





# Organizational Information

- **Website:**

- <http://www.christianbaun.de>

- **E-Mail:**

- [christianbaun@fb2.fra-uas.de](mailto:christianbaun@fb2.fra-uas.de)

- **Course material:**

- All course material can be found at the course website

## !!! ATTENTION !!!

- Beginning WS2021, the cloud computing course does not have a written exam anymore!
- Your grade will depend 100% on your work and the results in the semester project (see the course web page for more information)

# Generations of Computer Systems

Generation	Timeframe	Technological progress
0	until 1940	(Electro-)mechanical calculating machines
1	1940 – 1955	Electron tubes, relays, jack panels
2	1955 – 1965	Transistors, batch processing
3	1965 – 1980	Integrated circuits, time sharing
4	1980 – 2000	Very large-scale integration, Microprocessors, PCs/Workstations
5	2000 until ?	Distributed systems, <i>the network is the computer</i> , Virtualization

Quote from the magazine *Popular Mechanics* (1949)

„In the future, computers may weigh no more than 1.5 tonnes.“

## 5. Generation (2000 – ????)

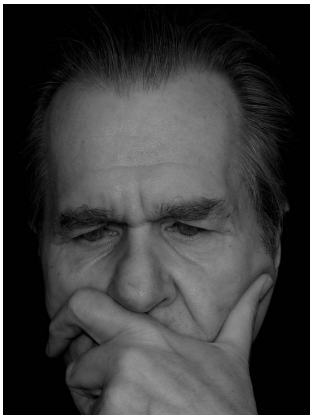
- Some keywords from the 5th generation:
  - *The network is the computer*
  - Distributed systems  $\implies$  **Cluster-, Cloud-, Grid-, P2P-Computing**
  - Multicore processors and **parallel applications**
  - Virtualization  $\implies$  **VMware, XEN, KVM, Docker...**
  - OpenSource  $\implies$  **Linux, BSD,...**
  - Communication everywhere  $\implies$  mobile systems, pervasive computing
  - New ways of working  $\implies$  e-Science, e-Learning, e-Business,...
  - Services  $\implies$  Service-oriented architectures (SOA), **Web Services**
  - Resources are requested and rent when needed  $\implies$  **on demand**
  - Artificial Intelligence (AI)

Many topics of the 5th generation will be discussed in this course

- Keywords for later generations:
  - Quantum computers (probably 7th or 8th generation)

# Brave new World?

Image Source: pixabay.com (CC0)



- Brings the concept *the network is the computer* only benefits?
- Who decides, which applications we are allowed to use in the future?
- Who decides, which books we were allowed to read in the future?
- How much freedom and self-determination do we want and need to give up?
  
- Some examples. . .



← → ↻ [www.telegraph.co.uk/technology/apple/7290849/Apple-removes-5000-apps-from-App-Store.html](http://www.telegraph.co.uk/technology/apple/7290849/Apple-removes-5000-apps-from-App-Store.html)

## Apple removes 5,000 apps from App Store

Apple has banned thousands of apps from the App Store, blaming inappropriate content



Developers report that Apple has started an App Store crackdown against apps featuring 'overtly sexual' content

By Claudine Beaumont, Technology Editor

11:07AM GMT 22 Feb 2010

 Comment

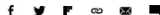
Apple has removed around 5,000 apps from its App Store, including some that it claims feature "overtly sexual" content.

Dozens of developers received a message from Apple stating that the company was refining the guidelines under which the App Store

← → ↻ [Sicher | https://www.cnet.com/news/commodore-64-iphone-emulator-approved-yanked/](https://www.cnet.com/news/commodore-64-iphone-emulator-approved-yanked/)

## Apple approves controversial Commodore 64 emulator only to remove it days later, after users find a work-around to access the BASIC interpreter.

BY DAVID MARTIN / SEPTEMBER 8, 2009 4:38 PM PDT



The Commodore 64 emulator application for iPhone, [previously rejected by Apple](#), was approved for availability in the App Store over the weekend, only to get pulled days later.

Apple blocked the sale of the iPhone app, dubbed C64, from the store on Tuesday without explaining why, according to developer [Manomio](#). And while Apple was not immediately available for comment regarding the C64 app, which is designed to enable users to play classic Commodore 64 games and run applications, Manomio says it believes that the yanking is related to an [available work-around](#) that enables users to activate the Commodore BASIC interpreter, a feature behind the application's initial App Store rejection.







📄 googlemobile.blogspot.de/2011/03/update-on-android-market-security.html

## An Update on Android Market Security

Saturday, March 5, 2011 | 10:08 PM

On Tuesday evening, the Android team was made aware of a number of malicious applications published to Android Market. Within minutes of becoming aware, we identified and removed the malicious applications. The applications took advantage of known vulnerabilities which don't affect Android versions 2.2.2 or higher. For affected devices, we believe that the only information the attacker(s) were able to gather was device-specific (IMEI/IMSI, unique codes which are used to identify mobile devices, and the version of Android running on your device). But given the nature of the exploits, the attacker(s) could access other data, which is why we've taken a number of steps to protect those who downloaded a malicious application:

1. We removed the malicious applications from Android Market, suspended the associated developer accounts, and contacted law enforcement about the attack.
2. We are remotely removing the malicious applications from affected devices. This remote application removal feature is one of many security controls the Android team can use to help protect users from malicious applications.
3. We are pushing an Android Market security update to all affected devices that undoes the exploits to prevent the attacker(s) from accessing any more information from affected devices. If your device has been affected, you will receive an email from android-market-support@google.com over the next 72 hours. You will also receive a notification on your device that "Android Market Security Tool March 2011" has been installed. You may also receive notification(s) on your device that an application has been removed. You are **not** required to take any action from there; the update will automatically undo the exploit. Within 24 hours of the exploit being undone, you will receive a second email.
4. We are adding a number of measures to help prevent additional malicious applications using similar exploits from being distributed through Android Market and are working with our partners to provide the fix for the underlying security issues.

