Multiprocessing

Contents

Chapter 1

Multiprocessing

Multiprocessing is the use of two or more central processing units (CPUs) within a single computer system.[1][2] The term also refers to the ability of a system to support more than one processor and/or the ability to allocate tasks between them.[3] There are many variations on this basic theme, and the definition of multiprocessing can vary with context, mostly as a function of how CPUs are defined (multiple cores on one die, multiple dies in one package, multiple packages in one system unit, etc.).

According to some on-line dictionaries, a **multiprocessor** is a computer system having two or more processing units (multiple processors) each sharing main memory and peripherals, in order to simultaneously process programs.[4][5] A 2009 textbook defined multiprocessor system similarly, but noting that the processors may share “some or all of the system’s memory and I/O facilities"; it also gave **tightly coupled system** as a synonymous term.[6]

At the operating system level, *multiprocessing* is sometimes used to refer to the execution of multiple concurrent processes in a system as opposed to a single process at any one instant.[7][8] When used with this definition, multiprocessing is sometimes contrasted with multitasking, which may use just a single processor but switch it in time slices between tasks (i.e. a time-sharing system). Multiprocessing however means true parallel execution of multiple processes using more than one processor.[8] Multiprocessing doesn't necessarily mean that a single process or task uses more than one processor simultaneously; the term parallel processing is generally used to denote that scenario.[7] Other authors prefer to refer to the operating system techniques as multiprogramming and reserve the term *multiprocessing* for the hardware aspect of having more than one processor.[2][9] The remainder of this article discusses multiprocessing only in this hardware sense.

In Flynn’s taxonomy, multiprocessors as defined above are MIMD machines.[10][11] As they are normally construed to be tightly coupled (share memory), multiprocessors are not the entire class of MIMD machines, which also contains message passing multicomputer systems.[10]

1.1 Pre-history

According to a 1985 article in *Byte*, possibly the first expression of the idea of multiprocessing is found in the 1842 words of Federico Luigi Menabrea, which said about Charles Babbage's analytical engine: “The machine can be brought into play so as to give several results at the same time, which will greatly abridge the whole amount of processes.”[12]

1.2 Key topics

1.2.1 Processor symmetry

In a **multiprocessing** system, all CPUs may be equal, or some may be reserved for special purposes. A combination of hardware and operating system software design considerations determine the symmetry (or lack thereof) in a given system. For example, hardware or software considerations may require that only one particular CPU respond to all hardware interrupts, whereas all other work in the system may be distributed equally among CPUs; or execution of kernel-mode code may be restricted to only one particular CPU, whereas user-mode code may be executed in any combination of processors. Multiprocessing systems are often easier to design if such restrictions are imposed, but they tend to be less efficient than systems in which all CPUs are utilized.

Systems that treat all CPUs equally are called symmetric multiprocessing (SMP) systems. In systems where all CPUs are not equal, system resources may be divided in a number of ways, including asymmetric multiprocessing (ASMP), non-uniform memory access (NUMA) multiprocessing, and clustered multiprocessing.

1.2.2 Instruction and data streams

In multiprocessing, the processors can be used to execute a single sequence of instructions in multiple contexts (single-instruction, multiple-data or SIMD, often used in vector processing), multiple sequences of instructions in a single context (multiple-instruction, single-data or MISD, used for redundancy in fail-safe systems and sometimes applied to describe pipelined processors or hyper-threading), or multiple sequences of instructions in multiple contexts (multiple-instruction, multiple-data or MIMD).

1.2.3 Processor coupling

Tightly coupled multiprocessor system

Tightly coupled multiprocessor systems contain multiple CPUs that are connected at the bus level. These CPUs may have access to a central shared memory (SMP or UMA), or may participate in a memory hierarchy with both local and shared memory (NUMA). The IBM p690 Regatta is an example of a high end SMP system. Intel Xeon processors dominated the multiprocessor market for business PCs and were the only major x86 option until the release of AMD's Opteron range of processors in 2004. Both ranges of processors had their own onboard cache but provided access to shared memory; the Xeon processors via a common pipe and the Opteron processors via independent pathways to the system RAM.

Chip multiprocessors, also known as multi-core computing, involves more than one processor placed on a single chip and can be thought of the most extreme form of tightly-coupled multiprocessing. Mainframe systems with multiple processors are often tightly-coupled.

Loosely coupled multiprocessor system

Main article: shared nothing architecture

Loosely coupled multiprocessor systems (often referred to as clusters) are based on multiple standalone single or dual processor commodity computers interconnected via a high speed communication system (Gigabit Ethernet is common). A Linux Beowulf cluster is an example of a loosely-coupled system.

Tightly-coupled systems perform better and are physically smaller than loosely-coupled systems, but have historically required greater initial investments and may depreciate rapidly; nodes in a loosely coupled system are usually inexpensive commodity computers and can be recycled as independent machines upon retirement from the cluster.

Power consumption is also a consideration. Tightly coupled systems tend to be much more energy efficient than clusters. This is because considerable economy can be realized by designing components to work together from the beginning in tightly coupled systems, whereas loosely coupled systems use components that were not necessarily intended specifically for use in such systems.

Loosely-coupled systems have the ability to run different operating systems or OS versions on different systems.

1.2.4 Multiprocessor Communication Architecture

Message passing

Separate address space for each processor.

processors communicate via message passing.

processors have provide memories.

focus attention on costly non-local operations.

Shared memory

Processors communicate with shared address space

Processors communicate by memory read/write

Easy on small-scale machines

Lower latency

SMP or NUMA architecture

1.3 Flynn’s taxonomy

1.3.1 SISD multiprocessing

Main article: SISD

In a single-instruction stream, single-data stream computer one processor sequentially processes instructions, each instruction processes one data item. One example is the “von Neumann” architecture with RISC.

1.3.2 SIMD multiprocessing

Main article: SIMD

In a single-instruction stream, multiple data stream computer one processor handles a stream of instructions, each one of which can perform calculations in parallel on multiple data locations.

SIMD multiprocessing is well suited to parallel or vector processing, in which a very large set of data can be divided into parts that are individually subjected to identical but independent operations. A single instruction stream directs the operation of multiple processing units to perform the same manipulations simultaneously on potentially large amounts of data.

For certain types of computing applications, this type of architecture can produce enormous increases in performance, in terms of the elapsed time required to complete a given task. However, a drawback to this architecture is that a large part of the system falls idle when programs or system tasks are executed that cannot be divided into units that can be processed in parallel.

Additionally, programs must be carefully and specially written to take maximum advantage of the architecture, and often special optimizing compilers designed to produce code specifically for this environment must be used. Some compilers in this category provide special constructs or extensions to allow programmers to directly specify operations to be performed in parallel (*e.g.,* DO FOR ALL statements in the version of FORTRAN used on the ILLIAC IV, which was a SIMD multiprocessing supercomputer).

SIMD multiprocessing finds wide use in certain domains such as computer simulation, but is of little use in general-purpose desktop and business computing environments.

1.3.3 MISD multiprocessing

Main article: MISD

MISD multiprocessing offers mainly the advantage of redundancy, since multiple processing units perform the same tasks on the same data, reducing the chances of incorrect results if one of the units fails. MISD architectures may involve comparisons between processing units to detect failures. Apart from the redundant and fail-safe character of this type of multiprocessing, it has few advantages, and it is very expensive. It does not improve performance. It can be implemented in a way that is transparent to software. It is used in array processors and is implemented in fault tolerant machines.

Another example of MISD is pipelined image processing where every image pixel is piped through several hardware units performing several steps of image transformation.

1.3.4 MIMD multiprocessing

Main article: MIMD

MIMD multiprocessing architecture is suitable for a wide variety of tasks in which completely independent and parallel execution of instructions touching different sets of data can be put to productive use. For this reason, and because it is easy to implement, MIMD predominates in multiprocessing.

Processing is divided into multiple threads, each with its own hardware processor state, within a single software-defined process or within multiple processes. Insofar as a system has multiple threads awaiting dispatch (either system or user threads), this architecture makes good use of hardware resources.

MIMD does raise issues of deadlock and resource contention, however, since threads may collide in their access to resources in an unpredictable way that is difficult to manage efficiently. MIMD requires special coding in the operating system of a computer but does not require application changes unless the programs themselves use multiple threads (MIMD is transparent to single-threaded programs under most operating systems, if the programs do not voluntarily relinquish control to the OS). Both system and user software may need to use software constructs such as *semaphores* (also called *locks* or *gates*) to prevent one thread from interfering with another if they should happen to cross paths in referencing the same data. This gating or locking process increases code complexity, lowers performance, and greatly increases the amount of testing required, although not usually enough to negate the advantages of multiprocessing.

Similar conflicts can arise at the hardware level between processors (cache contention and corruption, for example), and must usually be resolved in hardware, or with a combination of software and hardware (*e.g.,* cache-clear instructions).

1.4 See also

- Symmetric multiprocessing
- Asymmetric multiprocessing
- Multi-core processor
- BMDFM – Binary Modular Dataflow Machine, a SMP MIMD runtime environment
- Software lockout
- OpenHMPP

1.5 References

[1] Raj Rajagopal (1999). *Introduction to Microsoft Windows NT Cluster Server: Programming and Administration*. CRC Press. p. 4. ISBN 978-1-4200-7548-9.

[2] Mike Ebbers; John Kettner; Wayne O'Brien; Bill Ogden, IBM Redbooks (2012). *Introduction to the New Mainframe: z/OS Basics*. IBM Redbooks. p. 96. ISBN 978-0-7384-3534-3.

[3] Chip multiprocessing

[4] http://www.yourdictionary.com/multiprocessor

[5] http://www.thefreedictionary.com/multiprocessor

[6] Irv Englander (2009). *The architecture of Computer Hardware and Systems Software. An Information Technology Approach.* (4th ed.). Wiley. p. 265.

[7] Deborah Morley; Charles Parker (13 February 2012). *Understanding Computers: Today and Tomorrow, Comprehensive*. Cengage Learning. p. 183. ISBN 1-133-19024-3.

[8] Shibu K. V. *Introduction to Embedded Systems*. Tata McGraw-Hill Education. p. 402. ISBN 978-0-07-014589-4.

[9] Ashok Arora (2006). *Foundations of Computer Science*. Laxmi Publications. p. 149. ISBN 978-81-7008-971-1.

[10] Ran Giladi (2008). *Network Processors: Architecture, Programming, and Implementation*. Morgan Kaufmann. p. 293. ISBN 978-0-08-091959-1.

[11] Sajjan G. Shiva (20 September 2005). *Advanced Computer Architectures*. CRC Press. p. 221. ISBN 978-0-8493-3758-1.

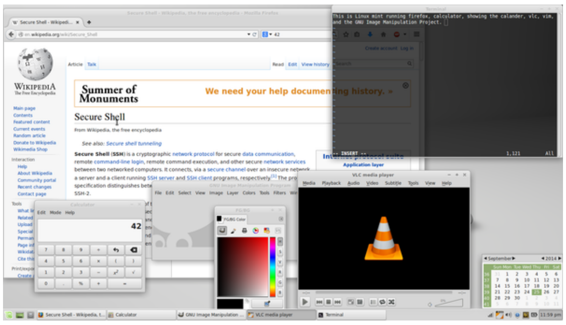
[12] Multiprocessing, BYTE magazine Volume 10, Number 05 (May 1985), p. 169

Chapter 2

Computer multitasking

For other uses, see Multitasking (disambiguation).

In computing, **multitasking** is a method where multiple tasks (also known as processes) are performed during the same period of time – they are executed concurrently (in overlapping time periods, new tasks starting before others have ended) instead of sequentially (one completing before the next starts). The tasks share common processing resources, such as central processing units (CPUs) and main memory.



```
This is Linux mint running firefox, calculator, showing the calander, vlc, vim,
and the GNU Image Manipulation Project.
-- INSERT --                                                    1,121        All
```

Secure Shell - Wikipedia, the free encyclopedia - Mozilla Firefox
en.wikipedia.org/wiki/Secure_Shell
Wikipedia
The Free Encyclopedia
Summer of Monuments
We need your help documenting history. »
Secure Shell
Calculator
42
GNU Image Manipulation Program
VLC media player
Terminal
September
2014
11:59 pm

Modern desktop operating systems are capable of handling large numbers of different processes at the same time. This screenshot shows Linux Mint running simultaneously Xfce desktop environment, Firefox, a calculator program, the built-in calendar, Vim, GIMP, and VLC media player.

Multitasking does *not* necessarily mean that multiple tasks are executing at exactly the same instant. In other words, multitasking does not imply parallel execution, but it does mean that more than one task can be part-way through execution at the same time, and that more than one task is advancing over a given period of time.

In the case of a computer with a single CPU, only one task is said to be running at any point in time, meaning that the CPU is actively executing instructions for that task. Multitasking solves the problem by scheduling which task may be the one running at any given time, and when another waiting task gets a turn. The act of reassigning a CPU from one task to another one is called a context switch. When context switches occur frequently enough, the illusion of parallelism is achieved.

Even on multiprocessor or multicore computers, which have multiple CPUs/cores so more than one task can be executed at once (physically, one per CPU or core), multitasking allows many more tasks to be run than there are CPUs. The term *multitasking* has become an international term, as the same word is used in many other languages such as German, Italian, Dutch, Danish and Norwegian.

Operating systems may adopt one of many different scheduling strategies, which generally fall into the following categories:

- In multiprogramming systems, the running task keeps running until it performs an operation that requires waiting for an external event (e.g. reading from a tape) or until the computer's scheduler forcibly swaps the running task out of the CPU. Multiprogramming systems are designed to maximize CPU usage.
- In time-sharing systems, the running task is required to relinquish the CPU, either voluntarily or by an external event such as a hardware interrupt. Time sharing systems are designed to allow several programs to execute apparently simultaneously.
- In real-time systems, some waiting tasks are guaranteed to be given the CPU when an external event occurs. Real time systems are designed to control mechanical devices such as industrial robots, which require timely processing.

2.1 Multiprogramming

In the early days of computing, CPU time was expensive, and peripherals were very slow. When the computer ran a program that needed access to a peripheral, the central processing unit (CPU) would have to stop executing program instructions while the peripheral processed the data. This was deemed very inefficient.

The first computer using a multiprogramming system was the British *Leo III* owned by J. Lyons and Co. Several different programs in batch were loaded in the computer memory, and the first one began to run. When the first program reached an instruction waiting for a peripheral, the context of this program was stored away, and the second program in memory was given a chance to run. The process continued until all programs finished running.

The use of multiprogramming was enhanced by the arrival of virtual memory and virtual machine technology, which enabled individual programs to make use of memory and operating system resources as if other concurrently running programs were, for all practical purposes, non-existent and invisible to them.

Multiprogramming doesn't give any guarantee that a program will run in a timely manner. Indeed, the very first program may very well run for hours without needing access to a peripheral. As there were no users waiting at an interactive terminal, this was no problem: users handed in a deck of punched cards to an operator, and came back a few hours later for printed results. Multiprogramming greatly reduced wait times when multiple batches were being processed.

2.2 Cooperative multitasking

See also: Nonpreemptive multitasking

The expression “time sharing” usually designated computers shared by interactive users at terminals, such as IBM’s TSO, and VM/CMS. The term “time-sharing” is no longer commonly used, having been replaced by “multitasking”, following the advent of personal computers and workstations rather than shared interactive systems.

Early multitasking systems used applications that voluntarily ceded time to one another. This approach, which was eventually supported by many computer operating systems, is known today as cooperative multitasking. Although it is now rarely used in larger systems except for specific applications such as CICS or the JES2 subsystem, cooperative multitasking was once the scheduling scheme employed by Microsoft Windows (prior to Windows 95 and Windows NT) and Mac OS (prior to OS X) in order to enable multiple applications to be run simultaneously. Windows 9x also used cooperative multitasking, but only for 16-bit legacy applications, much the same way as pre-Leopard PowerPC versions of Mac OS X used it for Classic applications. The network operating system NetWare used cooperative multitasking up to NetWare 6.5. Cooperative multitasking is still used today on RISC OS systems.[1]

As a cooperatively multitasked system relies on each process regularly giving up time to other processes on the system, one poorly designed program can consume all of the CPU time for itself, either by performing extensive calculations or by busy waiting; both would cause the whole system to hang. In a server environment, this is a hazard that makes the entire environment unacceptably fragile.

2.3 Preemptive multitasking

Main article: Preemption (computing)

Preemptive multitasking allows the computer system to guarantee more reliably each process a regular “slice” of operating time. It also allows the system to deal rapidly with important external events like incoming data, which might require the immediate attention of one or another process. Operating systems were developed to take advantage of these hardware capabilities and run multiple processes preemptively. Preemptive multitasking was supported on DEC’s PDP-8 computers, and implemented in OS/360 MFT in 1967, in MULTICS (1964), and Unix (1969); it is a core feature of all Unix-like operating systems, such as Linux, Solaris and BSD with its derivatives.[2]

At any specific time, processes can be grouped into two categories: those that are waiting for input or output (called "I/O bound"), and those that are fully utilizing the CPU ("CPU bound"). In primitive systems, the software would often "poll", or "busywait" while waiting for requested input (such as disk, keyboard or network input). During this time, the system was not performing useful work. With the advent of interrupts and preemptive multitasking, I/O bound processes could be “blocked”, or put on hold, pending the arrival of the necessary data, allowing other processes to utilize the CPU. As the arrival of the requested data would generate an interrupt, blocked processes could be guaranteed a timely return to execution.

The earliest preemptive multitasking OS available to home users was Sinclair QDOS on the Sinclair QL, released in 1984, but very few people bought the machine. Commodore’s powerful Amiga, released the following year, was the first commercially successful home computer to use the technology, and its multimedia abilities make it a clear ancestor of contemporary multitasking personal computers. Microsoft made preemptive multitasking a core feature of their flagship operating system in the early 1990s when developing Windows NT 3.1 and then Windows 95. It was later adopted on the Apple Macintosh by Mac OS X that, as a Unix-like operating system, uses preemptive multitasking for all native applications.

A similar model is used in Windows 9x and the Windows NT family, where native 32-bit applications are multitasked preemptively, and legacy 16-bit Windows 3.x programs are multitasked cooperatively within a single process, although in the NT family it is possible to force a 16-bit application to run as a separate preemptively multitasked process.[3] 64-bit editions of Windows, both for the x86-64 and Itanium architectures, no longer provide support for legacy 16-bit applications, and thus provide preemptive multitasking for all supported applications.

2.4 Real time

Another reason for multitasking was in the design of real-time computing systems, where there are a number of possibly unrelated external activities needed to be controlled by a single processor system. In such systems a hierarchical interrupt system is coupled with process prioritization to ensure that key activities were given a greater share of available process time.

2.5 Multithreading

As multitasking greatly improved the throughput of computers, programmers started to implement applications as sets of cooperating processes (e. g., one process gathering input data, one process processing input data, one process writing out results on disk). This, however, required some tools to allow processes to efficiently exchange data.

Threads were born from the idea that the most efficient way for cooperating processes to exchange data would be to share their entire memory space. Thus, threads are effectively processes that run in the same memory context and share other resources with their parent processes, such as open files. Threads are described as *lightweight processes* because switching between threads does not involve changing the memory context.[4][5][6]

While threads are scheduled preemptively, some operating systems provide a variant to threads, named *fibers*, that are scheduled cooperatively. On operating systems that do not provide fibers, an application may implement its own fibers using repeated calls to worker functions. Fibers are even more lightweight than threads, and somewhat easier to program with, although they tend to lose some or all of the benefits of threads on machines with multiple processors.

Some systems directly support multithreading in hardware.

2.6 Memory protection

Main article: Memory protection

Essential to any multitasking system is to safely and effectively share access to system resources. Access to memory must be strictly managed to ensure that no process can inadvertently or deliberately read or write to memory locations outside of the process's address space. This is done for the purpose of general system stability and data integrity, as well as data security.

In general, memory access management is the operating system kernel's responsibility, in combination with hardware mechanisms (such as the memory management unit (MMU)) that provide supporting functionalities. If a process attempts to access a memory location outside of its memory space, the MMU denies the request and signals the kernel to take appropriate actions; this usually results in forcibly terminating the offending process. Depending on the software and kernel design and the specific error in question, the user may receive an access violation error message such as "segmentation fault".

In a well designed and correctly implemented multitasking system, a given process can never directly access memory that belongs to another process. An exception to this rule is in the case of shared memory; for example, in the System V inter-process communication mechanism the kernel allocates memory to be mutually shared by multiple processes. Such features are often used by database management software such as PostgreSQL.

Inadequate memory protection mechanisms, either due to flaws in their design or poor implementations, allow for security vulnerabilities that may be potentially exploited by malicious software.

2.7 Memory swapping

Use of a swap file or swap partition is a way for the operating system to provide more memory than is physically available by keeping portions of the primary memory in secondary storage. While multitasking and memory swapping are two completely unrelated techniques, they are very often used together, as swapping memory allows more tasks to be loaded at the same time. Typically, a multitasking system allows another process to run when the running process hits a point where it has to wait for some portion of memory to be reloaded from secondary storage.

