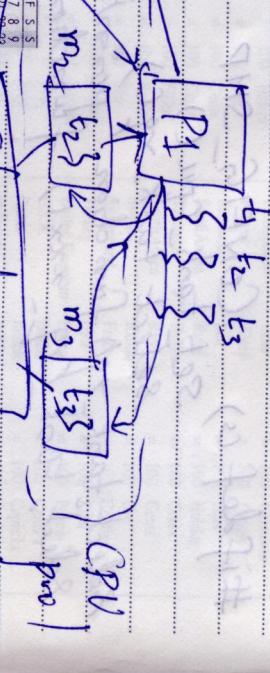


2 Wednesday each Scheduling class ('fair', real-time etc.,) task thread info) } Kernel/sched/fair.c. select task_fair() { Kernel/sched/fair.c. select_ife_bibling() } thread_idest_group() } from idest_cpus() }

Linux Mulligne Scheduler:
use source for next ('Linux Scalability Eff') Linux creates light weight process by clone() call. clone() underlies fork(), vfork(), clone()
3 Thursday Goal of VIRGO Linux:
To design a Linux with cloud capability, i.e. the kernel threads / light weight processes execute across CPUs in many machines in cloud. For example threads t_1, t_2, t_3 of a process p_1 executed in machines m_1, m_2, m_3 respectively.



1. Implementation of above should be a PRAM. Call CPU have access to a shared memory

2. Synchronization of threads across CPU is a nontrivial theory.

3. Communication cost across CPUs will be huge if CPUs communicate directly. Hence all data needs to be shared by memory.

Astro Infer (ASFER) 15/4/2013 Saturday 5

During past 10 years, a design and development of a reasonably complete inference model has been implemented to mine birth data - and has a code for Pathogens.

At present, given set of bones, a Sunday 6 belonging to particular genre of vertebrate a fact file is generated that encodes the bones, containing planets and abundant data. The fact file can have encoded bones slope shapes, which are Zodical (Starting

DEC	M	T	W	T	F	S	S	M	T	W	F	S
2012	10	11	12	13	14	15	16	17	18	19	20	21
	24	25	26	27	28	29	30	31				

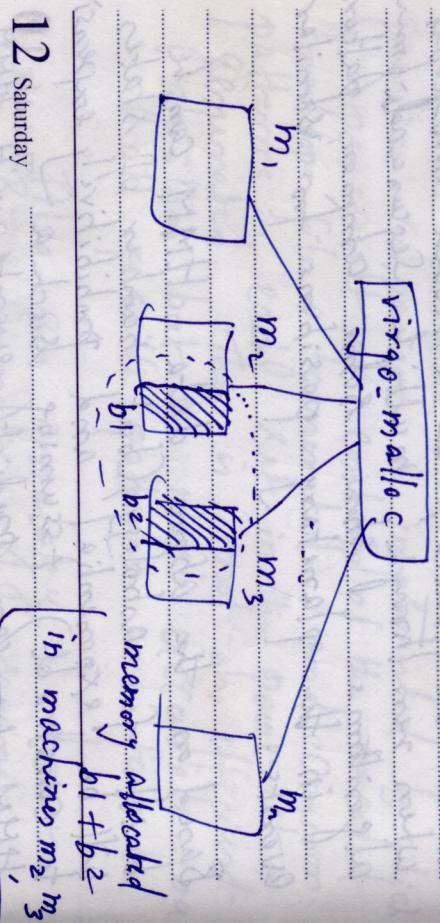
FEB	M	T	W	T	F	S	S	M	T	W	F	S
2013	11	12	13	14	15	16	17	18	19	20	21	22
	25	26	27	28	29	30	31					

Don't use `clone()` in
JANUARY
remote machine, 2013

do similar to the above. For example
2013 `piCloud` invokes a function
on host with:

Memory pooling:

A wrapper call over `malloc()` with name
`virgo_malloc` with same parameters but which
allocates from a pool of memory
scattered across a cloud.



12 Saturday

GPU pooling:

`virgo_close()` transparently executes
a function by creating a thread in one
of the GPUs in the pool in the network.

`virgo_close()` looks up a map of
machine to load and gets the best with
least load and invokes `RPC` on that
host. (to notify if `RPC` uses `clone()`)

Host of the Cloud APIs

Advantages of `virgo_clone()`

As example, code that uses `clone()`
can be replaced with `virgo_clone()` in all places
in the code that has `clone()` calls. With this, the
threads created are automatically executed on a cloud
and is part of Linux kernel.