# Google Chrome OS

## Releasing the Chromium OS open source project

11/19/2009 10:31:00 AM

In July we [announced](#) that we were working on Google Chrome OS, an open source operating system for people who spend most of their time on the web.

Today we are open-sourcing the project as Chromium OS. We are doing this early, a year before Google Chrome OS will be ready for users, because we are eager to engage with partners, the open source community and developers. As with the Google Chrome browser, development will be done in the open from this point on. This means the code is free, accessible to anyone and open for contributions. The Chromium OS project includes our current [code base](#), [user interface experiments](#) and some initial [designs](#) for ongoing development. This is the initial sketch and we will color it in over the course of the next year.

We want to take this opportunity to explain why we're excited about the project and how it is a fundamentally different model of computing.

First, it's all about the web. **All apps are web apps.** The entire experience takes place within the browser and there are no conventional desktop applications. This means users do not have to deal with installing, managing and updating programs.

# Amazon Kindle

Image Source: pixabay.com (CC0)



- Books can only be read with devices which are registered to a common Amazon account
- Sharing books is impossible
- Amazon can deny access to already purchased books

← → ↻ ⓘ www.nytimes.com/2009/07/18/technology/companies/18amazon.html

## Amazon Erases Orwell Books From Kindle

By [BRAD STONE](#)

Published: July 17, 2009

In [George Orwell](#)'s "1984," government censors erase all traces of news articles embarrassing to Big Brother by sending them down an incineration chute called the "memory hole."

On Friday, it was "1984" and another Orwell book, "Animal Farm," that were dropped down the memory hole — by [Amazon.com](#).

In a move that [angered customers](#) and generated waves of online pique, [Amazon](#) remotely deleted some digital editions of the books from the [Kindle](#) devices of readers who had bought them.

An Amazon spokesman, Drew Herdener, said in an e-mail message that the books were added to the Kindle store by a company that did not have rights to them, using a self-service function. "When we were notified of this by the rights holder, we removed the illegal copies from our systems and from customers' devices, and refunded customers," he said.

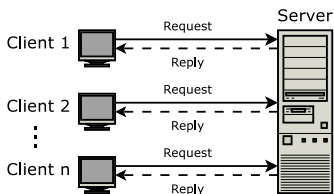
# Freedom vs. censorship

- Who decides in the future about censorship and freedom?
  - **Politics?**
    - <https://en.wikipedia.org/wiki/Censorship>
  - **Industry?**
    - [https://en.wikipedia.org/wiki/Censorship\\_by\\_Apple](https://en.wikipedia.org/wiki/Censorship_by_Apple)
    - [https://en.wikipedia.org/wiki/Censorship\\_by\\_Google](https://en.wikipedia.org/wiki/Censorship_by_Google)
    - [https://en.wikipedia.org/wiki/Censorship\\_of\\_Twitter](https://en.wikipedia.org/wiki/Censorship_of_Twitter)
  - **Population (customers/citizens)?**
    - [https://en.wikipedia.org/wiki/Freedom\\_of\\_speech](https://en.wikipedia.org/wiki/Freedom_of_speech)
    - [https://en.wikipedia.org/wiki/Political\\_freedom](https://en.wikipedia.org/wiki/Political_freedom)
    - [https://en.wikipedia.org/wiki/Artistic\\_freedom](https://en.wikipedia.org/wiki/Artistic_freedom)
    - [https://en.wikipedia.org/wiki/Intellectual\\_freedom](https://en.wikipedia.org/wiki/Intellectual_freedom)



# Client-Server

- A client-server system includes:
  - one or more **clients** which use the services of the server and accesses data, stored at the server ( $\implies$  consumers)
  - a **server**, which provides services and/or data ( $\implies$  producer)
- The connection establishment is initiated by the clients
- Communication works according to a protocol



- A client sends a request to the server responds with a reply
- The client-server architecture consists of 2 layers and is called two-tier model (tier = layer)

# Tasks in the Client-Server Model

- For a distributed application, that is based on the client-server architecture, 5 tasks can be separated from each other:
  - Display (graphical) user interface
  - Calculation of the (graphical) user interface
  - Data processing
  - Data management
  - Data storage
- The distribution of the tasks to clients and server determines the client types
- According to their areas of responsibility, 4 types of clients exist:
  - 1 **Text-/X-Terminals**
  - 2 **Thin/Zero Clients**
  - 3 **Applet Clients**
  - 4 **Fat Clients**

# Four Types of Clients in the Client-Server Model

## ● **X-Terminal** or **Text-Terminal**

- Only display the (graphical) user interface and transfer the user interaction to the server
- Calculation of the (graphical) user interface, data processing and data storage, data management are tasks of the server

## ● **Thin Clients** or **Zero Clients**

- Calculate and display the graphical user interface

## ● **Applet Clients** or **Network Computers**

- Calculate and display the graphical user interface and do a part of the data processing
- The clients process the applications (applets) themselves

## ● **Fat Clients**

- Only data management and data storage are located on the (file or database) server