2.8 Programming

Processes that are entirely independent are not much trouble to program in a multitasking environment. Most of the complexity in multitasking systems comes from the need to share computer resources between tasks and to synchronize the operation of co-operating tasks.

Various concurrent computing techniques are used to avoid potential problems caused by multiple tasks attempting to access the same resource.

Bigger systems were sometimes built with a central processor(s) and some number of I/O processors, a kind of asymmetric multiprocessing.

Over the years, multitasking systems have been refined. Modern operating systems generally include detailed mechanisms for prioritizing processes, while symmetric multiprocessing has introduced new complexities and capabilities.

2.9 See also

- Process state

2.10 References

[1] "Preemptive multitasking". *riscos.info*. 2009-11-03. Retrieved 2014-07-27.

[2] "UNIX, Part One". *The Digital Research Initiative*. ibiblio.org. 2002-01-30. Retrieved 2014-01-09.

[3] Smart Computing Article - Windows 2000 &16-Bit Applications

[4] Eduardo Ciliendo; Takechika Kunimasa (April 25, 2008). "Linux Performance and Tuning Guidelines" (PDF). *redbooks.ibm.com*. IBM. p. 4. Retrieved March 1, 2015.

[5] "Context Switch Definition". *linfo.org*. May 28, 2006. Retrieved February 26, 2015.

[6] "What are threads (user/kernel)?". *tldp.org*. September 8, 1997. Retrieved February 26, 2015.

Chapter 3

Symmetric multiprocessing

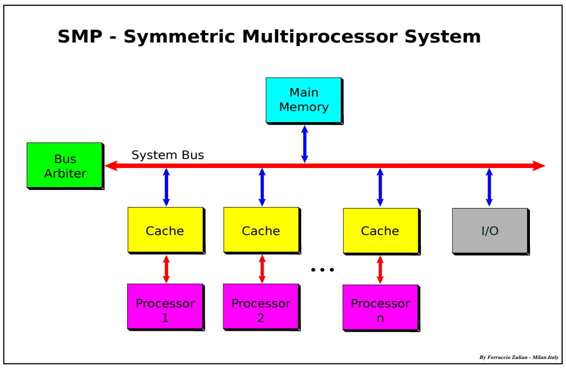


Diagram of a symmetric multiprocessing system

Symmetric multiprocessing (**SMP**) involves a symmetric multiprocessor system hardware and software architecture where two or more identical processors connect to a single, shared main memory, have full access to all I/O devices, and are controlled by a single operating system instance that treats all processors equally, reserving none for special purposes. Most multiprocessor systems today use an SMP architecture. In the case of multi-core processors, the SMP architecture applies to the cores, treating them as separate processors.

SMP systems are *tightly coupled multiprocessor systems* with a pool of homogeneous processors running independently, each processor executing different programs and working on different data and with capability of sharing common resources (memory, I/O device, interrupt system and so on) and connected using a system bus or a crossbar.

3.1 Design

SMP systems have centralized shared memory called *Main Memory* (MM) operating under a single operating system with two or more homogeneous processors. Usually each processor has an associated private high-speed memory known as cache memory (or cache) to speed-up the MM data access and to reduce the system bus traffic.

Processors may be interconnected using buses, crossbar switches or on-chip mesh networks. The bottleneck in the scalability of SMP using buses or crossbar switches is the bandwidth and power consumption of the interconnect among the various processors, the memory, and the disk arrays. Mesh architectures avoid these bottlenecks, and provide nearly linear scalability to much higher processor counts at the sacrifice of programmability:

> Serious programming challenges remain with this kind of architecture because it requires two distinct modes of programming, one for the CPUs themselves and one for the interconnect between the CPUs. A single programming language would have to be able to not only partition the workload, but also comprehend the memory locality, which is severe in a mesh-based architecture.[1]

SMP systems allow any processor to work on any task no matter where the data for that task are located in memory, provided that each task in the system is not in execution on two or more processors at the same time; with proper operating system support, SMP systems can easily move tasks between processors to balance the workload efficiently.

3.2 History

The earliest production system with multiple identical processors was the Burroughs B5000 which was functional in around 1961. However at run-time this was asymmetric, with one processor restricted to application programs while the other mainly handled the operating system and hardware interrupts.

IBM offered dual-processor computer systems based on its System/360 model 65 and the closely related model 67.[2] and 67-2[3] The operating systems that ran on these machines were OS/360 M65MP[4] and TSS/360. Other software, developed at universities, used both CPUs—notably MTS. Both processors could access data channels and initiate I/O.

In OS/360 M65MP, since the operating system kernel ran on both processors (though with a “big lock” around the

I/O handler) and peripherals could generally be attached to either processor.[5]

The MTS supervisor (UMMPS) ran on either or both CPUs of the IBM System/360 model 67-2. Supervisor locks were small and were used to protect individual common data structures that might be accessed simultaneously from either CPU.[6]

Digital Equipment Corporation's first multi-processor VAX system, the VAX-11/782, was asymmetric,[7] but later VAX multiprocessor systems were SMP.[8]

The first commercial Unix SMP implementation was the NUMA based Honeywell Information Systems Italy XPS-100 designed by Dan Gielan of VAST Corporation in 1985. Its design supported up to 14 processors although due to electrical limitations the largest marketed version was a dual processor system. The operating system was derived and ported by VAST Corporation from AT&T 3B20 Unix SysVr3 code used internally within AT&T.

3.3 Uses

Time-sharing and server systems can often use SMP without changes to applications, as they may have multiple processes running in parallel, and a system with more than one process running can run different processes on different processors.

On personal computers, SMP is less useful for applications that have not been modified. If the system rarely runs more than one process at a time, SMP is useful only for applications that have been modified for multithreaded (multitasked) processing. Custom-programmed software can be written or modified to use multiple threads, so that it can make use of multiple processors. However, most consumer products such as word processors and computer games are written in such a manner that they cannot gain large benefits from concurrent systems. For games this is usually because writing a program to increase performance on SMP systems can produce a performance loss on uniprocessor systems. Recently, however, multi-core chips are becoming more common in new computers, and the balance between installed uni- and multi-core computers may change in the coming years.

Multithreaded programs can also be used in time-sharing and server systems that support multithreading, allowing them to make more use of multiple processors.

3.4 Programming

Uniprocessor and SMP systems require different programming methods to achieve maximum performance. Programs running on SMP systems may experience a performance increase even when they have been written for uniprocessor systems. This is because hardware interrupts that usually suspend program execution while the kernel handles them can execute on an idle processor instead. The effect in most applications (e.g. games) is not so much a performance increase as the appearance that the program is running much more smoothly. Some applications, particularly compilers and some distributed computing projects, run faster by a factor of (nearly) the number of additional processors.

Systems programmers must build support for SMP into the operating system: otherwise, the additional processors remain idle and the system functions as a uniprocessor system.

SMP systems can also lead to more complexity regarding instruction sets. A homogeneous processor system typically requires extra registers for “special instructions” such as SIMD (MMX, SSE, etc.), while a heterogeneous system can implement different types of hardware for different instructions/uses.

3.5 Performance

When more than one program executes at the same time, an SMP system has considerably better performance than a uni-processor, because different programs can run on different CPUs simultaneously.

In cases where an SMP environment processes many jobs, administrators often experience a loss of hardware efficiency. Software programs have been developed to schedule jobs so that the processor utilization reaches its maximum potential. Good software packages can achieve this maximum potential by scheduling each CPU separately, as well as being able to integrate multiple SMP machines and clusters.

Access to RAM is serialized; this and cache coherency issues causes performance to lag slightly behind the number of additional processors in the system.

3.6 Systems

3.6.1 Entry-level systems

Before about 2006, entry-level servers and workstations with two processors dominated the SMP market. With the introduction of dual-core devices, SMP is found in most new desktop machines and in many laptop machines. The most popular entry-level SMP systems use the x86 instruction set architecture and are based on Intel’s Xeon, Pentium D, Core Duo, and Core 2 Duo based processors or AMD’s Athlon64 X2, Quad FX or Opteron 200 and 2000 series processors. Servers use those processors and other readily available non-x86 processor choices, including the Sun Microsystems UltraSPARC, Fujitsu SPARC64 III and later, SGI MIPS, Intel Itanium, Hewlett Packard PA-RISC, Hewlett-Packard (merged with Compaq, which acquired first Digital Equipment Corporation) DEC Alpha, IBM POWER and PowerPC (specifically G4 and G5 series, as well as earlier PowerPC 604 and 604e series) processors. In all cases, these systems are available in uniprocessor versions as well.

Earlier SMP systems used motherboards that have two or more CPU sockets. More recently, microprocessor manufacturers introduced CPU devices with two or more processors in one device, for example, the Itanium, POWER, UltraSPARC, Opteron, Athlon, Core 2, and Xeon all have multi-core variants. Athlon and Core 2 Duo multiprocessors are socket-compatible with uniprocessor variants, so an expensive dual socket motherboard is no longer needed to implement an entry-level SMP machine. It should also be noted that dual socket Opteron designs are technically ccNUMA designs, though they can be programmed as SMP for a slight loss in performance. Software based SMP systems can be created by linking smaller systems together. An example of this is the software developed by ScaleMP.

With the introduction of ARM Cortex-A9 multi-core SoCs, low-cost symmetric multiprocessing embedded systems began to flourish in the form of smartphones and tablet computers with a multi-core processor.

3.6.2 Mid-level systems

The Burroughs D825 first implemented SMP in 1962.[9][10] It was implemented later on other mainframes. Mid-level servers, using between four and eight processors, can be found using the Intel Xeon MP, AMD Opteron 800 and 8000 series and the above-mentioned UltraSPARC, SPARC64, MIPS, Itanium, PA-RISC, Alpha and POWER processors. High-end systems, with sixteen or more processors, are also available with all of the above processors.

Sequent Computer Systems built large SMP machines using Intel 80386 (and later 80486) processors. Some smaller 80486 systems existed, but the major x86 SMP market began with the Intel Pentium technology supporting up to two processors. The Intel Pentium Pro expanded SMP support with up to four processors natively. Later, the Intel Pentium II, and Intel Pentium III processors allowed dual CPU systems, except for the respective Celerons. This was followed by the Intel Pentium II Xeon and Intel Pentium III Xeon processors, which could be used with up to four processors in a system natively. In 2001 AMD released their Athlon MP, or MultiProcessor CPU, together with the 760MP motherboard chipset as their first offering in the dual processor marketplace. Although several much larger systems were built, they were all limited by the physical memory addressing limitation of 64 GiB. With the introduction of 64-bit memory addressing on the AMD64 Opteron in 2003 and Intel 64 (EM64T) Xeon in 2005, systems are able to address much larger amounts of memory; their addressable limitation of 16 EiB is not expected to be reached in the foreseeable future.

3.7 Alternatives

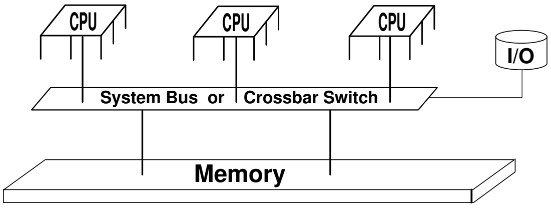


Diagram of a typical SMP system. Three processors are connected to the same memory module through a system bus or crossbar switch

SMP using a single shared system bus represents one of the earliest styles of multiprocessor machine architectures, typically used for building smaller computers with up to 8 processors.

Larger computer systems might use newer architectures such as NUMA (Non-Uniform Memory Access), which dedicates different memory banks to different processors. In a NUMA architecture, processors may access local memory quickly and remote memory more slowly. This can dramatically improve memory throughput as long as the data are localized to specific processes (and thus processors). On the downside, NUMA makes the cost of moving data from one processor to another, as in workload balancing, more expensive. The benefits of NUMA are limited to particular workloads, notably on servers where the data are often associated strongly with certain tasks or users.

Finally, there is computer clustered multiprocessing (such as Beowulf), in which not all memory is available to all processors. Clustering techniques are used fairly extensively to build very large supercomputers.

3.8 See also

- Asymmetric multiprocessing
- Binary Modular Dataflow Machine
- Locale (computer hardware)
- Massively parallel
- Non-Uniform Memory Access
- Sequent Computer Systems

- Simultaneous multithreading, where functional elements of a CPU core are allocated across multiple threads of execution.
- Software lockout
- Xeon Phi

3.9 References

[1] Lina J. Karam, Ismail AlKamal, Alan Gatherer, Gene A. Frantz, David V. Anderson, Brian L. Evans (2009). "Trends in Multi-core DSP Platforms". *IEEE Signal Processing Magazine, Special Issue on Signal Processing on Platforms with Multiple Cores*.

[2] IBM (September 1968). *IBM System/360 Model 65 Functional Characteristics*. Fourth Edition. A22-6884-3.

[3] IBM (February 1972). *IBM System/360 Model 67 Functional Characteristics*. Third Edition. GA27-2719-2.

[4] M65MP: An Experiment in OS/360 multiprocessing

[5] IBM, "OS I/O Supervisor PLM" - GY28-6616-9, Program Logic Manual, R21.7, April 1973

[6] *Time Sharing Supervisor Programs* by Mike Alexander (May 1971) has information on MTS, TSS, CP/67, and Multics

[7] VAX Product Sales Guide, pages 1-23 and 1-24: the VAX-11/782 is described as an asymmetric multiprocessing system in 1982

[8] VAX 8820/8830/8840 System Hardware User's Guide: by 1988 the VAX operating system was SMP

[9] 1962

[10] 1964 BRL Report

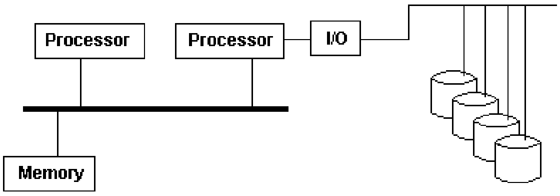
3.10 External links

- History of Multi-Processing
- Practical Parallel Programming in Pascal
- Linux and Multiprocessing
- Multicore News blog
- AMD

Chapter 4

Asymmetric multiprocessing

Asymmetric multiprocessing (**AMP**) was a software stopgap for handling multiple CPUs before symmetric multiprocessing (SMP) was available. It has also been used to provide less expensive options[1] on systems where SMP was available. In an asymmetric multiprocessing system, not all CPUs are treated equally; for example, a system might only allow (either at the hardware or operating system level) one CPU to execute operating system code or might only allow one CPU to perform I/O operations. Other AMP systems would allow any CPU to execute operating system code and perform I/O operations, so that they were symmetric with regard to processor roles, but attached some or all peripherals to particular CPUs, so that they were asymmetric with regard to peripheral attachment.



Asymmetric multiprocessing

Multiprocessing is the use of more than one CPU in a computer system. The CPU is the arithmetic and logic engine that executes user applications; an I/O interface such as a GPU, even if it is implemented using an embedded processor, does not constitute a CPU because it does not run the user's application program. With multiple CPUs, more than one set of program instructions can be executed at the same time. All of the CPUs have the same user-mode instruction set, so a running job can be rescheduled from one CPU to another.[2]

4.1 Background and history

For the room-size computers of the 1960s and 1970s, a cost-effective way to increase compute power was to add a second CPU. Since these computers were already close to the fastest available (near the peak of the price:performance ratio), two standard-speed CPUs were much less expensive than a CPU that ran twice as fast. Also, adding a second CPU was less expensive than a second complete computer, which would need its own peripherals, thus requiring much more floor space and an increased operations staff.

Notable early offerings by computer manufacturers were the Burroughs B5000, the DECsystem-1055, and the IBM System/360 model 65MP. There were also dual-CPU machines built at universities.[3]

The problem with adding a second CPU to a computer system was that the operating system had been developed for single-CPU systems, and extending it to handle multiple CPUs efficiently and reliably took a long time. To fill the gap, operating systems intended for single CPUs were initially extended to provide minimal support for a second CPU. In this minimal support, the operating system ran on the "boot" processor, with the other only allowed to run user programs. In the case of the Burroughs B5000, the second processor's hardware was not capable of running "control state" code.[4]

Other systems allowed the operating system to run on all processors, but either attached all the peripherals to one processor or attached particular peripherals to particular processors.

4.2 Burroughs B5000 and B5500

An option on the Burroughs B5000 was "Processor B". This second processor, unlike "Processor A" had no connection to the peripherals, though the two processors shared main memory, and Processor B could not run in Control State.[4] The operating system ran only on Processor A. When there was a user job to be executed, it might be run on Processor B, but when that job tried to access the operating system the processor halted and signaled Processor A. The requested operating system service was then run on Processor A.

On the B5500, either Processor A or Processor B could be designated as Processor 1 by a switch on the engineer's panel, with the other processor being Processor 2; both processors shared main memory and had hardware access to the I/O processors hence the peripherals, but only Processor 1 could respond to peripheral interrupts.[5] When a job on Processor 2 required an operating system service it would be rescheduled on Processor 1, which was responsible for both initiating I/O processor activity and responding to interrupts indicating completion. In practice, this meant that while user jobs could run on either Processor 1 or Processor 2 and could access intrinsic library routines that didn't require kernel support, the operating system would schedule them on the latter whenever possible.[6]

4.3 CDC 6500 and 6700

Control Data Corporation offered two configurations of its CDC 6000 series that featured two central processors. The CDC 6500[7] was a CDC 6400 with two central processors. The CDC 6700 was a CDC 6600 with the CDC 6400 central processor added to it.

These systems were organized quite differently from the other multiprocessors in this article. The operating system ran on the peripheral processors, while the user's application ran on the CPUs. Thus, the terms ASMP and SMP do not properly apply to these multiprocessors.

4.4 DECsystem-1055

Digital Equipment Corporation (DEC) offered a dual-processor version of its DECsystem-1050 which used two KA10 processors.[8][9] This offering was extended to later processors in the PDP-10 line.

4.5 PDP-11/74

Digital Equipment Corporation developed, but never released, a multiprocessor PDP-11, the PDP-11/74,[10] running a multiprocessor version of RSX-11M.[11] In that system, either processor could run operating system code, and could perform I/O, but not all peripherals were accessible to all processors; most peripherals were attached to one or the other of the CPUs, so that a processor to which a peripheral wasn't attached would, when it needed to perform an I/O operation on that peripheral, request the processor to which the peripheral was attached to perform the operation.[11]

4.6 VAX-11/782

DEC's first multi-processor VAX system, the VAX-11/782, was an asymmetric dual-processor system; only the first processor had access to the I/O devices.[12]

4.7 Univac 1108-II

The Univac 1108-II and its successors had up to three CPUs.[13][14] These computers ran the UNIVAC EXEC 8 operating system, but it is not clear from the surviving documentation where that operating system was on the path from asymmetric to symmetric multiprocessing.

4.8 IBM System/370 model 168

Two options were available for the IBM System/370 model 168 for attaching a second processor.[15] One was the IBM 3062 Attached Processing Unit, in which the second processor had no access to the channels, and was therefore similar to the B5000's Processor B or the second processor on a VAX-11/782. The other option offered a complete second CPU, and was thus more like the System/360 model 65MP.

4.9 See also

- 3B20C
- Multi-core (computing)
- Software lockout
- Giant lock
- Symmetric multiprocessing
- Heterogeneous computing
- big.LITTLE
- Tegra 3

4.10 Notes

[1] IBM (December 1976). *IBM System/370 System Summary*. Seventh Edition. pp. 6–12, 6–15–6.16.1. GA22·7001·6.

[2] Introduction to Multiprocessing: distinguishes "symmetric" from "master/slave"

[3] Early Computers at Stanford: the dual processor computer at the AI lab

[4] "Operational Characteristics of the Processors for the Burroughs B5000". Burroughs.

[5] "A Narrative Description of the B5500 MCP". p. 18.

[6] A Narrative Description of the B5500 MCP, pages 29 (initiate routine) and 40 (a note on parallel processing)

[7] CONTROL DATA 6400/6500/6600 COMPUTER SYSTEMS Reference Manual

[8] Introduction to DECsystem-10 Software, section 1.4 (DECsystem-10 Multiprocessing)

[9] DECsystem-10 Technical Summary page 2-1

[10] "(PDP-11) Multiprocessor FAQ”.

[11] “RSX-11M multiprocessing”. Digital Equipment Corporation.

[12] VAX Product Sales Guide, pages 1-23 and 1-24: the VAX-11/782 is described as an asymmetric multiprocessing system in 1982

[13] “Univac 1108-II announcement”. Sperry Rand.

[14] “A history of Univac computers and Operating Systems”.

[15] IBM (January 1976). *IBM System/370 Model 168 Functional Characteristics*. Fifth Edition. GA22·7010-4.

4.11 References

- Bell, C. Gordon, Mudge, J. Craig, McNamara John E. “The PDP-10 Family”. (1979). Part V of *Computer Engineering: A DEC View of Hardware Systems Design*. Digital Equipment Corp.

- Rajkumar Buyya (editor): *High Performance Cluster Computing: Architectures and Systems*, Volume 1, ISBN 0-13-013784-7, Prentice Hall, NJ, USA, 1999.

