Genode as general-purpose OS progress report and demonstration

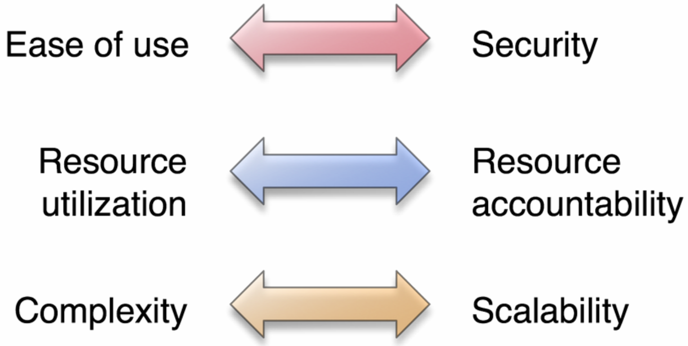
Norman Feske
<norman.feske@genode-labs.com>

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

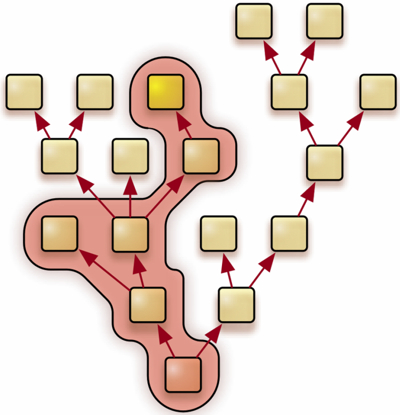
Myths


Ease of use
Security
Resource utilization
Resource accountability
Complexity
Scalability

Key technologies

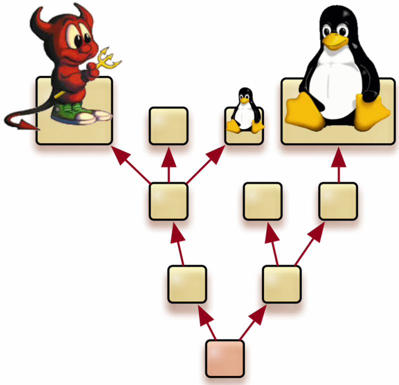
- Microkernels
- Decomponentization, kernelization
- Capability-based security
- Virtualization

Genode architecture



→ Application-specific TCB

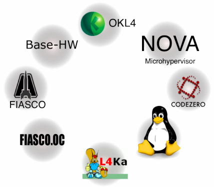
Combined with virtualization



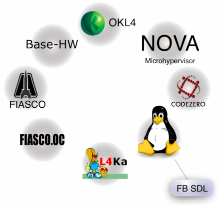
How to get there? Found a company!

- 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

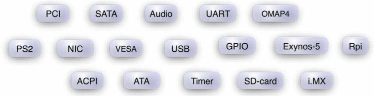
Components


OKL4
Base-HW
NOVA
Microhypervisor
FIASCO
CODEZERO
FIASCO.OC
L4Ka

Components

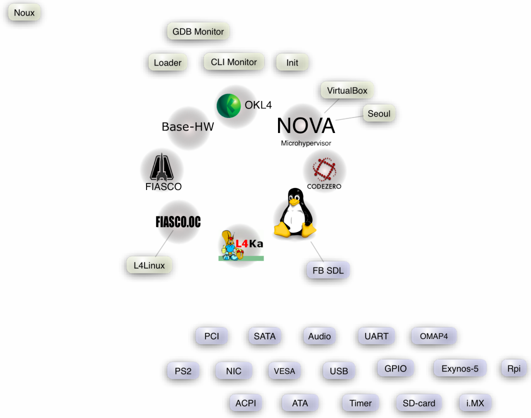


OKL4
Base-HW
NOVA
Microhypervisor
FIASCO
CODEZERO
FIASCO.OC
L4Ka
FB SDL

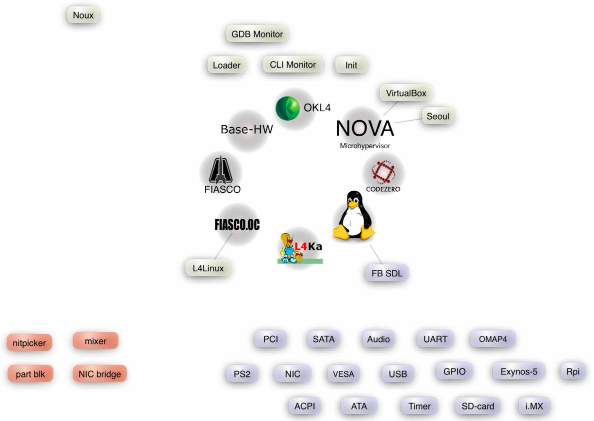


PCI
SATA
Audio
UART
OMAP4
PS2
NIC
VESA
USB
GPIO
Exynos-5
Rpi
ACPI
ATA
Timer
SD-card
i.MX