# (Text-)Terminal: WYSE WY-50 (early 1990s)

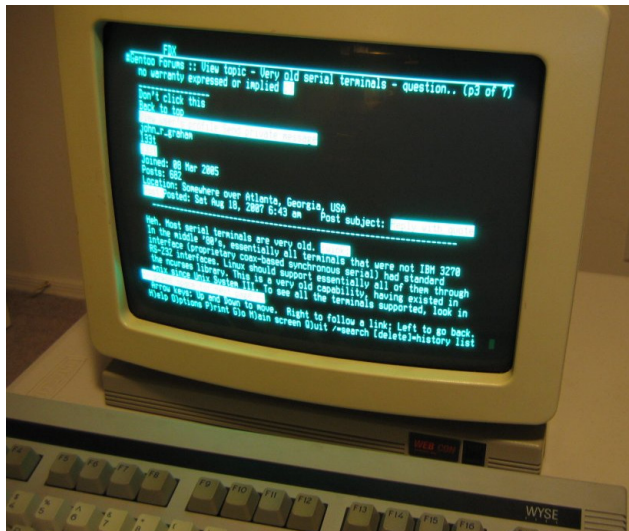


Image source:  
Google  
image search

# (X-)Terminal: NCD 88K (mid-1990s)



Image source: [http://en.wikipedia.org/wiki/X\\_terminal](http://en.wikipedia.org/wiki/X_terminal) and [http://www.geekdot.com/uploads/images/m88k/NCD17c/NCD88k\\_KDE.jpg](http://www.geekdot.com/uploads/images/m88k/NCD17c/NCD88k_KDE.jpg)

# Network Computer: SUN JavaStation (1996 – 2000)

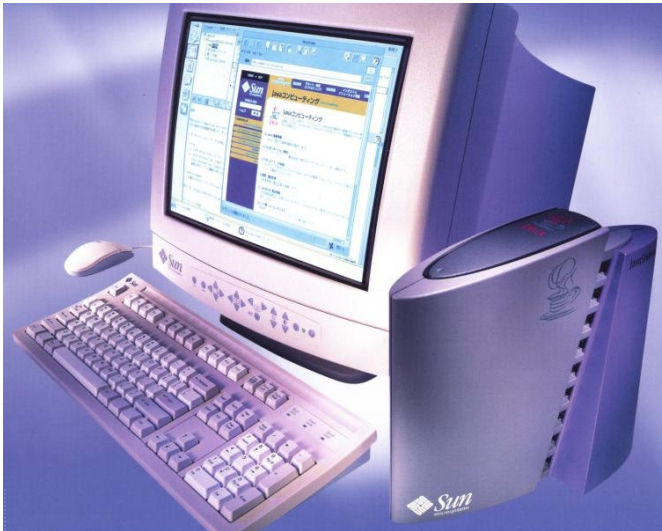


Image source:  
Google  
image search

# Fat Clients: Available everywhere

Image source: Google image search



**ALDI informiert** ...ab Mittwoch, 24. März

**ECHTE 3.0 GHz** 

**PC TITANIUM XL MD 8083**  
WIRELESS MULTIMEDIA HOME ENTERTAINMENT DESIGN CENTER

**999,-**  
3 Jahre Garantie

**AIR XL**  
Besonders leise durch das optimierte AIRXL Belüftungssystem.  
Neueste Lüftertechnologie macht den Medion® TITANIUM MD 8083 flüsterleise.

**Intel® Pentium® 4 Prozessor**  
mit Hyper-Threading-Technologie  
800 MHz Front Side Bus

**ATI Radeon 9800 XXL**  
128 MB DDR SDRAM  
256 Bit Memory Interface

**WLAN** 54MBit/s IEEE 802.11g  
802.11g kompatibel  
Drahtloser Netzwerkzugang\*, der perfekte Einstieg in die kabellose Netzwerktechnologie zu Hause und im Büro.  
\*Abhängigkeit des Standards befindet sich ein WLAN Access Point

**Pioneer Multi-Standard DVD/CD-Brenner**  
16x DVD-ROM

**Seagate 200 GB Festplatte**  
Ultra ATA 100 Interface  
Superschnelle 7200 U/Min.

**SAMSUNG 512 MB Speicher**  
DDR 400 MHz,  
54 Bit dual channel memory | 1 Bank frei zur Aufrüstung!

**CONNECT XL**






# Thin Clients

Image source: HP



# More Thin Clients

The screenshot shows the IGEL website's product page for thin clients. It features a navigation bar with categories like 'GRÜNDE FÜR IGEL', 'PRODUKTE', 'LÖSUNGEN', etc., and a 'DOWNLOAD' button. Below the navigation, five thin client models are displayed: UD Pocket, UD2, UD3, UD6, and UD9. Each model is accompanied by an image and a brief description of its use cases. At the bottom, a comparison table provides technical specifications for each model.

	UD Pocket	UD2	UD3	UD6	UD9
<b>UD Pocket</b>					
	Pool-Zugriff auf VDI - immer und überall	Office-Anwendungen HD-Videostreaming	HD-Videostreaming Ultra HD 4K-Inhalt Office-Anwendungen	CAD- und 3D- Videokonstruktion UC mit hoher Auflösung (720p)	CAD- und 3D- Videokonstruktion UC mit hoher Auflösung (720p)
CPU	Erfordert X86 64-Bit-Unterstützung	Intel Atom E3815 1,46 GHz (Single-Core)	AMD Steppe Eagle GX-424CC 2.4 GHz (Quad-Core)	Intel Celeron J1900 1,99-2,42 GHz (Quad-Core)	Intel Celeron J1900 1,99-2,42 GHz (Quad-Core)
Stromverbrauch (Leerlauf   Standby)		5 W   0,6 W	4,7 W   0,42 W	8 W   < 0,9 W	26 W   < 2,5 W
Flash (SATA SSD)	8 GB	4 GB	4GB (LX) 32GB (W10)	4GB (LX) 32GB (W10)	4 GB