- Rajkumar Buyya (editor): *High Performance Cluster Computing: Programming and Applications*, Volume 2, ISBN 0-13-013785-5, Prentice Hall, NJ, USA, 1999.

4.12 External links

- OpenMP tutorial for parallel programming

- Multicore News blog

- History of Multi-Processing

- Linux and Multiprocessing

- ASOSI: Asymmetric Operating System Infrastructure, Proc. 21st Conference on Parallel and Distributed Computing and Communication Systems, (PDCCS 2008), New Orleans, Louisiana, pp. 193-198, 2008

Chapter 5

Non-uniform memory access

For other people or places with the same name as this abbreviation, see Numa.

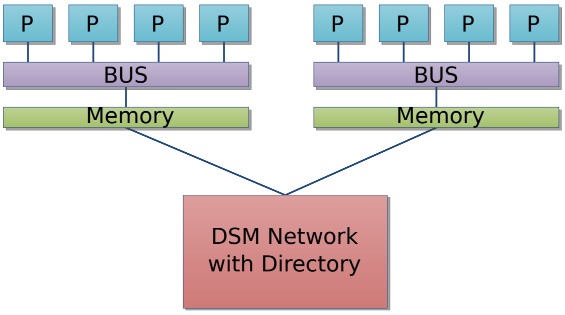
Non-uniform memory access (**NUMA**) is a computer memory design used in multiprocessing, where the memory access time depends on the memory location relative to the processor. Under NUMA, a processor can access its own local memory faster than non-local memory (memory local to another processor or memory shared between processors). The benefits of NUMA are limited to particular workloads, notably on servers where the data are often associated strongly with certain tasks or users.[1]

NUMA architectures logically follow in scaling from symmetric multiprocessing (SMP) architectures. They were developed commercially during the 1990s by Burroughs (later Unisys), Convex Computer (later Hewlett-Packard), Honeywell Information Systems Italy (HISI) (later Groupe Bull), Silicon Graphics (later Silicon Graphics International), Sequent Computer Systems (later IBM), Data General (later EMC), and Digital (later Compaq, now HP). Techniques developed by these companies later featured in a variety of Unix-like operating systems, and to an extent in Windows NT.

The first commercial implementation of a NUMA-based Unix system was the Symmetrical Multi Processing XPS-100 family of servers, designed by Dan Gielan of VAST Corporation for Honeywell Information Systems Italy.

5.1 Basic concept

Modern CPUs operate considerably faster than the main memory they use. In the early days of computing and data processing, the CPU generally ran slower than its own memory. The performance lines of processors and memory crossed in the 1960s with the advent of the first supercomputers. Since then, CPUs increasingly have found themselves “starved for data” and having to stall while waiting for data to arrive from memory. Many supercomputer designs of the 1980s and 1990s focused on providing high-speed memory access as opposed to faster processors, allowing the computers to work on large data sets at speeds other systems could not approach.



One possible architecture of a NUMA system. The processors connect to the bus or crossbar by connections of varying thickness/number. This shows that different CPUs have different access priorities to memory based on their relative location.

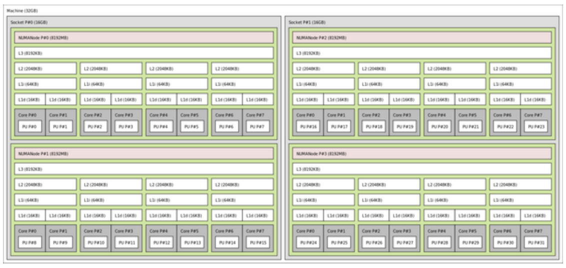
Limiting the number of memory accesses provided the key to extracting high performance from a modern computer. For commodity processors, this meant installing an ever-increasing amount of high-speed cache memory and using increasingly sophisticated algorithms to avoid cache misses. But the dramatic increase in size of the operating systems and of the applications run on them has generally overwhelmed these cache-processing improvements. Multi-processor systems without NUMA make the problem considerably worse. Now a system can starve several processors at the same time, notably because only one processor can access the computer’s memory at a time.[2]

NUMA attempts to address this problem by providing separate memory for each processor, avoiding the performance hit when several processors attempt to address the same memory. For problems involving spread data (common for servers and similar applications), NUMA can improve the performance over a single shared memory by a factor of roughly the number of processors (or separate memory banks).[3] Another approach to addressing this problem, utilized mainly by non-NUMA systems, is the multi-channel memory architecture; multiple memory channels are increasing the number of simultaneous memory accesses.[4]

Of course, not all data ends up confined to a single task, which means that more than one processor may require the same data. To handle these cases, NUMA systems include additional hardware or software to move data between memory banks. This operation slows the processors attached to those banks, so the overall speed increase due to NUMA depends heavily on the nature of the running tasks.[3]

Intel announced NUMA compatibility for its x86 and Itanium servers in late 2007 with its Nehalem and Tukwila CPUs.[5] Both CPU families share a common chipset; the interconnection is called Intel Quick Path Interconnect (QPI).[6] AMD implemented NUMA with its Opteron processor (2003), using HyperTransport. Freescale's NUMA for PowerPC is called CoreNet.

5.2 Cache coherent NUMA (ccNUMA)



Topology of a ccNUMA Bulldozer server.

Nearly all CPU architectures use a small amount of very fast non-shared memory known as cache to exploit locality of reference in memory accesses. With NUMA, maintaining cache coherence across shared memory has a significant overhead. Although simpler to design and build, non-cache-coherent NUMA systems become prohibitively complex to program in the standard von Neumann architecture programming model.[7]

Typically, ccNUMA uses inter-processor communication between cache controllers to keep a consistent memory image when more than one cache stores the same memory location. For this reason, ccNUMA may perform poorly when multiple processors attempt to access the same memory area in rapid succession. Support for NUMA in operating systems attempts to reduce the frequency of this kind of access by allocating processors and memory in NUMA-friendly ways and by avoiding scheduling and locking algorithms that make NUMA-unfriendly accesses necessary.[8]

Alternatively, cache coherency protocols such as the MESIF protocol attempt to reduce the communication required to maintain cache coherency. Scalable Coherent Interface (SCI) is an IEEE standard defining a directory-based cache coherency protocol to avoid scalability limitations found in earlier multiprocessor systems. For example, SCI is used as the basis for the NumaConnect technology.[9][10]

As of 2011, ccNUMA systems are multiprocessor systems based on the AMD Opteron processor, which can be implemented without external logic, and the Intel Itanium processor, which requires the chipset to support NUMA. Examples of ccNUMA-enabled chipsets are the SGI Shub (Super hub), the Intel E8870, the HP sx2000 (used in the Integrity and Superdome servers), and those found in NEC Itanium-based systems. Earlier ccNUMA systems such as those from Silicon Graphics were based on MIPS processors and the DEC Alpha 21364 (EV7) processor.

5.3 NUMA vs. cluster computing

One can view NUMA as a tightly coupled form of cluster computing. The addition of virtual memory paging to a cluster architecture can allow the implementation of NUMA entirely in software. However, the inter-node latency of software-based NUMA remains several orders of magnitude greater (slower) than that of hardware-based NUMA.[1]

5.4 Software support

Since NUMA largely influences memory access performance, certain software optimizations are needed to allow scheduling threads and processes close to their in-memory data.

- Microsoft Windows 7 and Windows Server 2008 R2 add support for NUMA architecture over 64 logical cores.[11]
- Java 7 added support for NUMA-aware memory allocator and garbage collector.[12]
- The Linux kernel 2.5 already had basic support built-in,[13] which was further extended in subsequent releases. Linux kernel version 3.8 brought a new NUMA foundation which allowed more efficient NUMA policies to be built in the next kernel releases.[14][15] Linux kernel version 3.13 brought numerous policies that attempt to put a process near its memory, together with handling of cases such as shared pages between processes, or transparent huge pages; new sysctl settings are allowing NUMA balancing to be enabled or disabled, as well as various NUMA memory balancing parameters to be configured.[16][17][18]
- OpenSolaris models NUMA architecture with lgroups.

5.5 See also

- Uniform memory access (UMA)

- Cluster computing
- Symmetric multiprocessing (SMP)
- Cache only memory architecture (COMA)
- Scratchpad memory (SPM)
- Supercomputer
- Silicon Graphics (SGI)
- HiperDispatch
- Intel QuickPath Interconnect (QPI)
- HyperTransport

5.6 References

[1] Nakul Manchanda; Karan Anand (2010-05-04). "Non-Uniform Memory Access (NUMA)". New York University. Retrieved 2014-01-27.

[2] Sergey Blagodurov; Sergey Zhuravlev; Mohammad Dashti; Alexandra Fedorov (2011-05-02). "A Case for NUMA-aware Contention Management on Multicore Systems" (PDF). Simon Fraser University. Retrieved 2014-01-27.

[3] Zoltan Majo; Thomas R. Gross (2011). "Memory System Performance in a NUMA Multicore Multiprocessor" (PDF). ACM. Retrieved 2014-01-27.

[4] "Intel Dual-Channel DDR Memory Architecture White Paper" (PDF, 1021 KB) (Rev. 1.0 ed.). Infineon Technologies North America and Kingston Technology. September 2003. Archived from the original on 2011-09-29. Retrieved 2007-09-06.

[5] Intel Corp. (2008). Intel QuickPath Architecture [White paper]. Retrieved from http://www.intel.com/pressroom/archive/reference/whitepaper_QuickPath.pdf

[6] Intel Corporation. (September 18th, 2007). Gelsinger Speaks To Intel And High-Tech Industry's Rapid Technology Caden[Press release]. Retrieved from http://www.intel.com/pressroom/archive/releases/2007/20070918corp_b.htm

[7] "ccNUMA: Cache Coherent Non-Uniform Memory Access". slideshare.net. 2014. Retrieved 2014-01-27.

[8] Per Stenstromt; Truman Joe; Anoop Gupta (2002). "Comparative Performance Evaluation of Cache-Coherent NUMA and COMA Architectures" (PDF). ACM. Retrieved 2014-01-27.

[9] David B. Gustavson (September 1991). "The Scalable Coherent Interface and Related Standards Projects". *SLAC Publication 5656*. Stanford Linear Accelerator Center. Retrieved January 27, 2014.

[10] "The NumaChip enables cache coherent low cost shared memory". Numascale.com. Retrieved 2014-01-27.

[11] NUMA Support (MSDN)

[12] Java HotSpot™ Virtual Machine Performance Enhancements

[13] "Linux Scalability Effort: NUMA Group Homepage". *sourceforge.net*. 2002-11-20. Retrieved 2014-02-06.

[14] "Linux kernel 3.8, Section 1.8. Automatic NUMA balancing". *kernelnewbies.org*. 2013-02-08. Retrieved 2014-02-06.

[15] Jonathan Corbet (2012-11-14). "NUMA in a hurry". LWN.net. Retrieved 2014-02-06.

[16] "Linux kernel 3.13, Section 1.6. Improved performance in NUMA systems". *kernelnewbies.org*. 2014-01-19. Retrieved 2014-02-06.

[17] "Linux kernel documentation: Documentation/sysctl/kernel.txt". kernel.org. Retrieved 2014-02-06.

[18] Jonathan Corbet (2013-10-01). "NUMA scheduling progress". LWN.net. Retrieved 2014-02-06.

This article is based on material taken from the Free On-line Dictionary of Computing prior to 1 November 2008 and incorporated under the "relicensing" terms of the GFDL, version 1.3 or later.

5.7 External links

- NUMA FAQ
- Page-based distributed shared memory
- OpenSolaris NUMA Project
- Introduction video for the Alpha EV7 system architecture
- More videos related to EV7 systems: CPU, IO, etc
- NUMA optimization in Windows Applications
- NUMA Support in Linux at SGI
- Intel Tukwila
- Intel QPI (CSI) explained
- current Itanium NUMA systems

Chapter 6

Multi-core processor

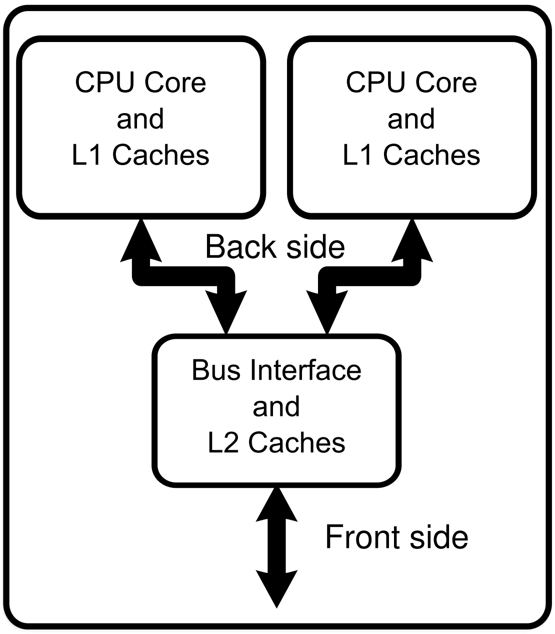
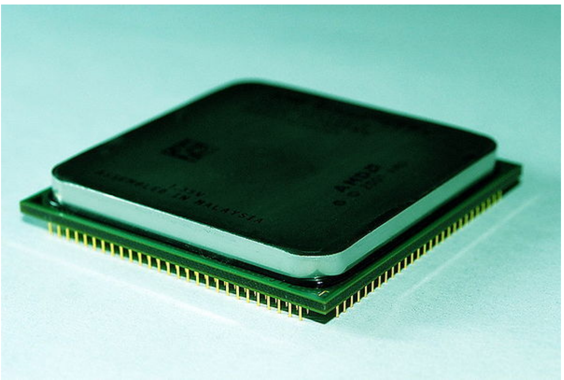


Diagram of a generic dual-core processor, with CPU-local level 1 caches, and a shared, on-die level 2 cache.

An Intel Core 2 Duo E6750 dual-core processor.



An AMD Athlon X2 6400+ dual-core processor.

A **multi-core processor** is a single computing component with two or more independent actual processing units (called "cores"), which are the units that read and execute program instructions.[1] The instructions are ordinary CPU instructions such as add, move data, and branch, but the multiple cores can run multiple instructions at the same time, increasing overall speed for programs amenable to parallel computing.[2] Manufacturers typically integrate the cores onto a single integrated circuit die (known as a chip multiprocessor or CMP), or onto multiple dies in a single chip package.

Processors were originally developed with only one core. In the mid 1980s Rockwell International manufactured versions of the 6502 with two 6502 cores on one chip as the R65C00, R65C21, and R65C29,[3][4] sharing the chip's pins on alternate clock phases. Other multi-core processors were developed in the early 2000s by Intel, AMD and others.

Multi-core processors may have two cores (dual-core CPUs, for example AMD Phenom II X2 and Intel Core Duo), four cores (quad-core CPUs, for example AMD Phenom II X4, Intel's i5 and i7 processors), six cores (hexa-core CPUs, for example AMD Phenom II X6 and Intel Core i7 Extreme Edition 980X), eight cores (octo-core CPUs, for example Intel Xeon E7-2820 and AMD FX-8350), ten cores (for example, Intel Xeon E7-2850), or more.

A multi-core processor implements multiprocessing in a single physical package. Designers may couple cores in a multi-core device tightly or loosely. For example, cores may or may not share caches, and they may implement message passing or shared memory inter-core communication methods. Common network topologies to interconnect cores include bus, ring, two-dimensional mesh, and crossbar. *Homogeneous* multi-core systems include only identical cores, heterogeneous multi-core systems have cores that are not identical. Just as with single-processor systems, cores in multi-core systems may implement architectures such as superscalar, VLIW, vector processing, SIMD, or multithreading.

Multi-core processors are widely used across many application domains including general-purpose, embedded, network, digital signal processing (DSP), and graphics.

The improvement in performance gained by the use of a multi-core processor depends very much on the software algorithms used and their implementation. In particular, possible gains are limited by the fraction of the software that can be run in parallel simultaneously on multiple cores; this effect is described by Amdahl's law. In the best case, so-called embarrassingly parallel problems may realize speedup factors near the number of cores, or even more if the problem is split up enough to fit within each core's cache(s), avoiding use of much slower main system memory. Most applications, however, are not accelerated so much unless programmers invest a prohibitive amount of effort in re-factoring the whole problem.[5] The parallelization of software is a significant ongoing topic of research.

6.1 Terminology

The terms *multi-core* and *dual-core* most commonly refer to some sort of central processing unit (CPU), but are sometimes also applied to digital signal processors (DSP) and system-on-a-chip (SoC). The terms are generally used only to refer to multi-core microprocessors that are manufactured on the *same* integrated circuit die; separate microprocessor dies in the same package are generally referred to by another name, such as *multi-chip module*. This article uses the terms "multi-core" and "dual-core" for CPUs manufactured on the *same* integrated circuit, unless otherwise noted.

In contrast to multi-core systems, the term *multi-CPU* refers to multiple physically separate processing-units (which often contain special circuitry to facilitate communication between each other).

The terms *many-core* and *massively multi-core* are sometimes used to describe multi-core architectures with an especially high number of cores (tens or hundreds).[6]

Some systems use many soft microprocessor cores placed on a single FPGA. Each "core" can be considered a "semiconductor intellectual property core" as well as a CPU core.

6.2 Development

While manufacturing technology improves, reducing the size of individual gates, physical limits of semiconductor-based microelectronics have become a major design concern. These physical limitations can cause significant heat dissipation and data synchronization problems. Various other methods are used to improve CPU performance. Some *instruction-level parallelism* (ILP) methods such as superscalar pipelining are suitable for many applications, but are inefficient for others that contain difficult-to-predict code. Many applications are better suited to *thread level parallelism* (TLP) methods, and multiple independent CPUs are commonly used to increase a system's overall TLP. A combination of increased available space (due to refined manufacturing processes) and the demand for increased TLP led to the development of multi-core CPUs.

6.2.1 Commercial incentives

Several business motives drive the development of multi-core architectures. For decades, it was possible to improve performance of a CPU by shrinking the area of the integrated circuit, which drove down the cost per device on the IC. Alternatively, for the same circuit area, more transistors could be used in the design, which increased functionality, especially for CISC architectures. Clock rates also increased by orders of magnitude in the decades of the late 20th century, from several megahertz in the 1980s to several gigahertz in the early 2000s.

As the rate of clock speed improvements slowed, increased use of parallel computing in the form of multi-core processors has been pursued to improve overall processing performance. Multiple cores were used on the same CPU chip, which could then lead to better sales of CPU chips with two or more cores. Intel has produced a 48-core processor for research in cloud computing; each core has an X86 architecture.[7] Intel has loaded Linux on each core.[8]

6.2.2 Technical factors

Since computer manufacturers have long implemented symmetric multiprocessing (SMP) designs using discrete CPUs, the issues regarding implementing multi-core processor architecture and supporting it with software are well known.

Additionally:

- Using a proven processing-core design without architectural changes reduces design risk significantly.
- For general-purpose processors, much of the motivation for multi-core processors comes from greatly

diminished gains in processor performance from increasing the operating frequency. This is due to three primary factors:

1. The *memory wall*; the increasing gap between processor and memory speeds. This, in effect, pushes for cache sizes to be larger in order to mask the latency of memory. This helps only to the extent that memory bandwidth is not the bottleneck in performance.
2. The *ILP wall*; the increasing difficulty of finding enough parallelism in a single instruction stream to keep a high-performance single-core processor busy.
3. The *power wall*; the trend of consuming exponentially increasing power with each factorial increase of operating frequency. This increase can be mitigated by "shrinking" the processor by using smaller traces for the same logic. The *power wall* poses manufacturing, system design and deployment problems that have not been justified in the face of the diminished gains in performance due to the *memory wall* and *ILP wall*.

In order to continue delivering regular performance improvements for general-purpose processors, manufacturers such as Intel and AMD have turned to multi-core designs, sacrificing lower manufacturing-costs for higher performance in some applications and systems. Multi-core architectures are being developed, but so are the alternatives. An especially strong contender for established markets is the further integration of peripheral functions into the chip.

6.2.3 Advantages

The proximity of multiple CPU cores on the same die allows the cache coherency circuitry to operate at a much higher clock-rate than is possible if the signals have to travel off-chip. Combining equivalent CPUs on a single die significantly improves the performance of cache snoop (alternative: Bus snooping) operations. Put simply, this means that signals between different CPUs travel shorter distances, and therefore those signals degrade less. These higher-quality signals allow more data to be sent in a given time period, since individual signals can be shorter and do not need to be repeated as often.

Assuming that the die can physically fit into the package, multi-core CPU designs require much less printed circuit board (PCB) space than do multi-chip SMP designs. Also, a dual-core processor uses slightly less power than two coupled single-core processors, principally because of the decreased power required to drive signals external to the chip. Furthermore, the cores share some circuitry, like the L2 cache and the interface to the front side bus (FSB). In terms of competing technologies for the available silicon die area, multi-core design can make use of proven CPU core library designs and produce a product with lower risk of design error than devising a new wider core-design. Also, adding more cache suffers from diminishing returns.