17/4/2013

Tuesday 15

Advantages of `virgo_clone()`

`RPC` in Linux is part of kernel. It is
just an implementation of Sun `RPC`
protocol specification and is only a library
with set of functions.

invoke(`fn, hostname`)

Monday 14

DEC	M	T	W	F	S	S	M	T	W	F	S	S
2012	10	11	12	13	14	15	16	17	18	19	20	21
	24	25	26	27	28	29	30	31			22	23

DEC	M	T	W	F	S	S	M	T	W	F	S	S
2013	11	12	13	14	15	16	17	18	19	20	21	22
	24	25	26	27	28	29	30	31			23	24

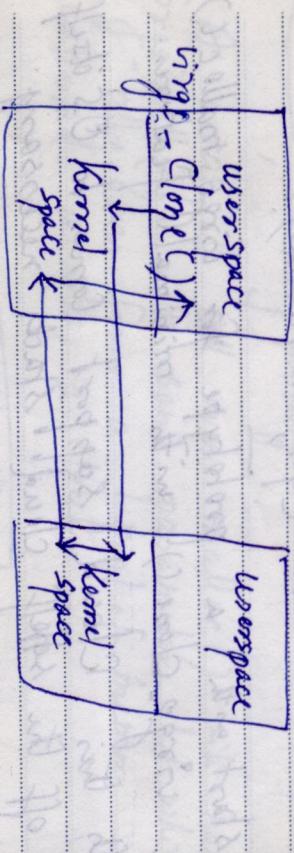
kernel_sendmsg(), kernel_recvmsg(), kernel_xm()

kernel_Socket communication, accept
etc.)

JANUARY Read linear/net/sum/pc • 2013

Focus / init/main.c
2013 - do basic setup C JANUARY
 do basic setup C

16 Wednesday



2/5/2013 : /tmp/ff/mfsd/mfscl.c

There is ~~one~~ ^{one} NFS - threads called? which does all the thread creation for NFS.

17 Thursday 3/5/2013 VIRGO Cloudy

Module ops need to be invoked by a
VIRGO Service (Similar to NES Service)
New VIRGO kernel service needs to be
written.

Ans. Powerful drivers/busy init. c
drives in. (c)

Source/Answers/base/cont.c
Doubtful

DEC	M	T	W	T	F	S	M	T	W	F	S
2012	10	11	12	13	14	15	16	17	18	19	20
	24	25	26	27	28	29	30	31			

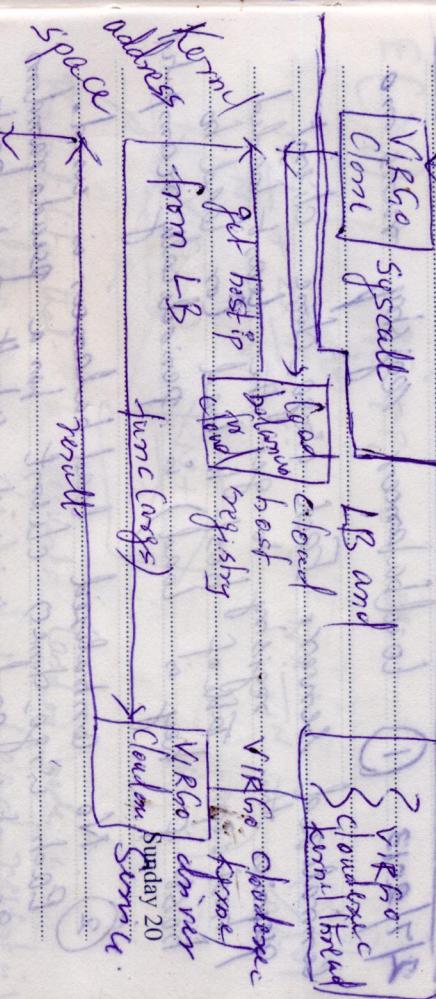
Quesadilla: This `doLingo init()` function would create a kernel thread for `doLingo Cloud exec service` that runs as a daemon thread when a machine boots up.

