CSC104 Winter 2013

Why and how of computing
week 10

Danny Heap

heap@cs.toronto.edu

BA4270 (behind elevators)

http://www.cdf.toronto.edu/~heap/104/F12/

416-978-5899

Text: Picturing Programs

Computer Science
UNIVERSITY OF TORONTO
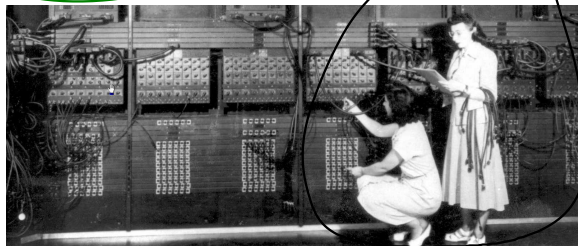
Outline

machines take over in batches

Machines began to take over setting the program counter to a new job, collecting the output, fetching memory... but it was still one job at a time.

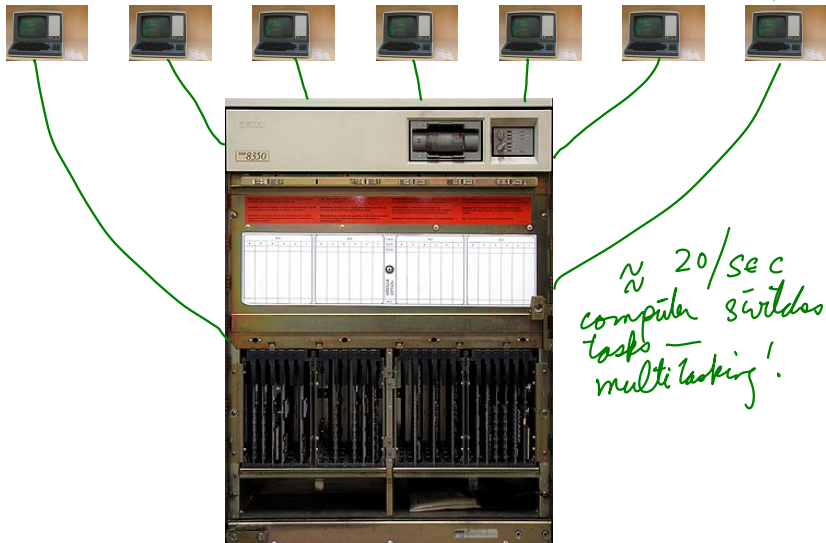as late 1980s - programs waited their turn
— batch.

time sharing, version 0.1

chat.

spread shed

word proces.

$\frac{60p}{6bs} \approx 10/s$

$\approx 20/sec$
computer switches
tasks —
multi tasking!

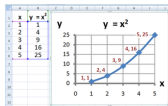# one user, one program, one computer

file 1970, 1980s

PC

apple

commodore '64

ran program (eg word processor), finished,
turned program off + started another,
made hardware accessible to user +
programs.   file → fdename.

OS

task-switching to time-splitting, v 1.0

mid - 1980

revolutionary

word processor

↓

print a page

use spreadsheet

freeze word proc

Does one task stop, or only appear to stop, for the other?

- wanted multi-tasking, like 1960s but personalized

unix (mostly) to the desktop
GUIs, time-sharing, networking, flame-wars

or unix-like

linux

Windows

Mac OS

artists,
musicians, multi-media

masses,
work-related.

programmers,
hobbyists,
academics.

an operating system should have

command-line
GUI — start programs, get info

→ users
programmers.

▶ kernel (shell, shielded access to hardware, referee sharing)

between programs

▶ utilities] - backup files
- maintain & install hardware/software

roundly connected

ring topology

given direction
sender gets
back as
confirmation
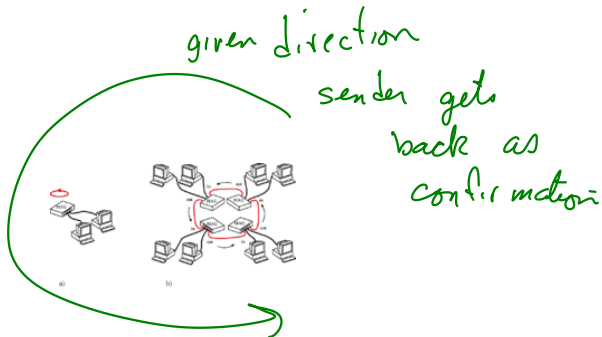
wait for the token

centrally connected
star configuration

hub could fail.

receiver

hub

Me, sending
info
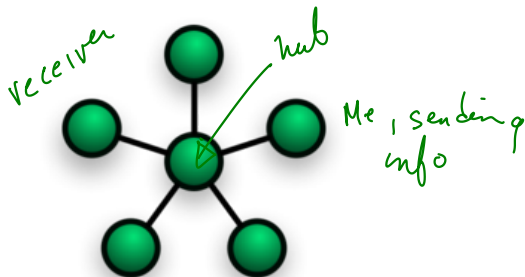
server runs things

all connected

bus configuration
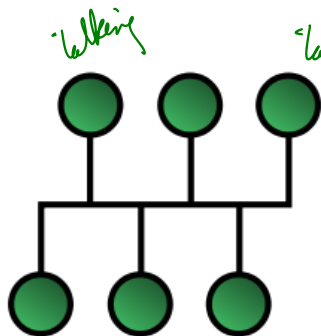
cooperate and back off

network of networks
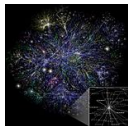
local networks
interconnected by
gateways
(click images)
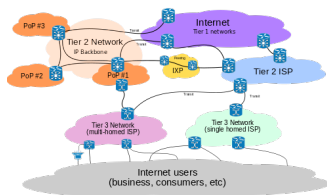
the medieval internet

NSFNET T3 Network 1992
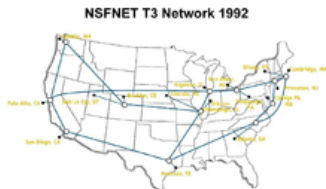
- ▶ email
- ▶ file transfer protocol
- ▶ Network File Service
- ▶ Tim Berners-Lee: WWW impossible without open protocols

Computer Science
UNIVERSITY OF TORONTO

now the internet ≈ WWW

antipodal clicking. . .

▶ where's the content?

▶ how's it move around

▶ who's in charge?

Notes