Multi-core chips also allow higher performance at lower energy. This can be a big factor in mobile devices that operate on batteries. Since each and every core in multi-core is generally more energy-efficient, the chip becomes more efficient than having a single large monolithic core. This allows higher performance with less energy. The challenge of writing parallel code clearly offsets this benefit.[9]

6.2.4 Disadvantages

Maximizing the usage of the computing resources provided by multi-core processors requires adjustments both to the operating system (OS) support and to existing application software. Also, the ability of multi-core processors to increase application performance depends on the use of multiple threads within applications.

Integration of a multi-core chip drives chip production yields down and they are more difficult to manage thermally than lower-density single-core designs. Intel has partially countered this first problem by creating its quad-core designs by combining two dual-core on a single die with a unified cache, hence any two working dual-core dies can be used, as opposed to producing four cores on a single die and requiring all four to work to produce a quad-core. From an architectural point of view, ultimately, single CPU designs may make better use of the silicon surface area than multiprocessing cores, so a development commitment to this architecture may carry the risk of obsolescence. Finally, raw processing power is not the only constraint on system performance. Two processing cores sharing the same system bus and memory bandwidth limits the real-world performance advantage. It has been claimed that if a single core is close to being memory-bandwidth limited, then going to dual-core might give 30% to 70% improvement; if memory bandwidth is not a problem, then a 90% improvement can be expected; however, Amdahl's law makes this claim dubious.[10] It would be possible for an application that used two CPUs to end up running faster on one dual-core if communication between the CPUs was the limiting factor, which would count as more than 100% improvement.

6.3 Hardware

6.3.1 Trends

The general trend in processor development has moved from dual-, tri-, quad-, hex-, oct-core chips to ones with tens or even thousands of cores. In addition, multi-core chips mixed with simultaneous multithreading, memory-on-chip, and special-purpose “heterogeneous” cores promise further performance and efficiency gains, especially in processing multimedia, recognition and networking applications. There is also a trend of improving energy-efficiency by focusing on performance-per-watt with advanced fine-grain or ultra fine-grain power management and dynamic voltage and frequency scaling (i.e. laptop computers and portable media players).

6.3.2 Architecture

The composition and balance of the cores in multi-core architecture show great variety. Some architectures use one core design repeated consistently (“homogeneous”), while others use a mixture of different cores, each optimized for a different, "heterogeneous" role.

The article “CPU designers debate multi-core future” by Rick Merritt, EE Times 2008,[11] includes these comments:

> Chuck Moore [...] suggested computers should be more like cellphones, using a variety of specialty cores to run modular software scheduled by a high-level applications programming interface.
>
> [...] Atsushi Hasegawa, a senior chief engineer at Renesas, generally agreed. He suggested the cellphone’s use of many specialty cores working in concert is a good model for future multi-core designs.
>
> [...] Anant Agarwal, founder and chief executive of startup Tilera, took the opposing view. He said multi-core chips need to be homogeneous collections of general-purpose cores to keep the software model simple.

6.4 Software effects

An outdated version of an anti-virus application may create a new thread for a scan process, while its GUI thread waits for commands from the user (e.g. cancel the scan). In such cases, a multi-core architecture is of little benefit for the application itself due to the single thread doing all the heavy lifting and the inability to balance the work evenly across multiple cores. Programming truly multithreaded code often requires complex co-ordination of threads and can easily introduce subtle and difficult-to-find bugs due to the interweaving of processing on data shared between threads (thread-safety). Consequently, such code is much more difficult to debug than single-threaded code when it breaks. There has been a perceived lack of motivation for writing consumer-level threaded applications because of the relative rarity of consumer-level demand for maximum use of computer hardware. Although threaded applications incur little additional performance penalty on single-processor machines, the extra overhead of development has been difficult to justify due to the preponderance of single-processor machines. Also, serial tasks like decoding the entropy encoding algorithms used in video codecs are impossible to parallelize because each result generated is used to help create the next result of the entropy decoding algorithm.

Given the increasing emphasis on multi-core chip design, stemming from the grave thermal and power consumption problems posed by any further significant increase in processor clock speeds, the extent to which software can be multithreaded to take advantage of these new chips is likely to be the single greatest constraint on computer performance in the future. If developers are unable to design software to fully exploit the resources provided by multiple cores, then they will ultimately reach an insurmountable performance ceiling.

The telecommunications market had been one of the first that needed a new design of parallel datapath packet processing because there was a very quick adoption of these multiple-core processors for the datapath and the control plane. These MPUs are going to replace[12] the traditional Network Processors that were based on proprietary micro- or pico-code.

Parallel programming techniques can benefit from multiple cores directly. Some existing parallel programming models such as Cilk Plus, OpenMP, OpenHMPP, FastFlow, Skandium, MPI, and Erlang can be used on multi-core platforms. Intel introduced a new abstraction for C++ parallelism called TBB. Other research efforts include the Codeplay Sieve System, Cray’s Chapel, Sun’s Fortress, and IBM’s X10.

Multi-core processing has also affected the ability of modern computational software development. Developers programming in newer languages might find that their modern languages do not support multi-core functionality. This then requires the use of numerical libraries to access code written in languages like C and Fortran, which perform math computations faster than newer languages like C#. Intel’s MKL and AMD’s ACML are written in these native languages and take advantage of multi-core processing. Balancing the application workload across processors can be problematic, especially if they have different performance characteristics. There are different conceptual models to deal with the problem, for example using a coordination language and program building blocks (programming libraries or higher-order functions). Each block can have a different native implementation for each processor type. Users simply program using these abstractions and an intelligent compiler chooses the best implementation based on the context.[13]

Managing concurrency acquires a central role in developing parallel applications. The basic steps in designing parallel applications are:

Partitioning The partitioning stage of a design is intended to expose opportunities for parallel execution. Hence, the focus is on defining a large number of small tasks in order to yield what is termed a fine-grained decomposition of a problem.

Communication The tasks generated by a partition are intended to execute concurrently but cannot, in general, execute independently. The computation to be performed in one task will typically require data associated with another task. Data must then be transferred between tasks so as to allow computation to proceed. This information flow is specified in the communication phase of a design.

Agglomeration In the third stage, development moves from the abstract toward the concrete. Developers revisit decisions made in the partitioning and communication phases with a view to obtaining an algorithm that will execute efficiently on some class of parallel computer. In particular, developers consider whether it is useful to combine, or agglomerate, tasks identified by the partitioning phase, so as to provide a smaller number of tasks, each of greater size. They also determine whether it is worthwhile to replicate data and computation.

Mapping In the fourth and final stage of the design of parallel algorithms, the developers specify where each task is to execute. This mapping problem does not arise on uniprocessors or on shared-memory computers that provide automatic task scheduling.

On the other hand, on the server side, multi-core processors are ideal because they allow many users to connect to a site simultaneously and have independent threads of execution. This allows for Web servers and application servers that have much better throughput.

6.4.1 Licensing

Vendors may license some software “per processor”. This can give rise to ambiguity, because a “processor” may consist either of a single core or of a combination of cores.

- Microsoft has stated that it would treat a socket as a single processor.[14][15]

- Oracle Corporation counts an AMD X2 or an Intel dual-core CPU as a single processor but uses other metrics for other types, especially for processors with more than two cores.[16]

6.5 Embedded applications

Embedded computing operates in an area of processor technology distinct from that of “mainstream” PCs. The same technological drivers towards multi-core apply here too. Indeed, in many cases the application is a “natural” fit for multi-core technologies, if the task can easily be partitioned between the different processors.

In addition, embedded software is typically developed for a specific hardware release, making issues of software portability, legacy code or supporting independent developers less critical than is the case for PC or enterprise computing. As a result, it is easier for developers to adopt new technologies and as a result there is a greater variety of multi-core processing architectures and suppliers.

As of 2010, multi-core network processing devices have become mainstream, with companies such as Freescale Semiconductor, Cavium Networks, Wintegra and Broadcom all manufacturing products with eight processors. For the system developer, a key challenge is how to exploit all the cores in these devices to achieve maximum networking performance at the system level, despite the performance limitations inherent in an SMP operating system. To address this issue, companies such as 6WIND provide portable packet processing software designed so that the networking data plane runs in a fast path environment outside the OS, while retaining full compatibility with standard OS APIs.[17]

In digital signal processing the same trend applies: Texas Instruments has the three-core TMS320C6488 and four-core TMS320C5441, Freescale the four-core MSC8144 and six-core MSC8156 (and both have stated they are working on eight-core successors). Newer entries include the Storm-1 family from Stream Processors, Inc with 40 and 80 general purpose ALUs per chip, all programmable in C as a SIMD engine and Picochip with three-hundred processors on a single die, focused on communication applications.

6.6 Hardware examples

6.6.1 Commercial

- Adapteva Epiphany, a many-core processor architecture which allows up to 4096 processors on-chip, although only a 16 core version has been commercially produced.

- Aeroflex Gaisler LEON3, a multi-core SPARC that also exists in a fault-tolerant version.

- Ageia PhysX, a multi-core physics processing unit.

- Ambric Am2045, a 336-core Massively Parallel Processor Array (MPPA)

- AMD

  - A-Series, dual-, triple-, and quad-core of Accelerated Processor Units (APU).
  - Athlon 64, Athlon 64 FX and Athlon 64 X2 family, dual-core desktop processors.
  - Athlon II, dual-, triple-, and quad-core desktop processors.
  - FX-Series, quad-, 6-, and 8-core desktop processors.
  - Opteron, dual-, quad-, 6-, 8-, 12-, and 16-core server/workstation processors.
  - Phenom, dual-, triple-, and quad-core processors.
  - Phenom II, dual-, triple-, quad-, and 6-core desktop processors.
  - Sempron X2, dual-core entry level processors.
  - Turion 64 X2, dual-core laptop processors.
  - Radeon and FireStream multi-core GPU/GPGPU (10 cores, 16 5-issue wide superscalar stream processors per core)

- Analog Devices Blackfin BF561, a symmetrical dual-core processor
- ARM MPCore is a fully synthesizable multi-core container for ARM11 MPCore and ARM Cortex-A9 MPCore processor cores, intended for high-performance embedded and entertainment applications.
- ASOCS ModemX, up to 128 cores, wireless applications.
- Azul Systems
  - Vega 1, a 24-core processor, released in 2005.
  - Vega 2, a 48-core processor, released in 2006.
  - Vega 3, a 54-core processor, released in 2008.
- Broadcom SiByte SB1250, SB1255, SB1455; BCM 2836 quad-core ARM SoC (designed for the Raspberry Pi 2)
- ClearSpeed
  - CSX700, 192-core processor, released in 2008 (32/64-bit floating point; Integer ALU)
- Cradle Technologies CT3400 and CT3600, both multi-core DSPs.
- Cavium Networks Octeon, a 32-core MIPS MPU.
- Freescale Semiconductor QorIQ series processors, up to 8 cores, Power Architecture MPU.
- Hewlett-Packard PA-8800 and PA-8900, dual core PA-RISC processors.
- IBM
  - POWER4, a dual-core processor, released in 2001.
  - POWER5, a dual-core processor, released in 2004.
  - POWER6, a dual-core processor, released in 2007.
  - POWER7, a 4,6,8-core processor, released in 2010.
  - POWER8, a 12-core processor, released in 2013.
  - PowerPC 970MP, a dual-core processor, used in the Apple Power Mac G5.
  - Xenon, a triple-core, SMT-capable, PowerPC microprocessor used in the Microsoft Xbox 360 game console.
- Kalray
  - MPPA-256, 256-core processor, released 2012 (256 usable VLIW cores, Network-on-Chip (NoC), 32/64-bit IEEE 754 compliant FPU)
- Sony/IBM/Toshiba's Cell processor, a nine-core processor with one general purpose PowerPC core and eight specialized SPUs (Synergystic Processing Unit) optimized for vector operations used in the Sony PlayStation 3
- Infineon Danube, a dual-core, MIPS-based, home gateway processor.
- Intel
  - Atom, single and dual-core processors for netbook systems.
  - Celeron Dual-Core, the first dual-core processor for the budget/entry-level market.
  - Core Duo, a dual-core processor.
  - Core 2 Duo, a dual-core processor.
  - Core 2 Quad, 2 dual-core dies packaged in a multi-chip module.
  - Core i3, Core i5 and Core i7, a family of multi-core processors, the successor of the Core 2 Duo and the Core 2 Quad.
  - Itanium 2, a dual-core processor.
  - Pentium D, 2 single-core dies packaged in a multi-chip module.
  - Pentium Extreme Edition, 2 single-core dies packaged in a multi-chip module.
  - Pentium Dual-Core, a dual-core processor.
  - Teraflops Research Chip (Polaris), a 3.16 GHz, 80-core processor prototype, which the company originally stated would be released by 2011.[18]

  - Xeon dual-, quad-, 6-, 8-, 10- and 15-core processors.[19]
  - Xeon Phi 57-core, 60-core and 61-core processors.
- IntellaSys
  - SEAforth 40C18, a 40-core processor[20]
  - SEAforth24, a 24-core processor designed by Charles H. Moore
- NetLogic Microsystems
  - XLP, a 32-core, quad-threaded MIPS64 processor
  - XLR, an eight-core, quad-threaded MIPS64 processor
  - XLS, an eight-core, quad-threaded MIPS64 processor
- Nvidia
  - GeForce 9 multi-core GPU (8 cores, 16 scalar stream processors per core)
  - GeForce 200 multi-core GPU (10 cores, 24 scalar stream processors per core)
  - Tesla multi-core GPGPU (10 cores, 24 scalar stream processors per core)
- Parallax Propeller P8X32, an eight-core microcontroller.
- picoChip PC200 series 200–300 cores per device for DSP & wireless
- Plurality HAL series tightly coupled 16-256 cores, L1 shared memory, hardware synchronized processor.
- Rapport Kilocore KC256, a 257-core microcontroller with a PowerPC core and 256 8-bit “processing elements”.
- SiCortex “SiCortex node” has six MIPS64 cores on a single chip.
- Sun Microsystems
  - MAJC 5200, two-core VLIW processor
  - UltraSPARC IV and UltraSPARC IV+, dual-core processors.
  - UltraSPARC T1, an eight-core, 32-thread processor.
  - UltraSPARC T2, an eight-core, 64-concurrent-thread processor.
  - UltraSPARC T3, a sixteen-core, 128-concurrent-thread processor.
  - SPARC T4, an eight-core, 64-concurrent-thread processor.
  - SPARC T5, a sixteen-core, 128-concurrent-thread processor.
- Texas Instruments
  - TMS320C80 MVP, a five-core multimedia video processor.
  - TMS320TMS320C66, 2,4,8 core dsp.
- Tilera
  - TILE64, a 64-core 32-bit processor
  - TILE-Gx, a 72-core 64-bit processor
- XMOS Software Defined Silicon quad-core XS1-G4

6.6.2 Free

- OpenSPARC

6.6.3 Academic

- MIT, 16-core RAW processor
- University of California, Davis, Asynchronous array of simple processors (AsAP)
  - 36-core 610 MHz AsAP
  - 167-core 1.2 GHz AsAP2
- University of Washington, Wavescalar processor
- University of Texas, Austin, TRIPS processor
- Linköping University, Sweden, ePUMA processor

6.7 Benchmarks

The research and development of multicore processors often compares many options, and benchmarks are developed to help such evaluations. Existing benchmarks include SPLASH-2, PARSEC, and COSMIC for heterogeneous systems.[21]

6.8 Notes

1. ^ Digital signal processors (DSPs) have used multi-core architectures for much longer than high-end general-purpose processors. A typical example of a DSP-specific implementation would be a combination of a RISC CPU and a DSP MPU. This allows for the design of products that require a general-purpose processor for user interfaces and a DSP for real-time data processing; this type of design is common in mobile phones. In other applications, a growing number of companies have developed

multi-core DSPs with very large numbers of processors.

2. ^ Two types of operating systems are able to use a dual-CPU multiprocessor: partitioned multiprocessing and symmetric multiprocessing (SMP). In a partitioned architecture, each CPU boots into separate segments of physical memory and operate independently; in an SMP OS, processors work in a shared space, executing threads within the OS independently.

6.9 See also

- Race condition
- Multicore Association
- Hyper-threading
- Multitasking
- PureMVC MultiCore – a modular programming framework
- XMTC
- Parallel Random Access Machine
- Partitioned global address space (PGAS)
- Thread
- CPU shielding
- GPGPU
- CUDA
- OpenCL (Open Computing Language) – a framework for heterogeneous execution
- Ateji PX – an extension of the Java language for parallelism
- BMDFM (Binary Modular Dataflow Machine) – Multi-core Runtime Environment

6.10 References

[1] Margaret Rouse (March 27, 2007). "Definition: multi-core processor". TechTarget. Retrieved March 6, 2013.

[2] CSA Organization

[3] "Rockwell R65C00/21 Dual CMOS Microcomputer and R65C29 Dual CMOS Microprocessor". Rockwell International. October 1984.

[4] "Rockwell 1985 Data Book". Rockwell International Semiconductor Products Division. January 1985.

[5] Aater Suleman (May 20, 2011). "What makes parallel programming hard?". FutureChips. Retrieved March 6, 2013.

[6] Programming Many-Core Chips. By András Vajda, page 3

[7] Ryan Shrout (December 2, 2009). "Intel Shows 48-core x86 Processor as Single-chip Cloud Computer". Retrieved March 6, 2013.

[8] "Intel unveils 48-core cloud computing silicon chip". BBC. December 3, 2009. Retrieved March 6, 2013.

[9] Aater Suleman (May 19, 2011). "Q & A: Do multicores save energy? Not really.". Retrieved March 6, 2013.

[10] Ni, Jun. "Multi-core Programming for Medical Imaging". Retrieved 17 February 2013.

[11] Rick Merritt (February 6, 2008). "CPU designers debate multi-core future". EE Times. Retrieved March 6, 2013.

[12] Multicore packet processing Forum

[13] John Darlinton, Moustafa Ghanem, Yike Guo, Hing Wing To (1996), "Guided Resource Organisation in Heterogeneous Parallel Computing", *Journal of High Performance Computing* **4** (1): 13–23

[14] Multicore Processor Licensing

[15] Compare: "Multi-Core Processor Licensing". *download.microsoft.com*. Microsoft Corporation. 2004-10-19. p. 1. Retrieved 2015-03-05. On October 19, 2004, Microsoft announced that our server software that is currently licensed on a per-processor model will continue to be licensed on a per-processor, and not on a per-core, model.

[16] Compare: "The Licensing Of Oracle Technology Products". OMT-CO Operations Management Technology Consulting GmbH. Retrieved 2014-03-04.

[17] Maximizing network stack performance

[18] 80-core prototype from Intel

[19] 15 core Xeon

[20] "40-core processor with Forth-based IDE tools unveiled"

[21] "COSMIC Heterogeneous Multiprocessor Benchmark Suite"

6.11 External links

- What Is A Processor Core?
- Embedded moves to multicore
- Multicore News blog
- IEEE: Multicore Is Bad News For Supercomputers

Chapter 7

Intel Atom (CPU)

This article is about the netbook and MID version of Atom. It is not to be confused with the Atom (system on chip) for smartphones and tablets.

Intel Atom is the brand name for a line of ultra-low-voltage IA-32 and x86-64 CPUs (or microprocessors) from Intel, originally designed in 45 nm complementary metal–oxide–semiconductor (CMOS) with subsequent models, codenamed *Cedar*, using a 32 nm process.[2] Atom is mainly used in netbooks, nettops, embedded applications ranging from health care to advanced robotics, and mobile Internet devices (MIDs).

Atom processors are based on the Bonnell microarchitecture.[3][4] On 21 December 2009, Intel announced the *Pine Trail* platform, including new Atom processor code-named *Pineview* (Atom N450), with total kit power consumption down 20%.[5] On 28 December 2011, Intel updated the Atom line with the *Cedar* processors.[2]

In December 2012, Intel launched the 64-bit *Centerton* family of Atom CPUs, designed specifically for use in servers.[6] *Centerton* adds features previously unavailable in Atom processors, such as Intel VT virtualization technology and support for ECC memory.[7] On 4 September 2013 Intel launched a 22 nm successor to *Centerton*, codenamed *Avoton*.[8]

In 2012, Intel announced a new system on chip (SoC) platform designed for smartphones and tablets which would use the Atom line of CPUs.[9] It is a continuation of the partnership announced by Intel and Google on 13 September 2011 to provide support for the Android operating system on Intel x86 processors.[10] This range competes with existing SoCs developed for the smartphone and tablet market from companies like Texas instruments, Nvidia, Qualcomm and Samsung.[11]

7.1 History

Intel Atom is a direct successor of the Intel A100 and A110 low-power microprocessors (code-named Stealey), which were built on a 90 nm process, had 512 kB L2 cache and ran at 600 MHz/800 MHz with 3 W TDP (Thermal Design Power). Prior to the Silverthorne announcement, outside sources had speculated that Atom would compete with AMD's Geode system-on-a-chip processors, used by the One Laptop per Child (OLPC) project, and other cost and power sensitive applications for x86 processors. However, Intel revealed on 15 October 2007 that it was developing another new mobile processor, codenamed Diamondville, for OLPC-type devices.[12]

“Atom” was the name under which Silverthorne would be sold, while the supporting chipset formerly code-named Menlow was called Centrino Atom.[13]

At Spring Intel Developer Forum (IDF) 2008 in Shanghai, Intel officially announced that Silverthorne and Diamondville are based on the same microarchitecture. Silverthorne would be called the Atom Z5xx series and Diamondville would be called the Atom N2xx series. The more expensive lower-power Silverthorne parts will be used in Intel mobile Internet devices (MIDs) whereas Diamondville will be used in low-cost desktop and notebooks. Several Mini-ITX motherboard samples have also been revealed.[14] Intel and Lenovo also jointly announced an Atom powered MID called the IdeaPad U8.[15]

In April 2008, a MID development kit was announced by Sophia Systems[16] and the first board called CoreExpress-ECO was revealed by a German company LiPPERT Embedded Computers, GmbH.[17] Intel offers Atom based motherboards.[18][19]

In December 2012, Intel released Atom for servers, the S1200 series. The primary difference between these processors and all prior versions, is that ECC memory support has been added, enabling the use of the Atom in mission-critical server environments that demand redundancy and memory failure protection.