None of these kernel threads in Virgo (service C) does module self-loading on the Virgo cloudsec kernel object and waits for connections.
if necessary above can be #ifdefed)

17/7/2013 VI.R6.0 - Cloud load balancer design

of Application User address space 1024000

Saturday 19



VIRGO syscall
 Close] syscall
 LB and
 forced
 before best
 putting in
 Registering
 VIRGO Close
 the file

from L.B. 100 -
func (args) 100 -
V.R.G. 100 -
Driver
Cloudy Sunday 20
Summer.

→ Primal →

Thus expected number of requests at a node i

$$= \sum \text{Value of } r.v \cdot \text{Pr}(r.v)$$

$$= 1 \times \frac{1}{N} + 2 \times \frac{1}{N^2} + \dots + k \times \frac{1}{N^k}$$

$$\sum_k \frac{k}{N^k}$$

$$\frac{1}{N} + \left(\sum_{k=1}^{\infty} \frac{k}{N^k} \right) + \frac{1}{N(N-1)} = \sum_k \frac{1}{N^k}$$

$$\frac{1}{N} + \frac{2}{N^2} + \frac{3}{N^3} + \dots + \frac{k}{N^k} = \sum_k \frac{1}{N^k}$$

$$26 \text{ Saturday } \frac{1}{N} + \frac{1}{N} \left(\frac{2}{N} + \frac{3}{N^2} + \dots + \frac{N}{N^{k-1}} \right) = \sum_k \frac{1}{N^k}$$

$$\frac{1}{N} + \frac{1}{N} \left(\frac{1+1}{N} + \frac{2+1}{N^2} + \dots + \frac{k-1+1}{N^{k-1}} \right) = \sum_k \frac{1}{N^k}$$

$$\frac{1}{N} + \frac{1}{N} \left(\frac{1}{N} + \frac{2}{N^2} + \dots + \frac{k-1}{N^{k-1}} + \frac{1}{N^k} \right) = \sum_k \frac{1}{N^k}$$

$$\frac{1}{N} + \frac{1}{N} \left(\sum_{k=1}^{\infty} \frac{1}{N^k} + \frac{N}{N-1} \right) = \sum_k \frac{1}{N^k}$$

DEC	M	T	W	T	F	S	M	T	W	F	S
2012	10	11	12	13	14	15	16	17	18	19	20

30. Wednesday $N^k - N + 1$ is the required

As number of requests $N^{k+1} + N^k$ $\rightarrow \infty$ out of k requests

$$k \rightarrow \infty \quad N + \frac{1}{N}$$

$$31 \quad \text{Thursday} \quad \frac{1}{N} + \frac{1}{N^2} + \frac{1}{N^3} + \dots + \frac{1}{N^k}$$

$$\frac{1}{N} \left(1 - \frac{1}{N} \right)^{k+1} + 1 + \left(1 - \frac{1}{N} \right)$$

31 Thursday

$$\frac{N \left(1 - \frac{1}{N^{k+1}} \right)}{N-1} = \frac{N^{k+1}}{N^{k+2} - N^{k+1}}$$

$$\frac{N^{k+1} \times \frac{N^2 + kN + k^2 + 1}{N-1}}{N^{k+1} (N-1)^2} = \frac{N^{k+1} (N^2 + kN + k^2 + 1)}{N^{k+1} (N-1)^2} = \frac{N^2 + kN + k^2 + 1}{N-1}$$

$$\frac{N^{k+2} - kN^k + N^k - 1}{N^{k+3} - 2N^k + N^k} = \text{Sum}_k = F(\text{Reg})$$

DEC	M	T	W	T	F	S	S	M	T	W	T	F	S
	10	11	12	13	14	15	16	17	18	19	20	21	22
2012	24	25	26	27	28	29	30	31					

$$\frac{1}{N} + \frac{1}{N} \left(\sum_k - \frac{k}{N^k} \right) + \frac{N^{k+1} - 1}{N^{k+3} - N^{k+2}} = \sum_k$$

$$\frac{1}{N} - \frac{k}{N^{k+1}} + \frac{N^k - 1}{N^{k+2}(N-1)} = \sum_k \left(\frac{1}{N} \right)$$

$$\frac{k+1(N-1) - kN(N-1) + N}{N(N-1)} = \sum_k \left(\frac{N-1}{N^2} \right)$$

Saturday 2

$$\frac{N^{k+1} - kN^2 + kN + k - 1}{N-1} = \text{Sum}_k$$

$$\frac{N^{k+1} - kN^2 + N}{N^{k+1}(N-1)^2} = \sum_k$$

Sunday 3

MARCH

8 Friday

→ Recursion 3 2013

If $\frac{N}{\log^c N}$ is set to a fixed starting point then no recursion MARCH is needed.