Components

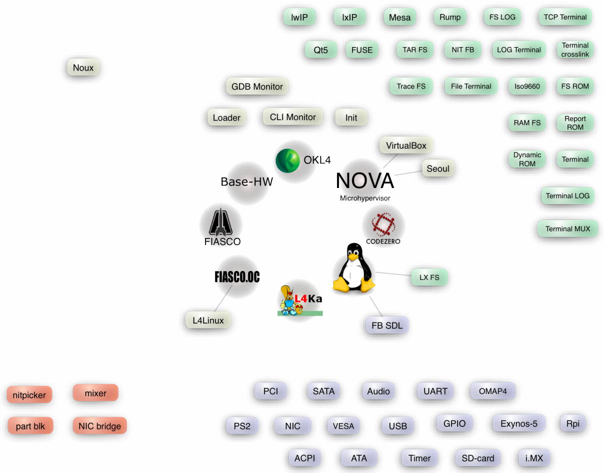

Noux
GDB Monitor
Loader
CLI Monitor
Init
VirtualBox
Seoul
OKL4
Base-HW
NOVA
Microhypervisor
FIASCO
CODEZERO
FIASCO.OC
L4Ka
L4Linux
FB SDL
PCI
SATA
Audio
UART
OMAP4
PS2
NIC
VESA
USB
GPIO
Exynos-5
Rpi
ACPI
ATA
Timer
SD-card
i.MX

Components

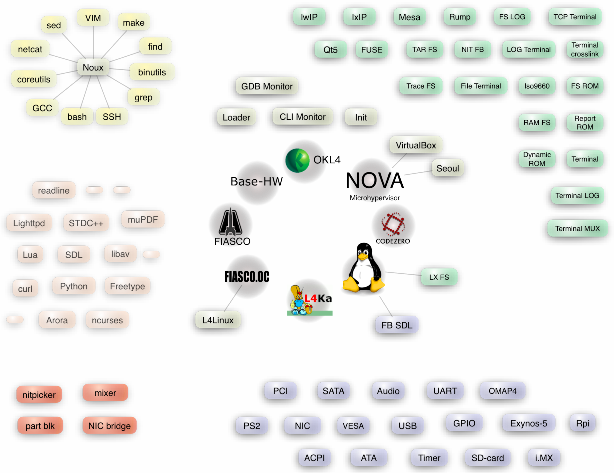


Noux
GDB Monitor
Loader
CLI Monitor
Init
VirtualBox
Seoul
OKL4
Base-HW
NOVA
Microhypervisor
FIASCO
CODEZERO
FIASCO.OC
L4Ka
L4Linux
FB SDL
nitpicker
mixer
part blk
NIC bridge
PCI
SATA
Audio
UART
OMAP4
PS2
NIC
VESA
USB
GPIO
Exynos-5
Rpi
ACPI
ATA
Timer
SD-card
i.MX

Components


lwIP
lxIP
Mesa
Rump
FS LOG
TCP Terminal
Qt5
FUSE
TAR FS
NIT FB
LOG Terminal
Terminal crosslink
Noux
GDB Monitor
Trace FS
File Terminal
Iso9660
FS ROM
Loader
CLI Monitor
Init
RAM FS
Report ROM
VirtualBox
OKL4
Seoul
Dynamic ROM
Terminal
Base-HW
NOVA
Microhypervisor
Terminal LOG
FIASCO
CODEZERO
Terminal MUX
FIASCO.OC
LX FS
L4Ka
L4Linux
FB SDL
nitpicker
mixer
part blk
NIC bridge
PCI
SATA
Audio
UART
OMAP4
PS2
NIC
VESA
USB
GPIO
Exynos-5
Rpi
ACPI
ATA
Timer
SD-card
i.MX

Components


Noux
VIM
sed
make
netcat
find
coreutils
binutils
grep
GCC
bash
SSH
readline
Lighttpd
STDC++
muPDF
Lua
SDL
libav
curl
Python
Freetype
Arora
ncurses
nitpicker
mixer
part blk
NIC bridge
lwIP
lxIP
Mesa
Rump
FS LOG
TCP Terminal
Qt5
FUSE
TAR FS
NIT FB
LOG Terminal
Terminal crosslink
GDB Monitor
Trace FS
File Terminal
Iso9660
FS ROM
Loader
CLI Monitor
Init
RAM FS
Report ROM
VirtualBox
OKL4
Seoul
Dynamic ROM
Terminal
Base-HW
NOVA
Microhypervisor
Terminal LOG
FIASCO
CODEZERO
Terminal MUX
FIASCO.OC
L4Ka
LX FS
L4Linux
FB SDL
PCI
SATA
Audio
UART
OMAP4
PS2
NIC
VESA
USB
GPIO
Exynos-5
Rpi
ACPI
ATA
Timer
SD-card
i.MX