7.2 Instruction set architecture

7.2.1 32-bit and 64-bit hardware support

All Atom processors implement the x86 (IA-32) instruction set; however, support for the Intel 64 instruction set was not added until the desktop *Diamondville* and desktop and mobile *Pineview* cores. The Atom N2xx and Z5xx series Atom models cannot run x86-64 code.[21] The *Centerton* server processors will support the Intel 64 instruction set.[7]

7.2.2 Intel 64 software support

Intel states the Atom supports 64-bit operation only "with a processor, chipset, BIOS" that all support Intel 64. Those Atom systems not supporting all of these cannot enable Intel 64.[22] As a result, the ability of an Atom-based system to run 64-bit versions of operating systems such as Ubuntu or Debian GNU/Linux may vary from one motherboard to another. Online retailer mini-itx.com has tested Atom-based motherboards made by Intel and Jetway, and while they were able to install 64-bit versions of Linux on Intel-branded motherboards with D2700 (Pineview) processors, Intel 64 support was not enabled on a Jetway-branded motherboard with a D2550 (Pineview) processor.[23]

Even among Atom-based systems which have Intel 64 enabled, not all are able to run 64-bit versions of Microsoft Windows. For those *Pineview* processors which support 64-bit operation, Intel Download Center currently provides 64-bit Windows 7 and Windows Vista drivers for Intel GMA 3150 graphics, found in *Pineview* processors.[24] However, no 64-bit Windows drivers are available for Intel Atom *Cedarview* processors, released Q3 2011.[25] However, Intel's Bay Trail-M processors, built on the Silvermont microarchitecture and released in the second half of 2013, regain 64-bit support, although driver support for Linux and Windows 7 is limited at launch.[26]

The lack of 64-bit Windows support for *Cedarview* processors appears to be due to a driver issue. A member of the Intel Enthusiast Team has stated in a series of posts on enthusiast site Tom's Hardware that while the Atom D2700 (Pineview) was designed with Intel 64 support, due to a "limitation of the board" Intel had pulled their previously-available 64-bit drivers for Windows 7 and would not provide any further 64-bit support.[27] Some system manufacturers have similarly stated that their motherboards with Atom Cedarview processors lack 64-bit support due to a "lack of Intel® 64-bit VGA driver support".[28] Because all *Cedarview* processors use the same Intel GMA 3600 or 3650 graphics as the D2700, this indicates that Atom Cedarview systems will remain unable to run 64-bit versions of Windows, even those which have Intel 64 enabled and are able to run 64-bit versions of Linux.

7.3 Availability

Atom processors became available to system manufacturers in 2008. Because they are soldered, like northbridges and southbridges, onto a mainboard, Atom processors are not available to home users or system builders as separate processors, although they may be obtained preinstalled on some ITX motherboards. The Diamondville and Pineview[29] Atom is used in the HP Mini Series, aigo MID Asus N10, Lenovo IdeaPad S10, Acer Aspire One & Packard Bell's "dot" (ZG5), recent ASUS Eee PC systems, Sony VAIO M-series, AMtek Elego, Dell Inspiron Mini Series, Gigabyte M912, LG X Series, Samsung NC10, Sylvania g Netbook Meso, Toshiba NB series (100, 200, 205, 255, 300, 500, 505), MSI Wind PC netbooks, RedFox Wizbook 1020i, Sony Vaio X Series, Zenith Z-Book, a range of Aleutia desktops, Magic W3 and the Archos.The Pineview line is also used in multiple AAC devices for the disabled individual who is unable to speak and the AAC device assists the user in everyday communication with dedicated speech software.

7.4 Performance

The performance of a single core Atom is about half that of a Pentium M of the same clock rate. For example, the Atom N270 (1.60 GHz) found in many netbooks such as the Eee PC can deliver around 3300 MIPS and 2.1 GFLOPS in standard benchmarks,[30] compared to 7400 MIPS and 3.9 GFLOPS for the similarly clocked (1.73 GHz) Pentium M 740.[31]

The Pineview platform has proven to be only slightly faster than the previous Diamondville platform. This is because the Pineview platform uses the same Bonnell execution core as Diamondville and is connected to the memory controller via the FSB, hence memory latency and performance in CPU-intensive applications are minimally improved.[32]

7.5 Bonnell microarchitecture

Main article: Bonnell (microarchitecture)

Intel Atom processors are based on the *Bonnell* microarchitecture,[3][4] which can execute up to two instructions per cycle. Like many other x86 microprocessors, it translates x86-instructions (CISC instructions) into simpler internal operations (sometimes referred to as micro-ops, i.e., effectively RISC style instructions) prior to execution. The majority of instructions produce one micro-op when translated, with around 4% of instructions used in typical programs producing multiple micro-ops. The number of instructions that produce more than one micro-op is significantly fewer than the P6 and NetBurst microarchitectures. In the Bonnell microarchitecture, internal micro-ops can contain both a memory load and a memory store in connection with an ALU operation, thus being more similar to the x86 level and more powerful than the micro-ops used in previous designs.[33] This enables relatively good performance with only two integer ALUs, and without any instruction reordering, speculative execution, or register renaming. The Bonnell microarchitecture therefore represents a partial revival of the principles used in earlier Intel designs such as P5 and the i486, with the sole purpose of enhancing the performance per watt ratio. However, Hyper-Threading is implemented in an easy (i.e., low power) way to employ the whole pipeline efficiently by avoiding the typical single thread dependencies.[33]

7.6 Collaborations

In March 2009, Intel announced that it would be collaborating with TSMC for the production of the Atom processors.[34] The deal was put on hold due to lack of demand in 2010.

On 13 September 2011 Intel and Google held a joint announcement of a partnership to provide support in Google's Android operating system for Intel processors (beginning with the Atom). This would allow Intel to supply chips for the growing smartphone and tablet market.[35]

7.7 Competition

Embedded processors based on the ARM version 7 instruction set architecture (such as Nvidia's Tegra 3 series, TI's 4 series and Freescale's i.MX51 based on the Cortex-A8 core, or the Qualcomm Snapdragon and Marvell Armada 500/600 based on custom ARMv7 implementations) offer similar performance to the low end Atom chipsets but at roughly one quarter the power consumption, and (like most ARM systems) as a single integrated system on a chip, rather than a two chip solution like the current Atom line. Although the second-generation Atom codenamed "Pineview" should greatly increase its competitiveness in performance/watt, ARM plans to counter the threat with the multi-core capable Cortex-A9 core as used in Nvidia's Tegra 2/3, TI's OMAP 4 series, and Qualcomm's next-generation Snapdragon series, among others.

The Nano and Nano Dual-Core series from VIA is slightly above the average thermal envelope of the Atom, but offers hardware AES support, random number generators, and out-of-order execution. Performance comparisons of the Intel Atom against the Via Nano indicate that a single core Intel Atom is easily outperformed by the Via Nano which is in turn outperformed by a dual core Intel Atom 330 in tests where multithreading is used. The Core 2 Duo SU7300 outperforms the dual-core Nano.[36][37][38][39][40][41][42][43]

The Xcore86 (also known as the PMX 1000) is x586 based System on Chip (SoC) that offers a below average thermal envelope compared to the Atom.

Kenton Williston of EE Times said that while Atom will not displace ARM from its current markets, the ability to apply the PC architecture into smaller, cheaper and lower power form factors will open up new markets for Intel.[44]

ARM has found that Intel's Atom processors offer less compatibility and lower performance than their chips when running Android, and higher power consumption and less battery life for the same tasks under both Android and Windows.[45]

Even AMD is in this competition with the Mullins brand based on Puma Microarchitecture who offers better Computing and even better Graphics performance with similar thermal power.

7.8 See also

- List of Intel Atom microprocessors
- Intel Edison
- Intel Quark

7.9 Notes

[1] "Intel® Atom™ Processor Z520". Intel. Archived from the original on 2011-07-04.

[2] Anand Lal Shimpi. "Intel's Atom N2600, N2800 & D2700: Cedar Trail, The Heart of the 2012 Netbook". Archived from the original on 2014-04-29. Retrieved 28 December 2011.

[3] Jeff Moriarty (1 April 2008). "'Atom 101' - Deciphering the Intel codewords around MIDs". Archived from the original on 2012-03-27. Retrieved 4 August 2010.

[4] Anand Lal Shimpi (27 January 2010). "Why Pine Trail Isn't Much Faster Than the First Atom". Archived from the original on 2014-01-04. Retrieved 4 August 2010.

[5] "Intel Announces Next-Generation Atom Platform". Intel. Archived from the original on 2013-06-06.

[6] "Products (Formerly Centerton)". Archived from the original on 2013-10-14. Retrieved 22 March 2013.

[7] Ryan Smith (11 December 2012). "Intel Launches Centerton Atom S1200 Family, First Atom for Servers". Archived from the original on 2014-05-02. Retrieved 22 March 2013.

[8] Inside Intel's Atom C2000-series 'Avoton' processors Archived February 9, 2014 at the Wayback Machine

[9] Intel Raises Bar on Smartphones, Tablets and Ultrabook™ Devices

[10] Antara News : Intel, Google announce partnership for Android smartphones

[11] Sadauskas, Andrew (30 April 2012). "Intel battles ARM with new handset". *smartcompany.com.au*. Retrieved 29 May 2012.

[12] "Intel to unveil OLPC chips in Shanghai next April". *InfoWorld*. 15 October 2007. Archived from the original on 2012-03-11.

[13] "Intel Announces Atom Brand for Silverthorne, Menlow". PC World. Archived from the original on 2008-07-09.

[14] "Intel Developer Forum Spring 2008: Day 1 – Hardware Upgrade". Hwupgrade.it. 30 July 2005. Archived from the original on 2012-01-12. Retrieved 4 April 2010.

[15] "Lenovo exhibits Atom based MID Ideapad U8 at IDF 2008 : Specs, reviews and prices". Archived from the original on 2012-02-23.

[16] "MID dev kit sports Centrino Atom chipset". Archived from the original on 2009-03-02. Retrieved 29 January 2011.

[17] "Tiny Centrino Atom-based module unveiled". Archived from the original on 2009-04-27. Retrieved 29 January 2011.

[18] "Intel Desktop Board D945GCLF – Overview". Archived from the original on 2008-08-21. Retrieved 29 January 2011.

[19] "Intel offers $80 "Little Falls" Atom mobo". Archived from the original on 2009-02-16. Retrieved 29 January 2011.

[20] "Products: SPECIFICATIONS: Intel® Atom™ Processor".

[21] "Intel Atom Processor Specifications". Intel.com. Archived from the original on 2011-03-17. Retrieved 4 April 2010.

[22] "Intel N2600 : Atom Benchmarked: 4W Of Performance". Intel.com. 28 August 2012. Archived from the original on 2014-04-21. Retrieved 28 August 2012.

[23] "mini-itx.com - store - Intel Atom Mini-ITX boards". mini-itx.com. Archived from the original on 2013-06-13. Retrieved 4 March 2013.

[24] "Download Center". Intel.com. Archived from the original on 2014-03-18. Retrieved 4 March 2013.

[25] "Logic Supply Cedar View". logicsupply.com. Archived from the original on 2013-10-26. Retrieved 4 March 2013.

[26] "Logic Supply Bay Trail Offers Performance Boost". logicsupply.com. Archived from the original on 2014-03-17. Retrieved 17 March 2013.

[27] "[Solved] Atom D2700 (Cedar Trail) 32 bit?". tomshardware.com. 10 February 2012. Retrieved 4 March 2013.

[28] "ASRock > AD2700B-ITX". asrock.com. Retrieved 4 March 2013.

[29] "HP Mini 210-2072cl PC Product Specifications". Archived from the original on 2014-01-08.

[30] "SiSoft Sandra : Atom Benchmarked: 4W Of Performance". Tomshardware.com. 29 July 2008. Retrieved 4 April 2010.

[31] "Intel Pentium M 740 PCSTATS Review - Benchmarks: Office Productivity, SiSoft Sandra 2005". PCstats.com. Archived from the original on 2013-10-29.

[32] "Why Pine Trail Isn't Much Faster Than the First Atom". AnandTech. Archived from the original on 2010-02-01. Retrieved 4 April 2010.

[33] "Intel's Atom Architecture: The Journey Begins". AnandTech. Archived from the original on 2009-05-31. Retrieved 4 April 2010.

[34] "TSMC To Build Intel's Atom-Based Chips". Forbes. 2 March 2009. Archived from the original on 2012-10-27. Retrieved 3 March 2009.

[35] "Intel, Google announce partnership for Android smartphones". 14 September 2011. Archived from the original on 2013-12-04.

[36] "Intel Atom vs. VIA Nano Platform Comparo Introduction". TweakTown. 11 August 2008. Archived from the original on 2014-04-13. Retrieved 4 April 2010.

[37] "VIA Nano Dual Core Preview". 26 December 2010. Archived from the original on 2014-04-13. Retrieved 26 December 2010.

[38] Kyle Bennett. "Introduction & Power - Intel Atom vs.VIA Nano". Hardocp.com. Archived from the original on 2012-02-19. Retrieved 4 April 2010.

[39] "VIA Nano vs Intel Atom". TrustedReviews. Archived from the original on 2009-09-05. Retrieved 4 April 2010.

[40] "VIA Nano Outperforms Intel Atom in Actual Industry Performance Benchmarking tests". Mydigitallife.info. 31 July 2008. Archived from the original on 2010-01-02. Retrieved 4 April 2010.

[41] "Intel Atom Initial Benchmarking Data vs. Pentium and Celeron M Processors Before Official Release". Mydigitallife.info. 8 March 2008. Archived from the original on 2011-04-08. Retrieved 4 April 2010.

[42] "EEE PC vs MSI Wind - Atom vs Celeron CPU Performance Benchmark: Netbooks, EEE PC, MSI Wind, Aspire One and Akoya Resources". Eeejournal.com. 11 May 2008. Archived from the original on 2014-04-13. Retrieved 4 April 2010.

[43] "Intel Atom 230/330/VIA Nano performances contrasted". En.hardspell.com. 25 September 2008. Archived from the original on 2008-12-20. Retrieved 4 April 2010.

[44] "Analysis: The real scoop on Atom-ARM rivalry". Archived from the original on 2014-02-15. Retrieved 1 January 2012.

[45] Myslewski, Rik (2 May 2014). "ARM tests: Intel flops on Android compatibility, Windows power". *www.theregister.co.uk* (The Register). Archived from the original on 2014-05-03. Retrieved 2 May 2014.

7.10 References

- “Intel cranks 45nm ultramobile CPU”. *EE Times*. 18 April 2007. Retrieved 28 October 2007.
- “Intel reaches back in time for its ultralow power chips”. 28 January 2008. Retrieved 29 January 2008.
- “New Intel Centrino Atom Processor Technology Ushers in 'Best Internet Experience in Your Pocket'". 2 April 2008.
- “Intel’s Atom Architecture: The Journey Begins”. 2 April 2008.
- “The Battle of Low-Power Processors: Best Choice for a Nettop”. 27 September 2008.
- linuxdevices.com - Intel announces first Atom chips
- hardwaresecrets.com - Inside Atom Architecture
- computermonger.com - Intel Atom N280 vs N270 Benchmarked
- LinuxTECH.NET - Intel Pineview Atom based Motherboards Complete Overview

7.11 External links

- Intel - Intel Atom Processor Overview
- Intel Atom Processor
- Intel - Intel Atom Processor Family

Chapter 8

Intel Core

This article is about the Intel processor brand name. For the Intel microarchitecture that is the basis for the Core 2 processor family, see Intel Core (microarchitecture).

Intel Core is a brand name that Intel uses for various mid-range to high-end consumer and business microprocessors. These processors replaced the then-currently mid to high end Pentium processors, making them entry level, and bumping the Celeron series of processors to low end. Similarly, identical or more capable versions of Core processors are also sold as Xeon processors for the server and workstation market.

As of 2015 the current lineup of Core processors included the latest Intel Core i7, Intel Core i5, and Intel Core i3.[1]

8.1 Overview

Clock speed slowest 1.2 GHz to the fastest 4.0 GHz (Intel Core i7-4790K) (or 4.4 GHz via Intel Turbo Boost Technology)[3]

8.2 Enhanced Pentium M based

Main article: Enhanced Pentium M (microarchitecture)
For details about the processor core, see Yonah (microprocessor).

The original *Core* brand refers to Intel's 32-bit mobile dual-core x86 CPUs, which derived from the Pentium M branded processors. The processor family used a more enhanced version of the Intel P6 microarchitecture. It emerged in parallel with the NetBurst microarchitecture (Intel P68) of the Pentium 4 brand, and was a precursor of the 64-bit Core microarchitecture of Core 2 branded CPUs. The Core brand comprised two branches: the *Duo* (dual-core) and *Solo* (Duo with one disabled core, which replaced the Pentium M brand of single-core mobile processor).

Intel launched the Core brand on January 6, 2006 with the release of the 32-bit *Yonah* CPU – Intel’s first dual-core mobile (low-power) processor. Its dual-core layout closely resembled two interconnected Pentium M branded CPUs packaged as a single die (piece) silicon chip (IC). Hence, the 32-bit microarchitecture of Core branded CPUs – contrary to its name – had more in common with Pentium M branded CPUs than with the subsequent 64-bit Core microarchitecture of Core 2 branded CPUs. Despite a major rebranding effort by Intel starting January 2006, some companies continued to market computers with the Yonah core marked as Pentium M.

The Core series is also the first Intel processor used as the main CPU in an Apple Macintosh computer. The Core Duo was the CPU for the first generation MacBook Pro, while the Core Solo appeared in Apple’s Mac mini line. Core Duo signified the beginning of Apple’s shift to Intel processors across their entire line.

In 2007, Intel began branding the Yonah core CPUs intended for mainstream *mobile* computers as Pentium Dual-Core, not to be confused with the *desktop* 64-bit Core microarchitecture CPUs also branded as Pentium Dual-Core.

September 2007 and January 4, 2008 marked the discontinuation of a number of *Core* branded CPUs including several Core Solo, Core Duo, Celeron and one Core 2 Quad chip.[4][5]

8.2.1 Core Duo

Intel Core Duo[6] (product code 80539) consists of two cores on one die, a 2 MB L2 cache shared by both cores, and an arbiter bus that controls both L2 cache and FSB (front-side bus) access.

8.2.2 Core Solo

Intel Core Solo[7] (product code 80538) uses the same two-core die as the Core Duo, but features only one *active* core. Depending on demand, Intel may also simply disable one of the cores to sell the chip at the Core Solo price—this requires less effort than launching and maintaining a separate line of CPUs that physically only have one core. Intel used the same strategy previously with the 486 CPU in which early 486SX CPUs were in fact manufactured as 486DX CPUs but with the FPU disabled.

8.3 64-bit Core microarchitecture based

Main article: Core (microarchitecture)

The successor to Core is the mobile version of the Intel Core 2 line of processors using cores based upon the Intel **Core microarchitecture**,[8] released on July 27, 2006. The release of the mobile version of Intel Core 2 marks the reunification of Intel's desktop and mobile product lines as Core 2 processors were released for desktops and notebooks, unlike the first Intel Core CPUs that were targeted only for notebooks (although some small form factor and all-in-one desktops, like the iMac and the Mac Mini, also used Core processors).

Unlike the Intel Core, Intel Core 2 is a 64-bit processor, supporting Intel 64. Another difference between the original Core Duo and the new Core 2 Duo is an increase in the amount of Level 2 cache. The new Core 2 Duo has tripled the amount of on-board cache to 6 MB. Core 2 also introduced a quad-core performance variant to the single- and dual-core chips, branded Core 2 Quad, as well as an enthusiast variant, Core 2 Extreme. All three chips are manufactured at a 65 nm lithography, and in 2008, a 45 nm lithography and support Front Side Bus speeds ranging from 533 MHz to 1600 MHz. In addition, the 45 nm die shrink of the Core microarchitecture adds SSE4.1 support to all Core 2 microprocessors manufactured at a 45 nm lithography, therefore increasing the calculation rate of the processors.