# Advantages and Drawbacks of Thin Clients over Desktops

## • Advantages of Thin Clients

- Low acquisition costs (approx € 500)
- Reduced power consumption (a few watts)  $\implies$  reduced operating costs
- Reduced footprint (little space consumption)
- Reduced noise, because no hard drive and sometimes fanless
- Central storage of data is more efficient and more secure
- Reduced resource consumption because of virtualization on the server
- Reduced effort (cost) for administration

## • Drawbacks of Thin Clients

- No 3D graphics performance
- Limited extensibility
- Users fear storing their data outside of their PC (outside of their own sphere of influence)
- Server is a single point of failure and eventually a bottleneck



## Different Client-Server Scenarios (1/2)

- Company X runs 500 computer workplaces
- Calculate the electricity costs per year (including the leap year) for 24/7 operation when the electricity price is 0,35 €/kWh.
- **Scenario 1: Fat clients (PC)**
  - Electrical power rating per PC: 450 watts
  - Electrical power rating per screen: 80 watts
- Electricity costs per year for 500 PCs with screens:

$$0.53 \text{ kW} * 24 \frac{\text{h}}{\text{Day}} * 365.25 \frac{\text{Day}}{\text{Year}} * 0.35 \frac{\text{€}}{\text{kWh}} * 500 = \mathbf{813,046.5} \frac{\text{€}}{\text{Year}}$$

## Different Client-Server Scenarios (1/2)

### ● Scenario 2: Thin clients (PC)

- Electrical power rating per thin client: 30 watts
- Electrical power rating per screen: 80 watts
- Electrical power rating per server blade: 600 watts
- Each server blade has enough resources to interact with 30 thin clients

*Electricity costs per year (including the leap year) for 500 thin clients with screens:*

$$0.11 \text{ kW} * 24 \frac{\text{h}}{\text{Day}} * 365.25 \frac{\text{Day}}{\text{Year}} * 0.35 \frac{\text{€}}{\text{kWh}} * 500 = 168,745.5 \frac{\text{€}}{\text{Year}}$$

*17 server blades are required to run the 500 computer workplaces.*

*Electricity costs per year (including the leap year) for 17 server blades:*

$$0.6 \text{ kW} * 24 \frac{\text{h}}{\text{Day}} * 365.25 \frac{\text{Day}}{\text{Year}} * 0.35 \frac{\text{€}}{\text{kWh}} * 17 \approx 31,294.62 \frac{\text{€}}{\text{Year}}$$

*Electricity costs per year for the thin clients, screens and server blades:*

$$168,745.5 \frac{\text{€}}{\text{Year}} + 31,294.62 \frac{\text{€}}{\text{Year}} \approx \mathbf{200,040.12 \frac{\text{€}}{\text{Year}}}$$

# Summary about the Clients

Image Source: Google

- The era of **X-Terminals** and **Text-Terminals** is over
- **Applet Clients** did fail in the 1990s but their popularity may grow in the next years  
⇒ Google Chrome OS



- **Fat Clients** are standard today
- **Thin/Zero Clients** are rarely used today
  - Things change slowly in the industry
  - Thin Clients are a hot topic again because of rising energy costs
  - Keyword: Green IT



# Analysis of the 3 Options

- 1 Optimization of the algorithms used
  - Algorithms cannot be optimized infinitely
- 2 Increased compute performance with faster CPUs
  - The compute power of a computer cannot be increased infinitely
    - Symmetric multiprocessing (SMP) has limitations
    - The memory bus becomes a bottleneck  $\implies$  Von Neumann bottleneck (see slide 35)
    - Each additional CPU decreases the relative performance gain

Reason: The storage subsystems cannot deliver the data fast enough to fully utilize all available CPUs

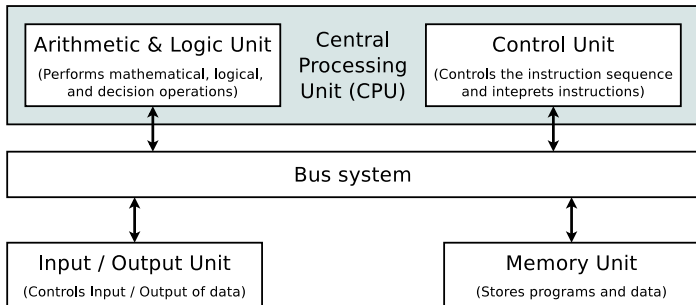
- 3 **Using more than just a single computer system to increase the performance**
  - The possible performance enhancement is potentially unlimited and it is only limited by these factors:
    - Performance of the nodes
    - Transfer rate of the network technology used
    - Maintenance and administration effort for the connected systems





# Von Neumann Bottleneck (1/2)

- The data and control bus is increasingly becoming a bottleneck between the CPU and memory
  - The main memory and the bus system are key factors for the performance of a computer



- The Von Neumann Architecture describes the structure of the general-purpose computer, which is not limited to a fixed program and has input and output devices
- Main difference to modern systems: A single Bus to connect I/O devices directly with the CPU, is impossible today



# Amdahl's Law

Image source: [archive.computerhistory.org](http://archive.computerhistory.org)

- Published in 1967
- Named after Gene Myron Amdahl
- Calculates the maximum expected acceleration of programs by parallel execution on multiple CPUs
- According to Amdahl, the performance gain is limited mainly by the sequential part of the problem
- A program can never be fully executed in parallel
  - Program components such as process initialization and memory allocation only run once on a single CPU
    - ⇒ Those parts of the program cannot be executed in parallel
  - Some parts of the program depend on the sequence of events, input-output and of intermediate results