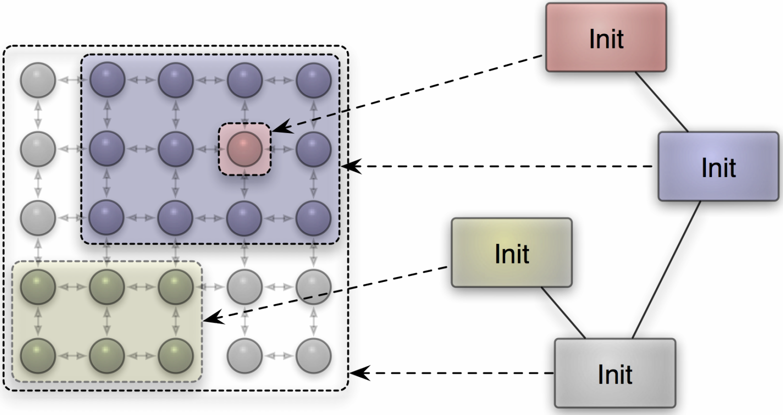
Outline

Outline

Fundamentals - NOVA kernel

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

Fundamentals - Affinity management



Fundamentals - OS infrastructure

- Device drivers
  $\rightarrow$ 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

Functionality

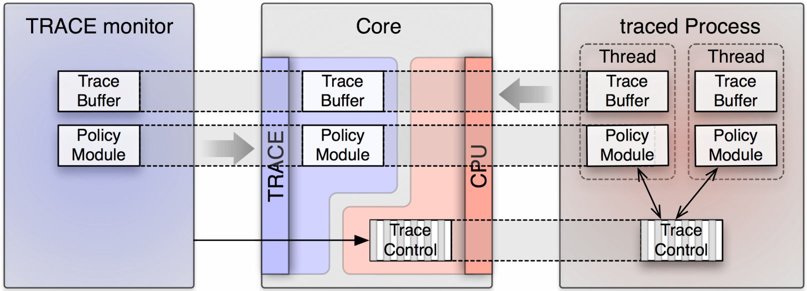
- Simple CLI
- Virtualization as a stop-gap solution
  - Vancouver aka Seoul
  - VirtualBox
- Noux runtime for GNU software
- GNU debugger
- Qt5
  - Change from QWS to QPA
  - QML

Outline

Tracing: Wish list

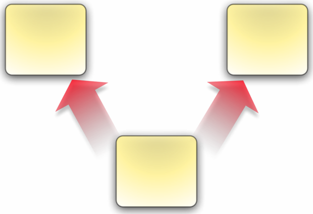
- Negligible performance overhead
- Kernel independence
- Accountability of used resources
- Useful level of abstraction
- Runtime-defined tracing policies
- Low-complexity implementation
- Online and offline analysis

Tracing: Mechanism



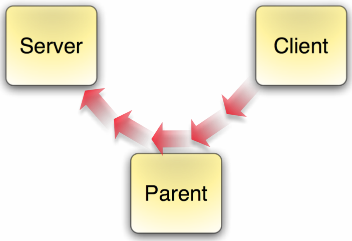
Resource management

Explicit assignment of physical resources to processes



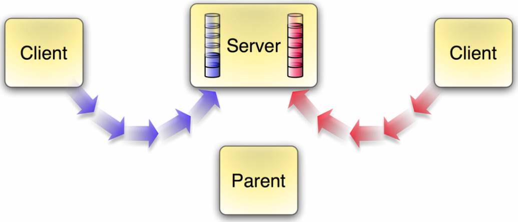
Resource management (II)

Resources can be attached to sessions



Resource management (III)

Server-side heap partitioning



Dynamic resource balancing

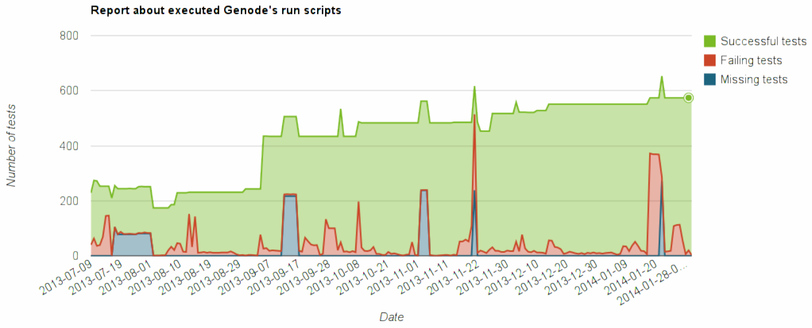
Not all use cases could be covered.