8.3.1 Core 2 Solo

The **Core 2 Solo**,[9] introduced in September 2007, is the successor to the Core Solo and is available only as an ultra-low-power mobile processor with 5.5 Watt thermal design power. The original U2xxx series “Merom-L” used a special version of the Merom chip with CPUID number 10661 (model 22, stepping A1) that only had a single core and was also used in some Celeron processors. The later SU3xxx are part of Intel's CULV range of processors in a smaller µFC-BGA 956 package but contain the same Penryn chip as the dual-core variants, with one of the cores disabled during manufacturing.

8.3.2 Core 2 Duo

The majority of the desktop and mobile Core 2 processor variants are **Core 2 Duo**[10][11] with two processor cores on a single Merom, Conroe, Allendale, Penryn, or Wolfdale chip. These come in a wide range of performance and power consumption, starting with the relatively slow ultra-low-power Uxxxx (10 W) and low-power Lxxxx (17 W) versions, to the more performance oriented Pxxxx (25 W) and Txxxx (35 W) mobile versions and the Exxxx (65 W) desktop models. The mobile Core 2 Duo processors with an 'S' prefix in the name are produced in a smaller µFC-BGA 956 package, which allows building more compact laptops.

Within each line, a higher number usually refers to a better performance, which depends largely on core and front-side bus clock frequency and amount of second level cache, which are model-specific. Core 2 Duo processors typically use the full L2 cache of 2, 3, 4, or 6 MB available in the specific stepping of the chip, while versions with the amount of cache reduced during manufacturing are sold for the low-end consumer market as Celeron or Pentium Dual-Core processors. Like those processors, some low-end Core 2 Duo models disable features such as Intel Virtualization Technology.

8.3.3 Core 2 Quad

Core 2 Quad[12][13] processors are multi-chip modules consisting of two dies similar to those used in Core 2 Duo, forming a quad-core processor. This allows twice the performance of a dual-core processors at the same clock frequency in ideal conditions.

Initially, all Core 2 Quad models were versions of Core 2 Duo desktop processors, Kentsfield derived from Conroe and Yorkfield from Wolfdale, but later Penryn-QC was added as a high-end version of the mobile dual-core Penryn.

The Xeon 32xx and 33xx processors are mostly identical versions of the desktop Core 2 Quad processors and can be used interchangeably.

8.3.4 Core 2 Extreme

Core 2 Extreme processors[14][15] are enthusiast versions of Core 2 Duo and Core 2 Quad processors, usually with a higher clock frequency and an unlocked clock multiplier, which makes them especially attractive for overclocking. This is similar to earlier Pentium processors labeled as Extreme Edition. Core 2 Extreme processors were released at a much higher price than their regular version, often $999 or more.

8.4 Nehalem microarchitecture based

Main article: Nehalem (microarchitecture)

With the release of the **Nehalem microarchitecture** in November 2008,[16] Intel introduced a new naming scheme for its Core processors. There are three variants, Core i3, Core i5 and Core i7, but the names no longer correspond to specific technical features like the number of cores. Instead, the brand is now divided from low-level (i3), through mid-range (i5) to high-end performance (i7),[17] which correspond to three, four and five stars in Intel's Intel Processor Rating[18] following on from the entry-level Celeron (one star) and Pentium (two stars) processors.[19] Common features of all Nehalem based processors include an integrated DDR3 memory controller as well as QuickPath Interconnect or PCI Express and Direct Media Interface on the processor replacing the aging quad-pumped Front Side Bus used in all earlier Core processors. All these processors have 256 KB L2 cache per core, plus up to 12 MB shared L3 cache. Because of the new I/O interconnect, chipsets and mainboards from previous generations can no longer be used with Nehalem based processors.

8.4.1 Core i3

Intel intended the **Core i3** as the new low end of the performance processor line from Intel, following the retirement of the Core 2 brand.[20][21]

The first Core i3 processors were launched on January 7, 2010.[22]

The first Nehalem based Core i3 was Clarkdale-based, with an integrated GPU and two cores.[23] The same processor is also available as Core i5 and Pentium, with slightly different configurations.

The Core i3-3xxM processors are based on Arrandale, the mobile version of the Clarkdale desktop processor. They are similar to the Core i5-4xx series but running at lower clock speeds and without Turbo Boost.[24] According to an Intel FAQ they do not support Error Correction Code (ECC) memory.[25] According to motherboard manufacturer Supermicro, if a Core i3 processor is used with a server chipset platform such as Intel 3400/3420/3450, the CPU supports ECC with UDIMM.[26] When asked, Intel confirmed that, although the Intel 5 series chipset supports non-ECC memory only with the Core i5 or i3 processors, using those processors on a motherboard with 3400 series chipsets it supports the ECC function of ECC memory.[27] A limited number of motherboards by other companies also support ECC with Intel Core ix processors; the Asus P8B WS is an example, but it does not support ECC memory under Windows non-server operating systems.[28]

8.4.2 Core i5

The first **Core i5** using the Nehalem microarchitecture was introduced on September 8, 2009, as a mainstream variant of the earlier Core i7, the Lynnfield core.[29][30] Lynnfield Core i5 processors have an 8 MB L3 cache, a DMI bus running at 2.5 GT/s and support for dual-channel DDR3-800/1066/1333 memory and have Hyper-threading disabled. The same processors with different sets of features (Hyper-Threading and other clock frequencies) enabled are sold as Core i7-8xx and Xeon 3400-series processors, which should not be confused with high-end Core i7-9xx and Xeon 3500-series processors based on Bloomfield. A new feature called Turbo Boost Technology was introduced which maximizes speed for demanding applications, dynamically accelerating performance to match the workload.

The Core i5-5xx mobile processors are named **Arrandale** and based on the 32 nm Westmere shrink of the Nehalem microarchitecture. Arrandale processors have integrated graphics capability but only two processor cores. They were released in January 2010, together with Core i7-6xx and Core i3-3xx processors based on the same chip. The L3 cache in Core i5-5xx processors is reduced to 3 MB, while the Core i5-6xx uses the full cache and the Core i3-3xx does not support for Turbo Boost.[31] **Clarkdale**, the desktop version of Arrandale, is sold as Core i5-6xx, along with related Core i3 and Pentium brands. It has Hyper-Threading enabled and the full 4 MB L3 cache.[32]

According to Intel "Core i5 desktop processors and desktop boards typically do not support ECC memory",[33] but information on limited ECC support in the Core i3 section also applies to Core i5 and i7.

8.4.3 Core i7

Intel Core i7 as an Intel brand name applies to several families of desktop and laptop 64-bit x86-64 processors using the Nehalem, Westmere, Sandy Bridge, Ivy Bridge and Haswell microarchitectures. The Core i7 brand targets the business and high-end consumer markets for both desktop and laptop computers,[35] and is distinguished from the Core i3 (entry-level consumer), Core i5 (mainstream consumer), and Xeon (server and workstation) brands.

Intel introduced the Core i7 name with the Nehalem-based Bloomfield Quad-core processor in late 2008.[36][37][38][39] In 2009 new Core i7 models based on the Lynnfield (Nehalem-based) desktop quad-core processor and the Clarksfield (Nehalem-based) quad-core mobile were added,[40] and models based on the Arrandale dual-core mobile processor (also Nehalem-based) were added in January 2010. The first six-core processor in the Core lineup is the Nehalem-based Gulftown, which was launched on March 16, 2010. Both the regular Core i7 and the *Extreme Edition* are advertised as five stars in the Intel Processor Rating.

In each of the first three microarchitecture generations of the brand, Core i7 has family members using two distinct system-level architectures, and therefore two distinct sockets (for example, LGA 1156 and LGA 1366 with

Nehalem). In each generation, the highest-performing Core i7 processors use the same socket and QPI-based architecture as the low-end Xeon processors of that generation, while lower-performing Core i7 processors use the same socket and PCIe/DMI/FDI architecture as the Core i5.

“Core i7” is a successor to the Intel Core 2 brand.[41][42][43][44] Intel representatives stated that they intend the moniker *Core i7* to help consumers decide which processor to purchase as Intel releases newer Nehalem-based products in the future.[45]

8.5 Sandy Bridge microarchitecture based

Main article: Sandy Bridge

In early 2011, Intel introduced a new microarchitecture named **Sandy Bridge microarchitecture**. It kept all the existing brands from Nehalem, including Core i3/i5/i7, and introduced new model numbers. The initial set of Sandy Bridge processors includes dual- and quad-core variants, all of which use a single 32 nm die for both the CPU and integrated GPU cores, unlike the earlier microarchitectures. All Core i3/i5/i7 processors with the Sandy Bridge microarchitecture have a four-digit model number. With the mobile version, the thermal design power can no longer be determined from a one- or two-letter suffix but is encoded into the CPU number. Starting with Sandy Bridge, Intel no longer distinguishes the code names of the processor based on number of cores, socket or intended usage; they all use the same code name as the microarchitecture itself.

Ivy Bridge is the codename for Intel’s 22 nm die shrink of the Sandy Bridge microarchitecture based on tri-gate (“3D”) transistors, introduced in April 2012.

8.5.1 Core i3

Released on January 20, 2011, the Core i3-2xxx line of desktop and mobile processors is a direct replacement of the 2010 “Clarkdale” Core i3-5xx and “Arrandale” Core i3-3xxM models, based on the new microarchitecture. While they require new sockets and chipsets, the user-visible features of the Core i3 are largely unchanged, including the lack of support for Turbo Boost and AES-NI. Unlike the Sandy Bridge based Celeron and Pentium processors, the Core i3 line does support the new Advanced Vector Extensions. This particular processor is the entry-level processor of this new series of Intel processors.

8.5.2 Core i5

In January 2011, Intel released new quad-core Core i5 processors based on the “Sandy Bridge” microarchitecture at CES 2011. New dual-core mobile processors and desktop processors arrived in February 2011.

The Core i5-2xxx line of desktop processors are mostly quad-core chips, with the exception of the dual-core Core i5-2390T, and include integrated graphics, combining the key features of the earlier Core i5-6xx and Core i5-7xx lines. The suffix after the four-digit model number designates unlocked multiplier (K), low-power (S) and ultra-low-power (T).

The desktop CPUs now all have four non-SMT cores (like the i5-750), with the exception of the i5-2390T. The DMI bus is running at 5 GT/s.

The mobile Core i5-2xxxM processors are all dual-core chips like the previous Core i5-5xxM series and share most the features with that product line.

8.5.3 Core i7

The Core i7 brand remains the high-end for Intel’s desktop and mobile processors, featuring the Sandy Bridge models with the largest amount of L3 cache and the highest clock frequency. Most of these models are very similar to their smaller Core i5 siblings. The quad-core mobile Core i7-2xxxQM/XM processors follow the previous “Clarksfield” Core i7-xxxQM/XM processors, but now also include integrated graphics.

8.6 Ivy Bridge microarchitecture based

Main article: Ivy Bridge (microarchitecture)

8.6.1 Core i3

The Ivy Bridge based Core-i3-3xxx line is a minor upgrade to 22 nm process technology and better graphics.

8.6.2 Core i5

8.6.3 Core i7

8.7 Haswell microarchitecture based

Main article: Haswell (microarchitecture)

8.7.1 Core i3

8.7.2 Core i5

8.7.3 Core i7

8.8 Broadwell microarchitecture based

Main article: Broadwell (microarchitecture)

The Broadwell microarchitecture was released by Intel on September 6, 2014, and began shipping in late 2014. It is the first to use a 14 nm chip.[46] Additional, mobile processors were launched in January 2015. [47]

8.8.1 Core i3

8.8.2 Core i5

8.8.3 Core i7

8.8.4 Core M

8.9 See also

- Centrino

8.10 References

[1] “Desktop Processors”. Intel.com. Retrieved 2010-12-13.

[2] http://arstechnica.com/gadgets/2014/09/intels-launches-three-core-m-cpus-promises-more-broadwell-early-2015/

[3] “Intel Launches Devil’s Canyon and Overclockable Pentium: i7-4790K, i5-4690K and G3258”. Anandtech. 3 June 2014. Retrieved 29 June 2014.

[4] “Intel already phasing out first quad-core CPU”. TG Daily. Retrieved 2007-09-07.

[5] “Intel to discontinue older Centrino CPUs in Q1 08”. TG Daily. Retrieved 2007-10-01.

[6] “Support for the Intel Core Duo Processor”. Intel.com. Retrieved 2010-12-13.

[7] “Support for the Intel Core Solo processor”. Intel.com. Retrieved 2010-12-13.

[8] “Intel Microarchitecture”. Intel.com. Retrieved 2010-12-13.

[9] “Intel Core2 Solo Mobile Processor – Overview”. Intel.com. Retrieved 2010-12-13.

[10] “Intel Core2 Duo Processor: Upgrade Today”. Intel.com. Retrieved 2010-12-13.

[11] “Intel Core2 Duo Mobile Processor”. Intel.com. Retrieved 2010-12-13.

[12] “Intel Core2 Quad Processor Overview”. Intel.com. Retrieved 2010-12-13.

[13] “Intel Core2 Quad Mobile Processors – Overview”. Intel.com. Retrieved 2010-12-13.

[14] “Support for the Intel Core2 Extreme Processor”. Intel.com. Retrieved 2010-12-13.

[15] “Intel Core2 Extreme Processor”. Intel.com. Retrieved 2010-12-13.

[16] “Intel Microarchitecture Codenamed Nehalem”. Intel.com. Retrieved 2010-12-13.

[17] “Public Roadmap Desktop, Mobile & Data Center” (PDF). Intel.com. Retrieved 2010-12-13.

[18] “Intel Processor Ratings”. Intel.com. Retrieved 2011-07-21.

[19] “Processor Ratings”. Intel. 2010-07-09. Retrieved 2010-12-13.

[20] “Intel Quietly Announces Core i5 and Core i3 Branding”. AnandTech. Retrieved 2010-12-13.

[21] “Intel confirms Core i3 as 'entry-level' Nehalem chip”. Apcmag.com. 2009-09-14. Retrieved 2010-12-13.

[22] “Core i5 and i3 CPUs With On-Chip GPUs Launched”. Hardware.slashdot.org. 2010-01-04. Retrieved 2010-12-13.

[23] “Intel May Unveil Microprocessors with Integrated Graphics Cores at Consumer Electronics Show”. Xbitlabs.com. Retrieved 2010-12-13.

[24] “Intel to launch four Arrandale CPUs for mainstream notebooks in January 2010”. Digitimes.com. 2009-11-13. Retrieved 2010-12-13.

[25] Intel Core i3 desktop processor frequently asked questions

[26] Supermicro FAQ on ECC with Core i3

[27] Intel correspondence quoted on silentpcreview forum

[28] Asus P8B WS specification: supports “ECC, Non-ECC, un-buffered Memory”, but “Non-ECC, un-buffered memory only support for client OS (Windows 7, Vista and XP).”

[29] “Support for the Intel Core i5 Processor”. Intel.com. Retrieved 2010-12-13.

[30] Anand Lal Shimpi, *Intel’s Core i7 870 & i5 750, Lynnfield: Harder, Better, Faster Stronger*, anandtech.com

[31] http://www.digitimes.com/news/a20091113PD209.html

[32] Intel ⍰⍰⍰⍰ E5300(⍰) ⍰⍰|CPU ⍰⍰|⍰⍰⍰⍰⍰⍰ ⍰⍰Core i5⍰i3⍰⍰⍰⍰|IT168 diy⍰⍰

[33] Intel Core i5 desktop processor frequently asked questions

[34] Intel Core i5-430UM processor – CN80617006042AE

[35] “Support for the Intel Core i7 Processor”. Intel.com. Retrieved 2010-12-13.

[36] Modine, Austin (2008-11-18). “Intel celebrates Core i7 launch with Dell and Gateway”. The Register. Retrieved 2008-12-06.

[37] “IDF Fall 2008: Intel un-retires Craig Barrett, AMD sets up anti-IDF camp”. Tigervision Media. 2008-08-11. Retrieved 2008-08-11.

[38] “Meet the Bloggers”. Intel Corporation. Retrieved 2008-08-11.

[39] “Getting to the Core---Intel’s new flagship client brand”. Intel Corporation. Retrieved 2008-08-11.

[40] "[Intel Roadmap update] Nehalem to enter mainstream market”. ExpReview. 2008-06-10. Retrieved 2008-08-11.

[41] “Intel Details Upcoming New Processor Generations” (Press release). Intel Corporate. 2008-08-11.

[42] “Intel Core i7-920 Processor (8M Cache, 2.66 GHz, 4.80 GT/s Intel QPI)". Intel. Retrieved 2008-12-06.

[43] “Intel Core i7-940 Processor (8M Cache, 2.93 GHz, 4.80 GT/s Intel QPI)". Intel. Retrieved 2008-12-06.

[44] “Intel Core i7-965 Processor Extreme Edition (8M Cache, 3.20 GHz, 6.40 GT/s Intel QPI)". Intel. Retrieved 2008-12-06.

[45] Technology@Intel · Getting to the Core – Intel’s new flagship client brand

[46] “Intel Discloses Newest Microarchitecture and 14 Nanometer Manufacturing Process Technical Details”. *Intel.com*. Intel Corporation. 11 August 2014. Retrieved 6 September 2014.

[47] “Intel launched U-series Broadwell processors”. 10 January 2015. Retrieved 15 February 2015.

8.11 External links

- Intel Core technical specifications at the Wayback Machine (archived August 9, 2007)
- CPU Database. TechPowerUp.
- Intel Core Duo (Yonah) Performance Preview – Part II vs AMD 64 X2 and Intel Pentium M. Anandtech.
- Intel Core i7-3960X CPU Performance Comparison.
- Intel Centrino Duo Mobile Technology papers. Intel.
- Intel Product Information, providing a list of various processor generations

Chapter 9

List of Intel Core i5 microprocessors

The following is a list of Intel Core i5 brand microprocessors.

9.1 Desktop processors

9.1.1 Nehalem microarchitecture (1st generation)

“Lynnfield” (45 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Turbo Boost, Smart Cache.*
- FSB has been replaced with DMI.
- Transistors: 774 million
- Die size: 296 mm^2
- Stepping: B1

9.1.2 Westmere microarchitecture (1st generation)

"Clarkdale" (MCP, 32 nm dual-core)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), TXT, Intel VT-x, Intel VT-d, Hyper-Threading, Turbo Boost, AES-NI, Smart Cache.*
- Core i5-655K, Core i5-661 does not support *Intel TXT and Intel VT-d.*[1]
- Core i5-655K features an unlocked multiplier.
- FSB has been replaced with DMI.
- Contains 45 nm “Ironlake” GPU.
- Transistors: 382 million
- Die size: 81 mm^2
- Graphics Transistors: 177 million
- Graphics and Integrated Memory Controller die size: 114 mm^2
- Stepping: C2, K0

9.1.3 Sandy Bridge microarchitecture (2nd generation)

“Sandy Bridge” (dual-core, 32 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), TXT, Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost, AES-NI, Smart Cache, Intel Insider, vPro.*
- Transistors: 504 million
- Die size: 131 mm^2

“Sandy Bridge” (quad-core, 32 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), TXT, Intel VT-x, Intel VT-d, Turbo Boost, AES-NI, Smart Cache, Intel Insider, vPro.*
- All models support dual-channel DDR3-1333 RAM.
- Core i5-2300, Core i5-2310, Core i5-2320, Core i5-2380P, Core i5-2405S, Core i5-2450P, Core i5-2500K and Core i5-2550K does not support *Intel TXT, Intel VT-d, and Intel vPro.*[2]
- S processors feature lower-than-normal TDP (65W on 4-core models).
- T processors feature an even lower TDP (45W on 4-core models or 35W on 2-core models).
- K processors are unlockable and designed for overclocking. Other processors will have limited overclocking due to chipset limitations.[3]

- P processors disable the integrated graphics processor.
- Transistors: 1.16 billion[4]
- Die size: 216 mm^2

9.1.4 Ivy Bridge microarchitecture (3rd generation)

"Ivy Bridge" (dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost, AES-NI, Smart Cache, Intel Insider.*
- Die size: 93.6mm^2 or 118 mm^2 [5][6]

“Ivy Bridge” (quad-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Turbo Boost, AES-NI, Smart Cache, Intel Insider, vPro.*
- S processors feature lower-than-normal TDP (65 W on 4-core models).
- T processors feature an even lower TDP (45 W on 4-core models).
- K processors have unlocked turbo multiplier
- P processors disable the integrated graphics processor
- i5-3470, i5-3470S, i5-3475S, i5-3550, i5-3550S, i5 3570 and i5-3570T support *Intel TXT, Intel VT-d* and *vPro.*
- i5-3330, i5-3330S, and i5-3350P support *Intel VT-d.*
- Non-K processors will have limited turbo overclocking.
- Transistors: 1.4 billion
- Die size: 133 mm^2 or 160 mm^2

9.1.5 Haswell microarchitecture (4th generation)

“Haswell-DT” (dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost, AES-NI, Smart Cache, Intel Insider, vPro.*
- Transistors: 1.4 billion
- Die size: 177mm^2

“Haswell-DT” (quad-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Turbo Boost, AES-NI, Smart Cache, Intel Insider.*
- Intel Core i5-4430, Core i5-4430S, Core i5-4440, Core i5-4570, Core i5-4570S, Core i5-4670, Core i5-4670S, Core i5-4670T, Core i5-4690, Core i5-4690S, Core i5-4690K support *Intel VT-d.*
- Intel Core i5-4570, Core i5-4570S, Core i5-4670, Core i5-4670S, Core i5-4670T, Core i5-4590, Core i5-4690 support *vPro, Intel TSX, TXT.*
- Intel Core i5-4690K supports *Intel TSX* unlike its older counterpart, the i5-4670K.
- Transistors: 1.4 billion
- Die size: 185mm^2

“Haswell-H” (MCP, quad-core, 97 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Turbo Boost, AES-NI, Smart Cache, Intel Insider.*
- Core i5-4570R and Core i5-4670R also contain “Crystalwell": 128 MiB eDRAM built at (22 nm) acting as L4 cache
- Transistors: 1.4 billion
- Die size: 264mm^2 + 84mm^2

9.2 Mobile processors

9.2.1 Westmere microarchitecture (1st generation)

"Arrandale" (MCP, 32 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x,[9] Hyper-Threading, Turbo Boost, Smart Cache.*
- i5-5xx series (i5-520M, i5-520E, i5-540M, i5-560M, i5-580M, i5-520UM, i5-540UM, i5-560UM) supports *AES-NI, TXT and Intel VT-d.* [10]
- FSB has been replaced with DMI.
- Contains 45 nm "Ironlake" GPU.
- Transistors: 382 million
- Die size: 81 mm^2
- Graphics Transistors: 177 million
- Graphics and Integrated Memory Controller die size: 114 mm^2
- Stepping: C2, K0
- Core i5-520E has support for ECC memory and PCI express port bifurcation.