# Amdahl's Law – Principle (1/3)

Source: [https://en.wikipedia.org/wiki/Amdahls\\_law](https://en.wikipedia.org/wiki/Amdahls_law)

- The sequential and parallel executable parts of the program are identified
- $P$  is the parallel portion and  $(1 - P)$  is the sequential portion
- Total runtime of the program:

$$1 = (1 - P) + P$$

- Example: A program requires 20 hours CPU time with a single CPU
  - For a single hour, the process runs sequentially
  - The remaining 19 hours are 95% of the total effort and can be distributed to any number of CPUs
  - But the total computation time can never fall under a single hour
    - Not even with an infinite number of CPUs
  - Therefore, the maximum acceleration (SpeedUp) in theory is factor 20

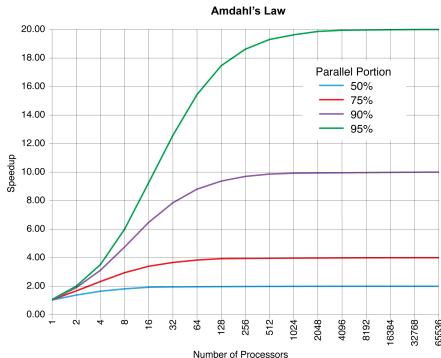
# Amdahl's Law – Principle (2/3)

Image source: Wikipedia

- $N$  = number of CPUs
- $(1 - P)$  = sequential portion
- $(P/N)$  = accelerated parallel portion
- $S$  = SpeedUp (acceleration)

$$S = \frac{1}{(1 - P) + \frac{P}{N}} \leq \frac{1}{(1 - P)}$$

- With an rising number of CPUs, the acceleration depends more and more of the sequential part
- The graph does converge to  $1/(1 - P)$



Bad news:  
This is too much optimistic.  
In practice, things are much worse!

## Amdahl's Law – Principle (3/3)

- The load caused by communication and synchronization rises with a growing number of CPUs
  - For this reason, the inequality is extended by factor  $o(N)$ , which grows when  $N$  grows

$$S = \frac{1}{(1 - P) + o(N) + \frac{P}{N}} \leq \frac{1}{(1 - P)}$$

- Because of  $o(N)$ , the graph does not converge to  $1/(1 - P)$  any longer
  - The graph reaches a maximum and then declines when additional CPUs are used  
⇒ see slides 46, 50 and 53

## Amdahl's Law – Issues

- Amdahl's law does not take into account the **cache** and the effects, which are caused by the cache in practice
  - A growing number of CPUs also increases the quantity of fast memory which is available
- In the optimal case, the entire data of the problem can be stored in the cache, which is a faster than the main memory
  - In such a case (very rare!), a super-linear SpeedUp may occur, which leads to an acceleration which exceeds the additional compute power

$$S_{(p)} = \frac{t_{(s)}}{t_{(p)}}$$

$S_{(p)}$  = Speedup Factor when using  $p$  CPU cores of a multiprocessor system

$t_{(s)}$  = Execution time by using a single CPU core

$t_{(p)}$  = Execution time by using  $p$  CPU cores

- The max. SpeedUp is usually  $p$  with  $p$  CPU cores ( $\implies$  linear SpeedUp)
  - A super-linear SpeedUp is greater than  $p$
- 
- The problems to be addressed in distributed computing today are very big and the sequential part of these problems is very small

# Granularity

- A daily life version of Amdahl's law is the wallpaper example
  - A painter needs 1 hour for wallpapering a room
  - Realistic: 2 painters wallpaper the room in 30 minutes
  - Unrealistic: 60 painters wallpaper the room in 1 minute
    - Reason: The painters are standing in each others way
    - There are disputes caused by limited resources (table, ladder...)
    - Probably with 60 painters, it would take more than 30 minutes
  - With 60 painters, a room cannot be wallpapered 60 times as fast
    - But this works for a hotel with 60 rooms, when the painters are distributed to the rooms
- Transferred to parallel computers, this means that **with a growing number of CPUs, the problem size should grow too**
- **The problem needs to *scale* with the number of CPUs**
- This finding initiated the development of Gustafson's law in 1988

# Gustafson's Law

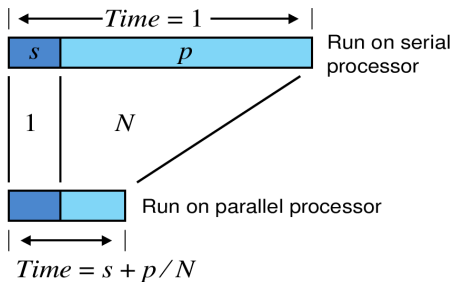
Image source: [siliconsemiconductor.net](http://siliconsemiconductor.net)

- Amdahl's law considered mainly small problems
  - But: the bigger a parallelizable problem is, the smaller is the portion of the sequential part
- Gustafson's Law from John Gustafson (1988) says that **a problem, which is sufficiently large, can be parallelized efficiently**
- Difference to Amdahl's law:
  - The parallel portion of the problem grows with the number of CPUs
  - The sequential part is not limiting, because it gets more and more unimportant as the number of CPUs rises



Gustafson, Montry, Benner. *Development of Parallel Methods For a 1024-Processor Hypercube*. Sandia National Laboratories. 1988

# Gustafson's Law: Speedup



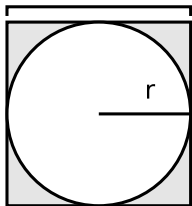
$$SpeedUp = \frac{1}{s + \frac{p}{N}}$$

- If the number of CPUs grows to infinity, the SpeedUp grows linear with the number of CPUs
- Big problems, where the SpeedUp is nearly equivalent to the number of CPUs, exist among others in hydrodynamics, structural engineering and meteorology

