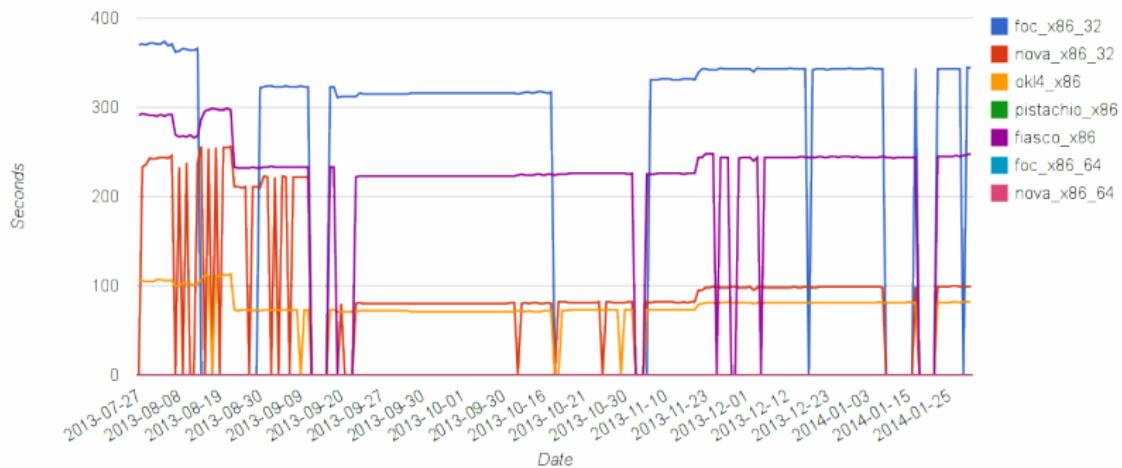
Automated tests (II)

Summary of native hardware runs

test	fiasco_x86	foc_arndale	foc_panda	foc_x86_32	foc_x86_64	hw_arndale	hw_imx33	hw_panda	nova_x86_32	nova_x86_64	ok14_x86	pistachio_x86
xml_generator	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
util_mmio	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
timer	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
timed_semaphore	ok	ok	ok	ok	ok	ok	error	error	error	ok	ok	ok
thread_join	ok	ok	ok	ok	ok	ok	error	error	error	ok	ok	ok
tar_rom	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
sub_rm	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
signal	ok	ok	ok	ok	ok	ok	error	error	error	ok	ok	ok
seadl-auto	-	-	-	-	-	-	-	-	-	ok	-	-
rom_bld	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
rm_fault	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	-	-
resource_yield	error	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
resource_request	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
python	ok	-	-	ok	ok	-	-	-	ok	ok	ok	ok
part_blk	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
noux_tool_chain_auto	ok	ok	ok	ok	-	error	-	error	ok	-	ok	-
noux_net_netcat	ok	ok	ok	ok	ok	error	ok	-	error	ok	ok	ok
noux	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
netperf_bxip_uah30	ok	ok	-	ok	ok	ok	-	-	ok	ok	ok	ok
netperf_bxip_bridge	error	ok	-	ok	ok	ok	-	-	ok	ok	ok	ok
netperf_bxip	ok	ok	ok	ok	ok	ok	error	-	ok	ok	ok	ok
netperf_bwip_uah30	ok	ok	-	ok	ok	ok	-	-	ok	ok	ok	ok
netperf_bwip_bridge	ok	ok	-	ok	ok	ok	-	-	ok	ok	ok	ok
netperf_bwip	ok	ok	ok	ok	ok	ok	-	-	ok	ok	ok	ok
mp_server	-	ok	ok	ok	ok	ok	-	-	ok	ok	ok	error
mem	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	-	-
lx_hybrid_pthread_ipc	-	-	-	-	-	-	-	-	-	-	-	-
lx_hybrid_exception	-	-	-	-	-	-	-	-	-	-	-	-
lx_hybrid_stacks	-	-	-	-	-	-	-	-	-	-	-	-
bwip	ok	ok	ok	ok	ok	ok	-	ok	ok	ok	ok	ok
libc_fat	-	-	-	-	-	-	-	-	-	-	-	-
ldso	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
4linnx_netperf_uah30	-	ok	-	-	-	-	-	-	-	-	-	-
4linnx_netperf_bridge	-	ok	-	-	-	-	-	-	-	-	-	-
4linnx_netperf	-	ok	-	-	-	-	-	-	-	-	-	-
4linnx	-	ok	ok	ok	-	-	-	-	-	-	-	-
gdb_monitor	-	-	-	-	-	-	-	-	-	-	-	-
failsafe	-	ok	ok	ok	ok	-	-	-	ok	ok	ok	ok
cleanall	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok	ok
affinity	-	ok	ok	ok	ok	-	-	-	ok	ok	-	-



Automated benchmarks





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Capability-based user interface

User interface that matches Genode's concepts

Ideas:

- Composed out of many small inter-changeable building blocks
- Data centric
- Capability-based
- Command-line and graphical interface





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014



Sidelines

- Samsung Exynos-5
(*SATA 3.0, USB 3, HDMI, eMMC, NIC, DVFS*)
- Freescale i.MX
- Raspberry Pi





Outline

1. Introduction
2. The long way towards general-purpose computing
 - Fundamentals
 - Functionality
 - Resource utilization
 - Stability
3. What is left to be desired?
4. Sidelines
5. Road map 2014





Road map 2014

- Capability-based user interface
- Custom base-hw kernel platform
 - ▶ MP support
 - ▶ Capability-based security
 - ▶ Virtualization
- 3rd-party source-code package management
- Improved block-level infrastructure
(block cache, block encryption)
- Wireless networking





Thank you

Genode OS Framework

<http://genode.org>

Genode Labs GmbH

<http://www.genode-labs.com>

Source code at GitHub

<http://github.com/genodelabs/genode>