9.2.2 Sandy Bridge microarchitecture (2nd generation)

"Sandy Bridge" (32 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Hyper-threading, Turbo Boost, AES-NI, Smart Cache.*
- All models except Core i5-24xxM support TXT and Intel VT-d.
- Core i5-2430M/i5-2435M and i5-2410M/i5-2415M can support *AES-NI* with laptop OEM-supplied BIOS processor configuration update.[11]
- Core i5-2515E has support for ECC memory.
- Transistors: 624 million
- Die size: 149 mm^2

9.2.3 Ivy Bridge microarchitecture (3rd generation)

"Ivy Bridge" (22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost, AES-NI, Smart Cache.*
- i5-3320M, i5-3360M, i5-3427U, i5-3437U, i5-3439Y, and i5-3610ME support *TXT* and *vPro*.

9.2.4 Haswell microarchitecture (4th generation)

"Haswell-MB" (dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Hyper-threading, Turbo Boost, AES-NI, Intel TSX-NI, Smart Cache.*
- Core i5-4300M and higher also support *Intel VT-d, Intel vPro, Intel TXT*
- Transistors: 1.3 billion
- Die size: 181 mm^2

"Haswell-ULT" (SiP, dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Hyper-threading, Turbo Boost, AES-NI, Smart Cache.*
- All models i5-4250U and up support *Intel VT-d*
- Core i5-43x0U also supports Intel vPro, Intel TXT
- Transistors: 1.3 billion
- Die size: 181 mm^2

"Haswell-ULX" (SiP, dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Hyper-threading, Turbo Boost, AES-NI, Smart Cache.*
- Core i5-4300Y and higher also support *Intel VT-d, Intel TSX-NI, Intel vPro, Intel TXT*
- Transistors: 1.3 billion
- Die size: 181 mm^2

"Haswell-H" (dual-core, 22 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost (except i5-4402EC and i5-4410E), AES-NI, Intel TSX-NI, Smart Cache.*
- Transistors: 1.3 billion
- Die size: 181 mm^2
- Embedded models support Intel vPro, ECC memory

9.2.5 Broadwell microarchitecture (5th generation)

"Broadwell-U" (dual-core, 14 nm)

- All models support: *MMX, SSE, SSE2, SSE3, SSSE3, SSE4.1, SSE4.2, AVX, AVX2, FMA3, Enhanced Intel SpeedStep Technology (EIST), Intel 64, XD bit (an NX bit implementation), Intel VT-x, Intel VT-d, Hyper-threading, Turbo Boost, AES-NI, Smart Cache*, and configurable TDP (cTDP) down
- Core i5-5300U and higher also support Intel vPro, Intel TXT, and Intel TSX-NI
- Transistors: 1.3-1.9 billion [12]
- Die size: 82-133 mm^2 [12]

9.3 See also

- Nehalem (microarchitecture)
- Westmere (microarchitecture)
- Sandy Bridge
- Ivy Bridge (microarchitecture)
- Haswell (microarchitecture)
- List of Intel Core i3 microprocessors
- List of Intel Core i7 microprocessors

9.4 Notes

[1] Turbo describes the available frequency bins (+133 MHz for processors based on Nehalem microarchitecture) of Intel Turbo Boost Technology that are available for 4, 3, 2, 1 active cores respectively (depending on the number of CPU cores, included in the processor).

[2] Turbo describes the available frequency bins (+100 MHz for processors based on Sandy Bridge, Ivy Bridge and Haswell microarchitectures) of Intel Turbo Boost Technology that are available for 4, 3, 2, 1 active cores respectively (depending on the number of CPU cores, included in the processor).

9.5 References

[1] Core i5-655K, Core i5-661 does not support Intel TXT and Intel VT-d

[2] Core i5-2300, Core i5-2310, Core i5-2320, Core i5-2380P, Core i5-2405S, Core i5-2450P, Core i5-2500F and Core i5-2550K do not support Intel TXT and Intel VT-d

[3] Fully unlocked versus "limited" unlocked core

[4] Counting Transistors: Why 1.16B and 995M Are Both Correct, by Anand Lal Shimpi on 14 September 2011, www.anandtech.com

[5] http://www.anandtech.com/show/5876/the-rest-of-the-ivy-bridge-die-sizes

[6] http://vr-zone.com/articles/intel-s-broken-ivy-bridge-sku-s-last-to-arrive/15449.html

[7] Specifications of Haswell Refresh CPUs

[8] Some details of Haswell Refresh CPUs

[9] http://ark.intel.com/Compare.aspx?ids=43544,43560

[10] http://ark.intel.com/ProductCollection.aspx?familyId=43483

[11] "Intel® Core™ i5-2410M Processor". Intel. Retrieved 2012-01-01.

[12] Cyril Kowaliski (2015-01-05). "Intel's Broadwell-U arrives aboard 15W, 28W mobile processors". techreport.com. Retrieved 2015-01-12.

9.6 External links

- Intel Core i5 desktop processor product order code table
- Intel Core i5 mobile processor product order code table
- Search MDDS Database
- Intel ARK Database

Chapter 10

Pentium Dual-Core

This article is about Pentium Dual-Core branded processors from 2006 to 2009. For current and other Pentium processors, see Pentium.
Not to be confused with Pentium D.

The **Pentium Dual-Core** brand was used for mainstream x86-architecture microprocessors from Intel from 2006 to 2009 when it was renamed to Pentium. The processors are based on either the 32-bit *Yonah* or (with quite different microarchitectures) 64-bit *Merom-2M*, *Allendale*, and *Wolfdale-3M* core, targeted at mobile or desktop computers.

In terms of features, price and performance at a given clock frequency, Pentium Dual-Core processors were positioned above Celeron but below Core and Core 2 microprocessors in Intel's product range. The Pentium Dual-Core was also a very popular choice for overclocking, as it can deliver high performance (when overclocked) at a low price.

Pentium Dual-Core E2160 Overclock

10.1 Processor cores

In 2006, Intel announced a plan[1] to return the Pentium trademark from retirement to the market, as a moniker of low-cost Core microarchitecture processors based on the single-core Conroe-L but with 1 MiB of cache. The identification numbers for those planned Pentiums were similar to the numbers of the latter Pentium Dual-Core microprocessors, but with the first digit “1”, instead of “2”, suggesting their single-core functionality. A single-core Conroe-L with 1 MiB cache was deemed as not strong enough to distinguish the planned Pentiums from the Celerons, so it was replaced by dual-core CPUs, adding “Dual-Core” to the line's name. Throughout 2009, Intel changed the name back from Pentium Dual-Core to Pentium in its publications. Some processors were sold under both names, but the newer E5400 through E6800 desktop and SU4100/T4x00 mobile processors were not officially part of the Pentium Dual-Core line.

10.1.1 Yonah

Main article: Yonah (microprocessor)

The first processors using the brand appeared in notebook computers in early 2007. Those processors, named Pentium T2060, T2080, and T2130,[2] had the 32-bit Pentium M-derived Yonah core, and closely resembled the Core Duo T2050 processor with the exception of having 1 MB of L2 cache instead of 2 MB. All three of them had a 533 MHz FSB connecting the CPU with the memory. Intel developed the Pentium Dual-Core at the request of laptop manufacturers.[3]

10.1.2 Allendale

Main article: Conroe (microprocessor) § Allendale

Intel Pentium E2180 @ 2.00GHz closeup

Subsequently, on June 3, 2007, Intel released the desktop Pentium Dual-Core branded processors[4] known as the Pentium E2140 and E2160.[5] An E2180 model was released later in September 2007. These processors support the Intel 64 extensions, being based on the newer, 64-bit Allendale core with Core microarchitecture. These closely resembled the Core 2 Duo E4300 processor with the exception of having 1 MB of L2 cache instead of 2 MB.[2] Both of them had an 800 MHz FSB. They targeted the budget market above the Intel Celeron (Conroe-L single-core series) processors featuring only 512 KB of L2 cache. Such a step marked a change in the Pentium brand, relegating it to the budget segment rather than its former position as the mainstream/premium brand.[6] These CPUs are highly overclockable.[7]

10.1.3 Merom-2M

Main article: Merom (microprocessor) § Merom-2M

The mobile version of the Allendale processor, the Merom-2M, was also introduced in 2007, featuring 1MB of L2 cache but only 533 MT/s FSB with the T23xx processors. The bus clock was subsequently raised to 667 MT/s with the T3xxx Pentium processors that are made from the same dies.

10.1.4 Wolfdale-3M

Main article: Wolfdale (microprocessor) § Wolfdale-3M

The 45 nm E5200 model was released by Intel on August 31, 2008, with a larger 2MB L2 cache over the 65 nm E21xx series and the 2.5 GHz clock speed. The E5200 model is also a highly overclockable processor, with some enthusiasts reaching over 6 GHz[8] clock speed using liquid nitrogen cooling. Intel released the E6500K model using this core. The model features an unlocked multiplier, but is currently only sold in China.

10.1.5 Penryn-3M

Main article: Penryn (microprocessor) § Penryn-3M

The Penryn core is the successor to the Merom core and Intel's 45 nm version of their mobile series of Pentium Dual-Core microprocessors. The FSB is increased from 667 MHz to 800 MHz and the voltage is lowered. Intel released the first Penryn based Pentium Dual-Core, the T4200, in December 2008. Later, mobile Pentium T4000, SU2000 and SU4000 processors based on Penryn were marketed as Pentium.

10.2 Rebranding

See also: Pentium

The *Pentium Dual-Core* brand has been discontinued in early 2010 and replaced by the *Pentium* name. The Desktop E6000 series and the OEM-only mobile Pentium SU2000 and all later models were always called Pentium, but the Desktop Pentium Dual-Core E2000 and E5000 series processors had to be rebranded.

10.3 Comparison to the Pentium D

Although using the *Pentium* name, the desktop Pentium Dual-Core is based on the Core microarchitecture, which can clearly be seen when comparing the specification to the Pentium D, which is based on the NetBurst microarchitecture first introduced in the Pentium 4. Below the 2 or 4 MiB of shared-L2-cache-enabled Core 2 Duo, the desktop Pentium Dual-Core has 1 or 2 MiB of shared L2 Cache. In contrast, the Pentium D processors have either 2 or 4 MiB of non-shared L2 cache. Additionally, the fastest-clocked Pentium D has a factory boundary of 3.73 GHz, while the fastest-clocked desktop Pentium Dual-Core reaches 3.2 GHz. A major difference among these processors is that the desktop Pentium Dual Core processors have a TDP of only 65 W while the Pentium D ranges between 95 to 130 W. Despite the reduced clock speed, and lower amounts of cache, Pentium dual-core outperfomed Pentium D by a fairly large margin.

10.4 See also

- Pentium
- List of Intel Pentium Dual-Core microprocessors

- List of Intel Pentium microprocessors

10.5 References

[1] DailyTech – Intel “Conroe-L” Details Unveiled

[2] “Intel Pentium E2140 & E2160 review”. TechSpot. Retrieved 2007-06-23.

[3] The multicore era is upon us

[4] “Pentium E/Celeron 400 to be releasing on June 3”. HKEPC Hardware. Retrieved 2007-03-16.

[5] Shilov, Anton. “Intel Readies Pentium E2000-Series Processors”. X-bit labs. Retrieved 2007-03-04.

[6] Intel Processor Numbers chart for Intel Pentium Dual-Core processors

[7] Schmid, Patrick (12 September 2007). "$89 Pentium Dual Core that Runs at 3.2 GHz”. Tom’s Hardware. Retrieved 2009-09-21.

[8] http://ripping.org/database.php?cpuid=858

10.6 External links

Chapter 11

Xeon

Not to be confused with the ATI Xenos, or Xenon.

The **Xeon** /ˈziːɒn/ is a brand of x86 microprocessors designed and manufactured by Intel Corporation, targeted at the non-consumer workstation, server, and embedded system markets. Primary advantages of the Xeon CPUs, when compared to the majority of Intel's desktop-grade consumer CPUs, are their multi-socket capabilities, higher core counts, and support for ECC memory.

11.1 Overview

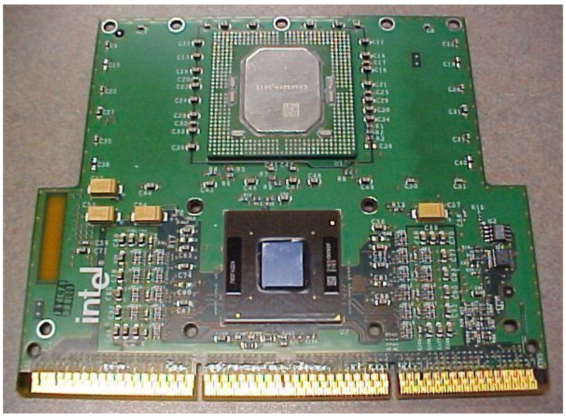
The *Xeon* brand has been maintained over several generations of x86 and x86-64 processors. Older models added the *Xeon* moniker to the end of the name of their corresponding desktop processor, but more recent models used the name *Xeon* on its own. The *Xeon* CPUs generally have more cache than their desktop counterparts in addition to multiprocessing capabilities.

11.2 P6-based Xeon

11.2.1 Pentium II Xeon

List: List of Intel Xeon microprocessors#"Drake” (250 nm)

The first Xeon-branded processor was the Pentium II Xeon (code-named "**Drake**"). It was released in 1998, replacing the Pentium Pro in Intel's server lineup. The Pentium II Xeon was a "*Deschutes*" Pentium II (and shared the same product code: 80523) with a full-speed 512 kB, 1 MB, or 2 MB L2 cache. The L2 cache was implemented with custom 512 kB SRAMs developed by Intel. The number of SRAMs depended on the amount of cache. A 512 kB configuration required one SRAM, a 1 MB configuration: two SRAMs, and a 2 MB configuration: four SRAMs on both sides of the PCB. Each SRAM was a 12.90 mm by 17.23 mm (222.21 mm^2) die fabricated in a 0.35 µm four-layer metal CMOS process and packaged in a cavity-down wire-bonded land grid array (LGA).[1] The additional cache required a larger module and thus the Pentium II Xeon used a larger slot, Slot 2. It was supported by the 440GX dual-processor workstation chipset and the 450NX quad- or octo-processor chipset.



A 450 MHz Pentium II Xeon with a 512 kB L2 cache. The cartridge cover has been removed.

11.2.2 Pentium III Xeon

List: List of Intel Xeon microprocessors#"Tanner” (250 nm)

In 1999, the Pentium II Xeon was replaced by the Pentium III Xeon. Reflecting the incremental changes from the Pentium II "*Deschutes*" core to the Pentium III "*Katmai*" core, the first Pentium III Xeon, named "**Tanner**", was just like its predecessor except for the addition of Streaming SIMD Extensions (SSE) and a few cache controller improvements. The product codes for **Tanner** mirrored that of *Katmai*; 80525.

List: List of Intel Xeon microprocessors#"Cascades” (180 nm)

The second version, named "**Cascades**", was based on the Pentium III "*Coppermine*" core. The "**Cascades**" Xeon used a 133 MHz bus and relatively small 256 kB on-die L2 cache resulting in almost the same capabilities as the Slot 1 *Coppermine* processors, which were capable of dual-processor operation but not quad-processor operation.

To improve this situation, Intel released another version, officially also named "**Cascades**", but often referred to as "**Cascades 2 MB**". That came in two variants: with 1 MB or 2 MB of L2 cache. Its bus speed was fixed at 100 MHz, though in practice the cache was able to offset this. The product code for **Cascades** mirrored that of *Coppermine*; 80526.

11.3 Netburst-based Xeon

11.3.1 Xeon (DP) & Xeon MP (32-bit)

Foster

List: List of Intel Xeon microprocessors#"Foster" (180 nm)
List: List of Intel Xeon microprocessors#"Foster MP" (180 nm)

In mid-2001, the Xeon brand was introduced ("Pentium" was dropped from the name). The initial variant that used the new NetBurst microarchitecture, "**Foster**", was slightly different from the desktop Pentium 4 ("*Willamette*"). It was a decent chip for workstations, but for server applications it was almost always outperformed by the older Cascades cores with a 2 MB L2 cache and AMD's Athlon MP. Combined with the need to use expensive Rambus Dynamic RAM, the Foster's sales were somewhat unimpressive.

At most two Foster processors could be accommodated in a symmetric multiprocessing (SMP) system built with a mainstream chipset, so a second version (**Foster MP**) was introduced with a 1 MB L3 cache and the Jackson Hyper-Threading capacity. This improved performance slightly, but not enough to lift it out of third place. It was also priced much higher than the dual-processor (DP) versions. The *Foster* shared the 80528 product code with Willamette.

Prestonia

List: List of Intel Xeon microprocessors#"Prestonia" (130 nm)

In 2002 Intel released a 130 nm version of Xeon branded CPU, codenamed "**Prestonia**". It supported Intel's new Hyper-Threading technology and had a 512 kB L2 cache. This was based on the "*Northwood*" Pentium 4 core. A new server chipset, E7500 (which allowed the use of dual-channel DDR SDRAM), was released to support this processor in servers, and soon the bus speed was boosted to 533 MT/s (accompanied by new chipsets: the E7501 for servers and the E7505 for workstations). The *Prestonia* performed much better than its predecessor and noticeably better than Athlon MP. The support of new features in the E75xx series also gave it a key advantage over the Pentium III Xeon and Athlon MP branded CPUs (both stuck with rather old chipsets), and it quickly became the top-selling server/workstation processor.

11.3.2 "Gallatin"

List: List of Intel Xeon microprocessors#"Gallatin" (130 nm)
List: List of Intel Xeon microprocessors#"Gallatin" MP (130 nm)

Subsequent to the *Prestonia* was the "**Gallatin**", which had an L3 cache of 1 MB or 2 MB. Its Xeon MP version also performed much better than the *Foster MP*, and was popular in servers. Later experience with the 130 nm process allowed Intel to create the Xeon MP branded *Gallatin* with 4 MB cache. The Xeon branded *Prestonia* and *Gallatin* were designated 80532, like Northwood.

11.3.3 Xeon (DP) & Xeon MP (64-bit)

Nocona and Irwindale

Main article: Pentium 4 § Prescott
List: List of Intel Xeon microprocessors#"Nocona" (90 nm)
List: List of Intel Xeon microprocessors#"Irwindale" (90 nm)

Due to a lack of success with Intel's Itanium and Itanium 2 processors, AMD was able to introduce x86-64, a 64-bit extension to the x86 architecture. Intel followed suit by including Intel 64 (formerly EM64T; it is almost identical to AMD64) in the 90 nm version of the Pentium 4 ("*Prescott*"), and a Xeon version codenamed "**Nocona**" with 1 MB L2 cache was released in 2004. Released with it were the E7525 (workstation), E7520 and E7320 (both server) chipsets, which added support for PCI Express, DDR-II and Serial ATA. The Xeon was noticeably slower than AMD's Opteron, although it could be faster in situations where Hyper-Threading came into play.