Source: <http://www.johngustafson.net/pubs/pub13/amdahl.pdf>

# Example: Calculation of $\pi$ via Monte Carlo Simulation

$2r$



$r$  = Radius  
 $A$  = Surface ratio  
 $C$  = Circle  
 $S$  = Square

- Inscribe a circle of radius  $r$  inside a square with side length  $2r$
- Generate random dots in the square
  - The number of dots in  $A_C$  in relation to the number of dots in  $A_S$  is equal to the surface ratio

$$\frac{A_C}{A_S} = \frac{\pi \cdot r^2}{(2 \cdot r)^2} = \frac{\pi \cdot r^2}{4 \cdot r^2} = \frac{\pi}{4}$$

- The dots can be generated (X/Y axis values via random) in parallel by the workers
- The master receives from each worker the number of calculated dots in  $A_C$  and calculates:

$$\frac{4 \cdot \text{dots in } A_C}{\text{dots in } A_S} = \pi$$

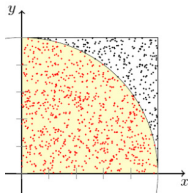
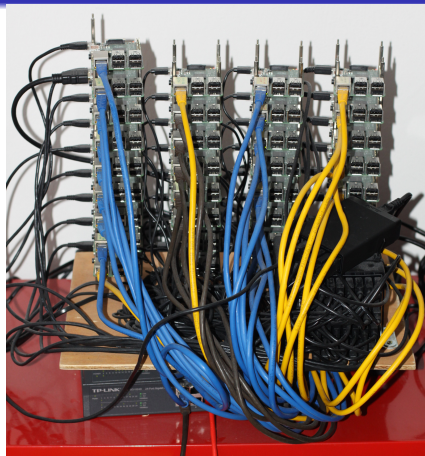


Image source: Wikipedia

# $\pi$ Approximation with 32 Raspberry Pi 2 and MPI



**Performance and Energy-Efficiency Aspects of Clusters of Single Board Computers.** *Christian Baun.* International Journal of Distributed and Parallel Systems (IJDPS), Vol.7, No.2/3/4, 2016, S.13-22.  
<http://aircconline.com/ijdps/V7N4/74161ijdps02.pdf>

- 33 Raspberry Pi 2 (900 MHz)
  - 32 worker nodes and 1 master
  - 128 CPU cores
  - 24.7 Gflops
  - 1 GB main memory per node
  - 100 MBit/s Ethernet

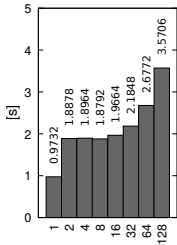
This computation power is similar to an Intel Core 2 Quad Q9450 2.66 Ghz from 2008, which has approx. 25.6 Gflops

Source: <https://www.tecchannel.de/a/test-intel-core-i7-mit-nehalem-quad-core,1775602>

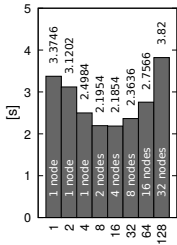
- **Do you think the problem size has a strong impact on the scalability?**

# Can you see Amdahl's Law and Gustafson's Law?

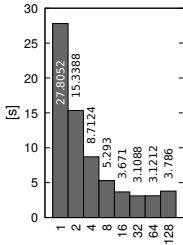
Pi approximated with 1,000,000 points  
(Mean Time of 5 Tests)



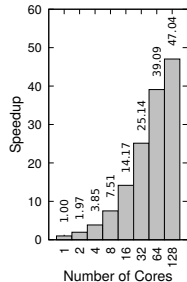
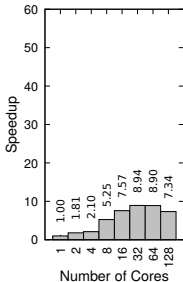
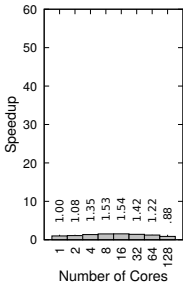
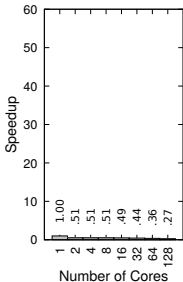
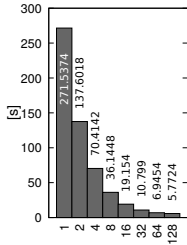
Pi approximated with 10,000,000 points  
(Mean Time of 5 Tests)



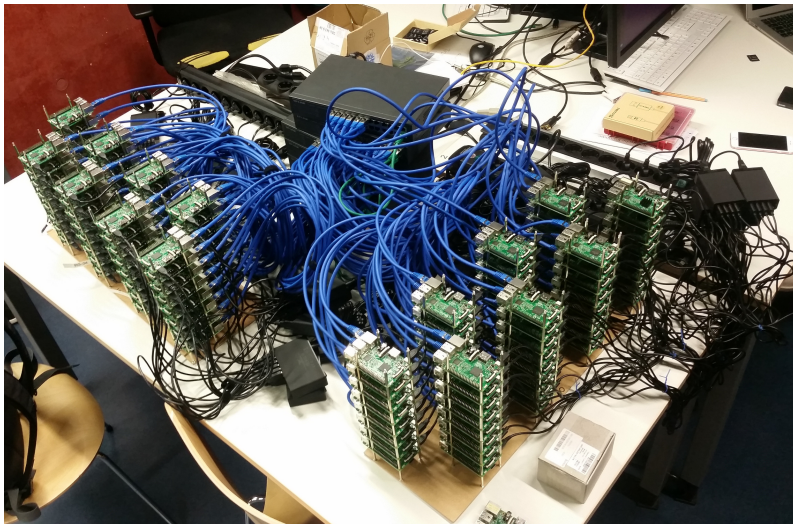
Pi approximated with 100,000,000 points  
(Mean Time of 5 Tests)