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

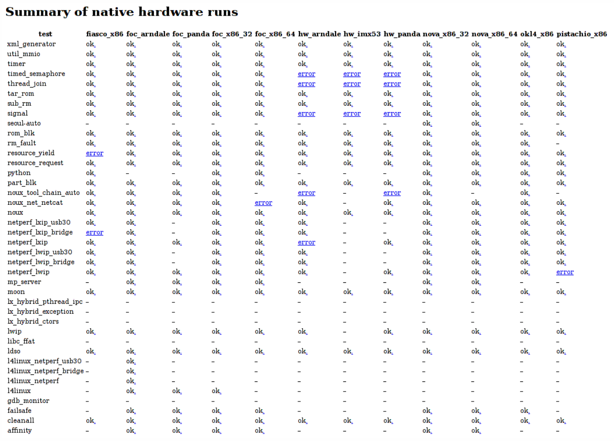
$\rightarrow$ *refined parent interface*

Outline

Automated tests



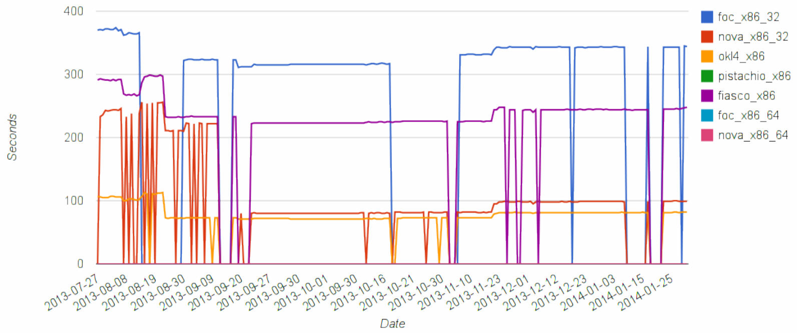
Automated tests (II)



Summary of native hardware runs

test	fiasco_x86	foc_arndale	foc_panda	foc_x86_32	foc_x86_64	hw_arndale	hw_imx53	hw_panda	nova_x86_32	nova_x86_64	okl4_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.	error	error	error	ok.	ok.	ok.	ok.
thread_join	ok.	ok.	ok.	ok.	ok.	error	error	error	ok.	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.	error	error	error	ok.	ok.	ok.	ok.
seoul-auto	–	–	–	–	–	–	–	–	ok.	ok.	–	–
rom_blk	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.
rm_fault	ok.	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.	error	ok.	–	ok.	ok.	ok.	ok.	ok.
noux	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.
netperf_lxip_usb30	ok.	ok.	–	ok.	ok.	ok.	–	–	ok.	ok.	ok.	ok.
netperf_lxip_bridge	error	ok.	–	ok.	ok.	ok.	–	–	ok.	ok.	ok.	ok.
netperf_lxip	ok.	ok.	ok.	ok.	ok.	error	–	ok.	ok.	ok.	ok.	ok.
netperf_lwip_usb30	ok.	ok.	–	ok.	ok.	ok.	–	–	ok.	ok.	ok.	ok.
netperf_lwip_bridge	ok.	ok.	–	ok.	ok.	ok.	–	–	ok.	ok.	ok.	ok.
netperf_lwip	ok.	ok.	ok.	ok.	ok.	ok.	–	ok.	ok.	ok.	ok.	error
mp_server	–	ok.	ok.	ok.	ok.	–	–	–	ok.	ok.	–	–
moon	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.
lx_hybrid_pthread_ipc	–	–	–	–	–	–	–	–	–	–	–	–
lx_hybrid_exception	–	–	–	–	–	–	–	–	–	–	–	–
lx_hybrid_ctors	–	–	–	–	–	–	–	–	–	–	–	–
lwip	ok.	ok.	ok.	ok.	ok.	ok.	–	ok.	ok.	ok.	ok.	ok.
libc_ffat	–	–	–	–	–	–	–	–	–	–	–	–
ldso	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.	ok.
l4linux_netperf_usb30	–	ok.	–	–	–	–	–	–	–	–	–	–
l4linux_netperf_bridge	–	ok.	–	–	–	–	–	–	–	–	–	–
l4linux_netperf	–	ok.	–	–	–	–	–	–	–	–	–	–
l4linux	–	ok.	ok.	ok.	–	–	–	–	–	–	–	–
gdb_monitor	–	–	–	–	–	–	–	–	–	–	–	–
failsafe	–	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

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

Sidelines

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

Outline

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