Monday 11

$$\frac{N}{\log^c N} = 1 \quad N = \log^c N$$

$$\log N = C \log \log N \quad (11+2)(11-2)$$

$$\log N = \frac{\log N}{\log \log N}$$

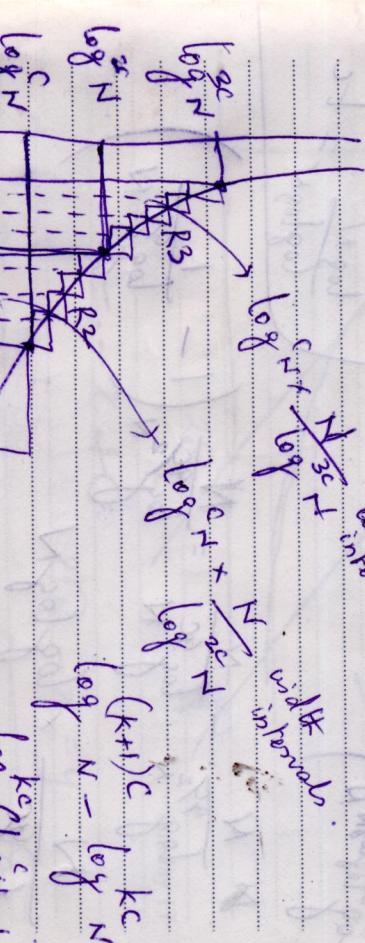
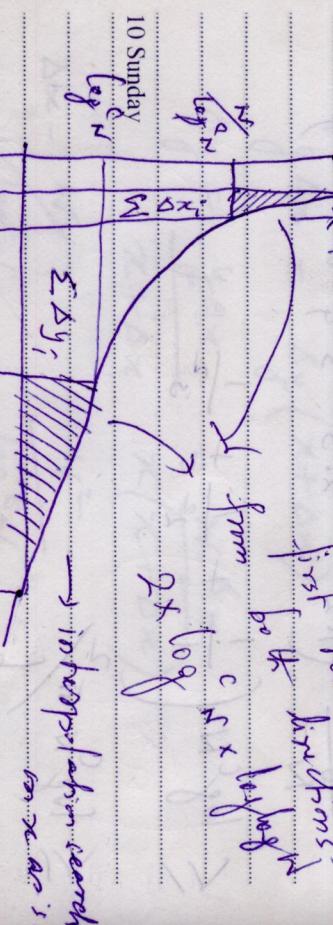
$$\log^c (N) + \log^{2c+1} (N) + \log^{3c+1} (N) + \dots$$

$$\frac{N}{\log^c N} - \frac{N}{\log^{2c} N} = \frac{N \log^c N - N}{\log^{2c} N}$$

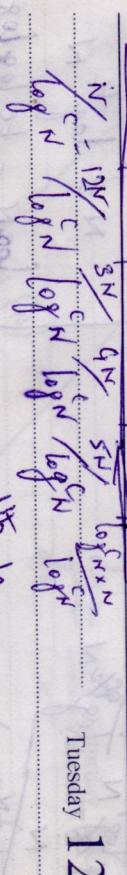
9 Saturday

$$\text{interpolation search in } \log^c N - 1$$

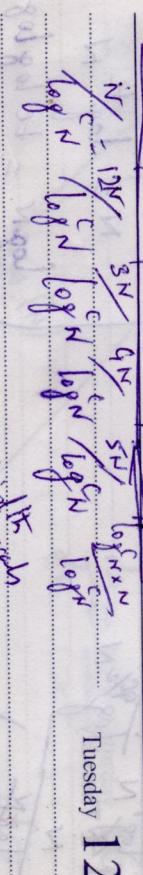
first recursion level
from bottom to top $\log^c N$



10 Sunday



11 Monday



M	T	W	F	S	M	T	F	S
2	3	4	5	6	7	8	9	10
11	12	13	14	15	16	17	18	19
2013	21	22	23	24	25	26	27	28

FEB	M	T	W	F	S	M	T	F	S
11	12	13	14	15	16	17	18	19	20
2013	21	22	23	24	25	26	27	28	29