Pi approximated with 1,000,000,000 points  
(Mean Time of 5 Tests)



# Our Cluster with 128 RPi 3 with 512 CPU Cores (until 2/2019)

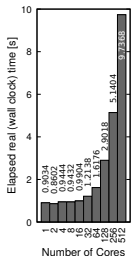


# It was not a Beauty but it worked well. . .

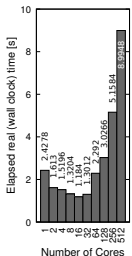


# With 512 CPU cores the Results get more interesting

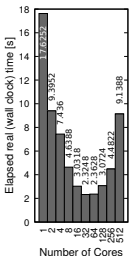
Pi approximated with 1,000,000 points  
(Mean Time of 5 Tests)



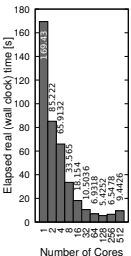
Pi approximated with 10,000,000 points  
(Mean Time of 5 Tests)



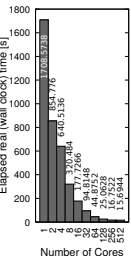
Pi approximated with 100,000,000 points  
(Mean Time of 5 Tests)



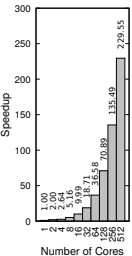
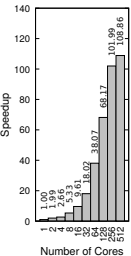
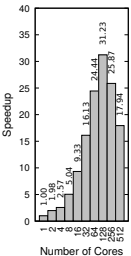
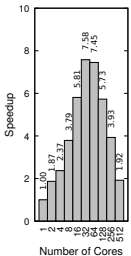
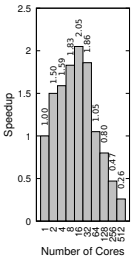
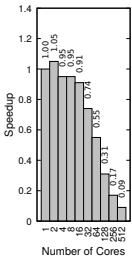
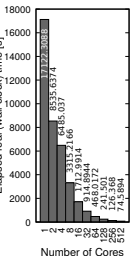
Pi approximated with 1,000,000,000 points  
(Mean Time of 5 Tests)



Pi approximated with 10,000,000,000 points  
(Mean Time of 5 Tests)



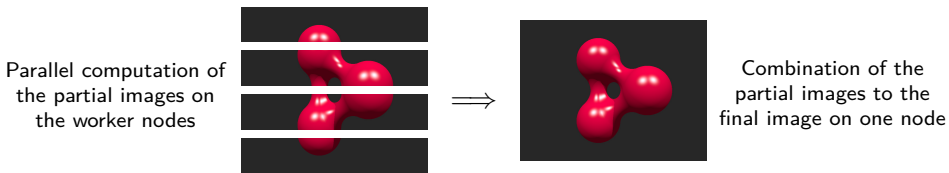
Pi approximated with 100,000,000,000 points  
(Mean Time of 5 Tests)



## Another Example: task-distributor + POV-Ray

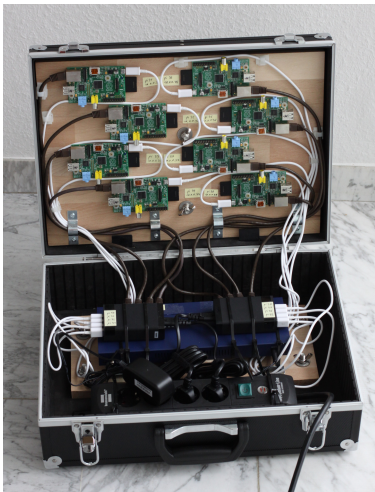
- Ray tracing is an interesting topic for parallel systems
- POV-Ray is a free, stable and feature-rich ray tracing solution  
<http://www.povray.org>
- Problem in 2015: no working (maintained) POV-Ray solution for parallel image computation in clusters existed
- Solution: **task-distributor**

<http://github.com/christianbaun/task-distributor>



Parallel image computation in clusters with task-distributor. *Christian Baun*. SpringerPlus 2016 5:632.  
<http://springerplus.springeropen.com/articles/10.1186/s40064-016-2254-x>

# Clusters used in 2015



- Clusters with 8 nodes (RPI 1) each
- One single core CPU per node
- 512 MB main memory per node

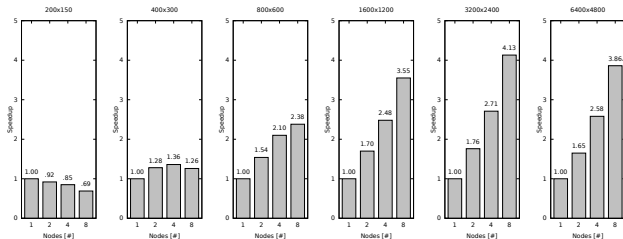
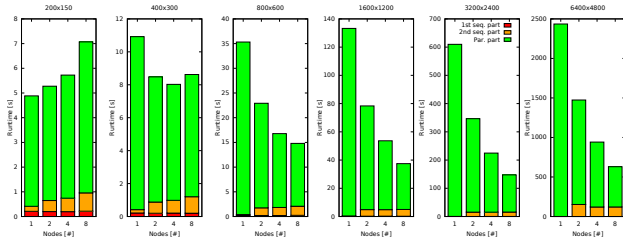


**Mobile clusters of single board computers: an option for providing resources to student projects and researchers.** *Christian Baun.* SpringerPlus 2016 5:360.

<http://springerplus.springeropen.com/articles/10.1186/s40064-016-1981-3>

# Some Observations with task-distributor + POV-Ray

● We see: Laws and challenges of distributed systems



- **Amdahl's law:** The performance gain is limited mainly by the sequential part of the problem
- **Gustafson's law:** A sufficiently large problem can be parallelized efficiently
- The sequential part gets more and more unimportant as the number of CPUs rises
- **Swap** with 6400x4800 (convert consumes approx. 500 MB RAM for putting together the partial images. But we had just 512 MB - 16 MB for the GPU - Space for Linux)

# Magnitudes of Data

- Magnitudes and units
- The size of storage is measured in Bytes

Kilobyte	(KB)	$10^3$	= 1,000 Bytes	$2^{10}$	= 1,024 Bytes
Megabyte	(MB)	$10^6$	= 1,000,000 Bytes	$2^{20}$	= 1,048,576 Bytes
Gigabyte	(GB)	$10^9$	= 1,000,000,000 Bytes	$2^{30}$	= 1,073,741,824 Bytes
Terabyte	(TB)	$10^{12}$	= 1,000,000,000,000 Bytes	$2^{40}$	= 1,099,511,627,776 Bytes
Petabyte	(PB)	$10^{15}$	= 1,000,000,000,000,000 Bytes	$2^{50}$	= 1,125,899,906,842,624 Bytes
Exabyte	(EB)	$10^{18}$	= 1,000,000,000,000,000,000 Byte	$2^{60}$	= 1,152,921,504,606,846,976 Bytes
Zettabyte	(ZB)	$10^{21}$	= 1,000,000,000,000,000,000,000 Byte	$2^{70}$	= 1,180,591,620,717,411,303,424 Bytes

## Bill Gates (1981)

„640 Kilobyte ought to be enough for anybody.“

- Common assumptions about data:
  - It is easy to store data today
  - It is easy to transmit and transport data today
- Are these assumptions correct?  $\implies$  **exercise sheet 1**

## Two Points of View

### Seymour Cray (1925 - 1996)

„If you were plowing a field, what would you rather use? Two strong oxen or 1024 chickens?“

### W. Gropp, E. Lusk, A. Skjellum. *Using MPI*. The MIT Press (1996)

„To pull a bigger wagon, it is easier to add more oxen than to grow a gigantic ox.“

- What does this mean?

# Seymour Cray (1925 – 1996)



- Seymour Cray founded Cray Research in 1972, the first successful company for the development and sale of supercomputers
- Cray was an opponent of the multiprocessing
- Crays supercomputers had few, but very powerful CPUs

„Anyone can build a fast CPU. The trick is to build a fast system.“

Image source: <http://www.cray-cyber.org/memory/scray.php>

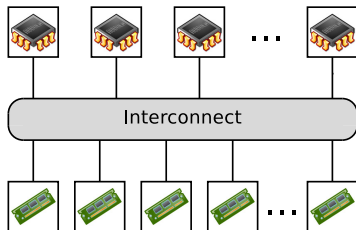
# Background

- Until the 1990s, the acceleration of single CPU had much potential to increase the compute power
  - Today, it is hardly possible to accelerate individual CPU cores, without causing an increase in the required electric power input, which causes to additional waste heat
- Ultimate limits will prevent Moore's law to be valid forever
  - Someday, the traditional way to improve the performance of CPUs (increasing the packing density and clock frequency) will not work any longer
- In the last years, increasing the CPU performance was achieved almost exclusively by increasing the number of CPU cores
- At the time of Seymour Cray, powerful computers were expensive
- Since several years, the CPUs of inexpensive desktop systems are almost as powerful as CPUs in supercomputers



# Shared Memory

- For systems with **shared memory**, the entire memory is part of a uniform address space, which is accessed by all CPUs
- The memory is accessed via an interconnect



- Problem: Write operations of the CPUs must be coordinated
- Further problem: Data inside the CPU caches
  - If a memory cell duplicated in multiple CPU caches, any change in the memory cell must be propagated to all caches

# Symmetric and Asymmetric Multiprocessing

- Most multiprocessor systems today operate according to the **symmetric multiprocessing** (SMP) principle
  - SMP allows to dynamically distribute the running processes to all available CPUs
  - All CPUs can access the memory with the same speed
- In multiprocessor systems, which operates according to the **asymmetric multiprocessing** principle, each CPU must be assigned to a fixed task
  - One or more CPUs run the operating system
  - The other processes are distributed to the remaining CPUs
  - Typically, the CPUs are identical
  - Today, it exists often a main CPU and some subordinated CPUs, which are focused to specific tasks









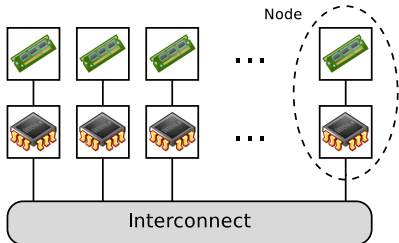
# BladeCenter for the Blades (IBM HS20)



Image source: <http://www.flickr.com/photos/jemimus/74452762/>

# Distributed Memory $\implies$ Cluster Systems (see Slide Set 2)

- Each CPU can only access its own local memory
- The communication between the CPUs takes place via a network connection
  - Network connections are much slower, compared with the data rate between CPU and memory
- In a parallel computer, every single CPU and its local memory, are independent nodes



- A system with distributed memory is also called **Cluster** or *Multicomputer*, because each node is an independent computer with a Von Neumann architecture
- Nodes of the cluster can also be SMP systems. . .