A slightly updated core called "**Irwindale**" was released in early 2005, with 2 MB L2 cache and the ability to have its clock speed reduced during low processor demand. Although it was a bit more competitive than the *Nocona* had been, independent tests showed that AMD's Opteron still outperformed *Irwindale*. Both of these Prescott-derived Xeons have the product code 80546.

Cranford and Potomac

Main article: Pentium 4 § Prescott
List: List of Intel Xeon microprocessors#"Cranford" (90 nm)
List: List of Intel Xeon microprocessors#"Potomac" (90 nm)

64-bit Xeon MPs were introduced in April 2005. The cheaper "**Cranford**" was an MP version of *Nocona*, while the more expensive "**Potomac**" was a *Cranford* with 8 MB of L3 cache. Like Nocona and Irwindale, they also have product code 80546.

11.3.4 Dual-Core Xeon

"Paxville DP"

List: List of Intel Xeon microprocessors#"Paxville DP" (90 nm)

The first dual-core CPU branded Xeon, codenamed **Paxville DP**, product code 80551, was released by Intel on 10 October 2005. Paxville DP had NetBurst microarchitecture, and was a dual-core equivalent of the single-core Irwindale (related to the Pentium D branded "*Smithfield"*") with 4 MB of L2 Cache (2 MB per core). The only Paxville DP model released ran at 2.8 GHz, featured an 800 MT/s front side bus, and was produced using a 90 nm process. <sup class="noprint Inline-Template "noprint Inline-Template"" style="white-space:nowrap;">[*check quotation syntax*]

7000-series "Paxville MP"

List: List of Intel Xeon microprocessors#"Paxville MP" (90 nm)

An MP-capable version of Paxville DP, codenamed **Paxville MP**, product code 80560, was released on 1 November 2005. There are two versions: one with 2 MB of L2 Cache (1 MB per core), and one with 4 MB of L2 (2 MB per core). Paxville MP, called the dual-core Xeon 7000-series, was produced using a 90 nm process. Paxville MP clock ranges between 2.67 GHz and 3.0 GHz (model numbers 7020–7041), with some models having a 667 MT/s FSB, and others having an 800 MT/s FSB.

7100-series "Tulsa"

List: List of Intel Xeon microprocessors#"Tulsa" (65 nm)

Released on 29 August 2006,[2] the 7100 series, codenamed **Tulsa** (product code 80550), is an improved version of Paxville MP, built on a 65 nm process, with 2 MB of L2 cache (1 MB per core) and up to 16 MB of L3 cache. It uses Socket 604 . Tulsa was released in two lines: the N-line uses a 667 MT/s FSB, and the M-line uses an 800 MT/s FSB. The N-line ranges from 2.5 GHz to 3.5 GHz (model numbers 7110N-7150N), and the M-line ranges from 2.6 GHz to 3.4 GHz (model numbers 7110M-7140M). L3 cache ranges from 4 MB to 16 MB across the models.[3]

5000-series "Dempsey"

List: List of Intel Xeon microprocessors#"Dempsey" (65 nm)

On 23 May 2006, Intel released the dual-core CPU (Xeon branded 5000 series) codenamed **Dempsey** (product code 80555). Released as the Dual-Core Xeon 5000-series, Dempsey is a NetBurst microarchitecture processor produced using a 65 nm process, and is virtually identical to Intel's "Presler" Pentium Extreme Edition, except for the addition of SMP support, which lets Dempsey operate in dual-processor systems. Dempsey ranges between 2.50 GHz and 3.73 GHz (model numbers 5020–5080). Some models have a 667 MT/s FSB, and others have a 1066 MT/s FSB. Dempsey has 4 MB of L2 Cache (2 MB per core). A Medium Voltage model, at 3.2 GHz and 1066 MT/s FSB (model number 5063), has also been released. Dempsey also introduces a new interface for Xeon processors: LGA 771, also known as **Socket J**. Dempsey was the first Xeon core in a long time to be somewhat competitive with its Opteron-based counterparts, although it could not claim a decisive lead in any performance metric – that would have to wait for its successor, the Woodcrest.

11.4 Pentium M (Yonah) based Xeon

11.4.1 LV (ULV), "Sossaman"

List: List of Intel Xeon microprocessors#"Sossaman" (65 nm)

On 14 March 2006, Intel released a dual-core processor codenamed **Sossaman** and branded as *Xeon* LV (low-voltage). Subsequently an ULV (ultra-low-voltage) version was released. The *Sossaman* was a low-/ultra-low-power and double-processor capable CPU (like AMD Quad FX), based on the "*Yonah*" processor, for ultra-dense non-consumer environment (i.e. targeted at the blade-server and embedded markets), and was rated at a thermal design power (TDP) of 31 W (LV: 1.66 GHz, 2 GHz and 2.16 GHz) and 15 W (ULV: 1.66 GHz).[4] As such, it supported most of the same features as earlier

Xeons: Virtualization Technology, 667 MT/s front side bus, and dual-core processing, but did not support 64-bit operations, so it could not run 64-bit server software, such as Microsoft Exchange Server 2007, and therefore was limited to 16 GB of memory. A planned successor, codenamed "*Merom MP*" was to be a drop-in upgrade to enable *Sossaman*-based servers to upgrade to 64-bit capability. However, this was abandoned in favour of low-voltage versions of the *Woodcrest LV* processor leaving the *Sossaman* at a dead-end with no upgrade path.

11.5 Core-based Xeon

11.5.1 Dual-Core

3000-series "Conroe"

Main article: Conroe (microprocessor)
List: List of Intel Xeon microprocessors#"Conroe" (65 nm)

The 3000 series, codenamed **Conroe** (product code 80557) dual-core Xeon (branded) CPU,[5] released at the end of September 2006, was the first Xeon for single-CPU operation. The same processor is branded as Core 2 Duo or as Pentium Dual-Core and Celeron, with varying features disabled. They use LGA 775 (Socket T), operate on a 1066 MHz front-side bus, support Enhanced Intel Speedstep Technology and Intel Virtualization Technology but do not support Hyper-Threading. Conroe Processors with a number ending in "5" have a 1333 MT/s FSB.[6]

- Models marked with a star are not present in Intel's database[7]

3100-series "Wolfdale"

Main article: Wolfdale (microprocessor)
List: List of Intel Xeon microprocessors#"Wolfdale" (45 nm)
List: List of Intel Xeon microprocessors#"Wolfdale-CL" (45 nm)

The 3100 series, codenamed **Wolfdale** (product code 80570) dual-core Xeon (branded) CPU, was just a rebranded version of the Intel's mainstream Core 2 Duo E7000/E8000 and Pentium Dual-Core E5000 processors, featuring the same 45 nm process and 6 MB of L2 cache. Unlike most Xeon processors, they only support single-CPU operation. They use LGA 775 (Socket T), operate on a 1333 MHz front-side bus, support Enhanced Intel Speedstep Technology and Intel Virtualization Technology but do not support Hyper-Threading.

5100-series "Woodcrest"

List: List of Intel Xeon microprocessors#"Woodcrest" (65 nm)

On 26 June 2006, Intel released the dual-core CPU (Xeon branded 5100 series) codenamed **Woodcrest** (product code 80556); it was the first Intel Core microarchitecture processor to be launched on the market. It is a server and workstation version of the Intel Core 2 processor. Intel claims that it provides an 80% boost in performance, while reducing power consumption by 20% relative to the Pentium D.

Most models have a 1333 MT/s FSB, except for the 5110 and 5120, which have a 1066 MT/s FSB. The fastest processor (5160) operates at 3.0 GHz. All Woodcrests use LGA 771 and all except two models have a TDP of 65 W. The 5160 has a TDP of 80 W and the 5148LV (2.33 GHz) has a TDP of 40 W. The previous generation Xeons had a TDP of 130 W. All models support Intel 64 (Intel's x86-64 implementation), the XD bit, and Virtualization Technology, with the "Demand Based Switching" power management option only on Dual-Core Xeon 5140 or above. Woodcrest has 4 MB of shared L2 Cache.

5200-series "Wolfdale-DP"

List: List of Intel Xeon microprocessors#"Wolfdale-DP" (45 nm)

On 11 November 2007, Intel released the dual-core CPU (Xeon branded 5200 series) codenamed **Wolfdale-DP** (product code 80573).[8] It is built on a 45 nm process like the desktop Core 2 Duo and Xeon-SP Wolfdale, featuring Intel 64 (Intel's x86-64 implementation), the XD bit, and Virtualization Technology. It is unclear whether the "Demand Based Switching" power management is available on the L5238.[9] Wolfdale has 6 MB of shared L2 Cache.

7200-series "Tigerton"

Main section: #Tigerton
List: List of Intel Xeon microprocessors#"Tigerton-DC" (65 nm)

The 7200 series, codenamed **Tigerton** (product code 80564) is an MP-capable processor, similar to the 7300 series, but, in contrast, only one core is active on each silicon chip, and the other one is turned off (blocked), resulting as a dual-core capable processor.

11.5.2 Quad-Core and Multi-Core Xeon

3200-series "Kentsfield"

Main article: Kentsfield (microprocessor)
List: List of Intel Xeon microprocessors#"Kentsfield" (65 nm)

Intel released relabeled versions of its quad-core (2×2) Core 2 Quad processor as the Xeon 3200-series (product code 80562) on 7 January 2007.[10] The 2 × 2 "quad-core" (dual-die dual-core[11]) comprised two separate dual-core die next to each other in one CPU package. The models are the X3210, X3220 and X3230, running at 2.13 GHz, 2.4 GHz and 2.66 GHz, respectively.[12] Like the 3000-series, these models only support single-CPU operation and operate on a 1066 MHz front-side bus. It is targeted at the "blade" market. The X3220 is also branded and sold as Core2 Quad Q6600, the X3230 as Q6700.

3300-series "Yorkfield"

Main article: Yorkfield (microprocessor)
List: List of Intel Xeon microprocessors#"Yorkfield" (45 nm)
List: List of Intel Xeon microprocessors#"Yorkfield-CL" (45 nm)

Intel released relabeled versions of its quad-core Core 2 Quad Yorkfield Q9400 and Q9x50 processors as the Xeon 3300-series (product code 80569). This processor comprises two separate dual-core dies next to each other in one CPU package and manufactured in a 45 nm process. The models are the X3320, X3350, X3360, X3370 and X3380, running at 2.50 GHz, 2.66 GHz, 2.83 GHz, 3.0 GHz, and 3.16 GHz, respectively. The L2 cache is a unified 6 MB per die (except for the X3320 with a smaller 3 MB L2 cache per die), and a front-side bus of 1333 MHz. All models feature Intel 64 (Intel's x86-64 implementation), the XD bit, and Virtualization Technology, as well as "Demand Based Switching".

The Yorkfield-CL (product code 80584) variant of these processors are X3323, X3353 and X3363. They have a reduced TDP of 80W and are made for single-CPU LGA 771 systems instead of LGA 775, which is used in all other Yorkfield processors. In all other respects, they are identical to their Yorkfield counterparts.

5300-series "Clovertown"

List: List of Intel Xeon microprocessors#"Clovertown" (65 nm)

A quad-core (2×2) successor of the Woodcrest for DP segment, consisting of two dual-core Woodcrest chips in one package similarly to the dual-core Pentium D branded CPUs (two single-core chips) or the quad-core Kentsfield. All Clovertowns use the LGA 771 package. The Clovertown has been usually implemented with two Woodcrest dies on a multi-chip module, with 8 MB of L2 cache (4 MB per die). Like Woodcrest, lower models use a 1066 MT/s FSB, and higher models use a 1333 MT/s FSB. Intel released **Clovertown**, product code 80563, on 14 November 2006[13] with models E5310, E5320, E5335, E5345, and X5355, ranging from 1.6 GHz to 2.66 GHz. All models support: MMX, SSE, SSE2, SSE3, SSSE3, Intel 64, XD bit (an NX bit implementation), Intel VT. The E and X designations are borrowed from Intel's Core 2 model numbering scheme; an ending of −0 implies a 1066 MT/s FSB, and an ending of −5 implies a 1333 MT/s FSB.[12] All models have a TDP of 80 W with the exception of the X5355, which has a TDP of 120 W. A low-voltage version of Clovertown with a TDP of 50 W has a model numbers L5310, L5320 and L5335 (1.6 GHz, 1.86 GHz and 2.0 GHz respectively). The 3.0 GHz X5365 arrived in July 2007, and became available in the Apple Mac Pro on 4 April 2007.[14] The X5365 performs up to around 38 GFLOPS in the LINPACK benchmark.

5400-series "Harpertown"

List: List of Intel Xeon microprocessors#"Harpertown" (45 nm)

On 11 November 2007 Intel presented Yorkfield-based Xeons – called Harpertown (product code 80574) – to the public. This family consists of dual die quad-core CPUs manufactured on a 45 nm process and featuring 1066 MHz, 1333 MHz, 1600 MHz front-side buses, with TDP rated from 40 W to 150 W depending on the model. These processors fit in the LGA 771 package. All models feature Intel 64 (Intel's x86-64 implementation), the XD bit, and Virtualization Technology. All except the E5405 and L5408 also feature Demand Based Switching.[15][16] The supplementary character in front of the model-number represents the thermal rating: an L depicts a TDP of 40 W or 50 W, an E depicts 80 W whereas an X is 120 W TDP or above. The speed of 3.00 GHz comes as four models, two models with 80 W TDP two other models with 120 W TDP with 1333 MHz or 1600 MHz front-side bus respectively. The fastest Harpertown is the X5492 whose TDP of 150 W is higher than those of the Prescott-based Xeon DP but having twice as many cores. (The X5482 is also sold under the name "Core 2 Extreme QX9775" for use in the Intel SkullTrail system.)

Intel 1600 MHz front-side bus Xeon processors will drop into the Intel 5400 (Seaburg) chipset whereas several mainboards featuring the Intel 5000/5200-chipset are enabled to run the processors with a 1333 MHz front-side bus speed. Seaburg features support for dual PCIe 2.0 x16 slots and up to 128 GB of memory.[17][18]

7300-series "Tigerton"

List: List of Intel Xeon microprocessors#"Tigerton" (65 nm)

The 7300 series, codenamed **Tigerton** (product code 80565) is a four-socket (packaged in Socket 604) and more capable quad-core processor, consisting of two dual core Core2 architecture silicon chips on a single ceramic module, similar to Intel's Xeon 5300 series Clovertown processor modules.[19]

The 7300 series uses Intel's Caneland (Clarksboro) platform.

Intel claims the 7300 series Xeons offer more than twice the performance per watt as Intel's previous generation 7100 series. The 7300 series' Caneland chipset provides a point to point interface allowing the full front side bus bandwidth per processor.

The 7xxx series is aimed at the large server market, supporting configurations of up to 32 CPUs per host.

7400-series "Dunnington"

List: List of Intel Xeon microprocessors#"Dunnington" (45 nm)

Dunnington[20] – the last CPU of the Penryn generation and Intel's first multi-core (above two) die – features a single-die six- (or *hexa-*) core design with three unified 3 MB L2 caches (resembling three merged 45 nm dual-core Wolfdale dies), and 96 kB L1 cache (Data) and 16 MB of L3 cache. It features 1066 MHz FSB, fits into the Tigerton's mPGA604 socket, and is compatible with the both the Intel Caneland, and IBM X4 chipsets. These processors support DDR2-1066 (533 MHz), and have a maximum TDP below 130 W. They are intended for blades and other stacked computer systems. Availability was scheduled for the second half of 2008. It was followed shortly by the Nehalem microarchitecture. Total transistor count is 1.9 billion.[21]

Announced on 15 September 2008.[22]

11.6 Nehalem-based Xeon

11.6.1 3400-series "Lynnfield"

Main article: Lynnfield (microprocessor)
List: List of Intel Xeon microprocessors#"Lynnfield" (45 nm)

Xeon 3400-series processors based on **Lynnfield** fill the gap between the previous 3300-series "Yorkfield" processors and the newer 3500-series "Bloomfield". Like Bloomfield, they are quad-core single-package processors based on the Nehalem microarchitecture, but were introduced almost a year later, in September 2009. The same processors are marketed for mid-range to high-end desktops systems as Core i5 and Core i7. They have two integrated memory channels as well as PCI Express and Direct Media Interface (DMI) links, but no QuickPath Interconnect (QPI) interface.

11.6.2 3400-series "Clarkdale"

Main article: Clarkdale (microprocessor)
List: List of Intel Xeon microprocessors#"Clarkdale" (MCP, 32 nm)

At low end of the 3400-series is not a Lynnfield but a **Clarkdale** processor, which is also used in the Core i3-500 and Core i5-600 processors as well as the Celeron G1000 and G6000 Pentium series. A single model was released in March 2010, the Xeon L3406. Compared to all other Clarkdale-based products, this one does not support integrated graphics, but has a much lower thermal design power of just 30 W. Compared to the Lynnfield-based Xeon 3400 models, it only offers two cores.

11.6.3 3500-series "Bloomfield"

Main article: Bloomfield (microprocessor)
List: List of Intel Xeon microprocessors#"Bloomfield" (45 nm)

Bloomfield is the codename for the successor to the Xeon Core microarchitecture, is based on the Nehalem microarchitecture and uses the same 45 nm manufacturing methods as Intel's Penryn. The first processor released with the Nehalem architecture is the desktop Intel Core i7, which was released in November 2008. This is the server version for single CPU systems. This is a **single-socket** Intel Xeon processor. The performance improvements over previous Xeon processors are based mainly on:

- Integrated memory controller supporting three memory channels of DDR3 UDIMM (Unbuffered) or RDIMM (Registered)

- A new point-to-point processor interconnect *QuickPath*, replacing the legacy front side bus

- Simultaneous multithreading by multiple cores and hyper-threading (2× per core).

11.6.4 <span id="5500-series “Gainestown”">5500-series “Gainestown”

List: List of Intel Xeon microprocessors#"Gainestown” (45 nm)

Gainestown or **Nehalem-EP**, the successor to the Xeon Core microarchitecture, is based on the Nehalem microarchitecture and uses the same 45 nm manufacturing methods as Intel’s Penryn. The first processor released with the Nehalem microarchitecture is the desktop Intel Core i7, which was released in November 2008. Server processors of the Xeon 55xx range were first supplied to testers in December 2008.[23]

The performance improvements over previous Xeon processors are based mainly on:

- Integrated memory controller supporting three memory channels of DDR3 SDRAM.
- A new point-to-point processor interconnect *QuickPath*, replacing the legacy front side bus. Gainestown has two QuickPath interfaces.
- Hyper-threading (2× per core, starting from 5518), that was already present in pre-Core Duo processors.

11.6.5 <span id="C3500/C5500-Series “Jasper Forest”">C3500/C5500-series “Jasper Forest”

List: List of Intel Xeon microprocessors#"Jasper Forest” (45 nm)

Jasper Forest is a Nehalem-based embedded processor with PCI Express connections on-die, core counts from 1 to 4 cores and power envelopes from 23 to 85 watts.[24]

The uni-processor version without QPI comes as LC35xx and EC35xx, while the dual-processor version is sold as LC55xx and EC55xx and uses QPI for communication between the processors. Both versions use a DMI link to communicate with the 3420 that is also used in the 3400-series Lynfield Xeon processors, but use an LGA 1366 package that is otherwise used for processors with QPI but no DMI or PCI Express links. The CPUID code of both Lynnfield and Jasper forest is 106Ex, i.e. family 6, model 30.

The Celeron P1053 belongs into the same family as the LC35xx series, but lacks some RAS features that are present in the Xeon version.

11.6.6 3600/5600-series “Gulftown” & “Westmere-EP”

Main article: Gulftown (microprocessor)
List: List of Intel Xeon microprocessors#"Gulftown” (32 nm), List of Intel Xeon microprocessors#"Westmere-EP” (32 nm)

Gulftown or **Westmere-EP**, a six-core 32 nm Westmere-based processor, is the basis for the Xeon 36xx and 56xx series and the Core i7−980X. It launched in the first quarter of 2010. The 36xx-series follows the 35xx-series Bloomfield uni-processor model while the 56xx-series follows the 55xx-series Gainestown dual-processor model and both are socket compatible to their predecessors.

11.6.7 6500/7500-series “Beckton”

List: List of Intel Xeon microprocessors#"Beckton” (45 nm)

Beckton or **Nehalem-EX** (EXpandable server market) is a Nehalem-based processor with up to eight cores and uses buffering inside the chipset to support up to 16 standard DDR3 DIMMS per CPU socket without requiring the use of FB-DIMMS.[25] Unlike all previous Xeon MP processors, Nehalem-EX uses the new LGA 1567 package, replacing the Socket 604 used in the previous models, up to Xeon 7400 “Dunnington”. The 75xx models have four QuickPath interfaces, so it can be used in up-to eight-socket configurations, while the 65xx models are only for up to two sockets. Designed by the Digital Enterprise Group (DEG) Santa Clara and Hudson Design Teams, Beckton is manufactured on the P1266 (45 nm) technology. Its launch in March 2010 coincided with that of its direct competitor, AMD’s Opteron 6xxx “Magny-Cours”.[26]

Most models limit the number of cores and QPI links as well as the L3 Cache size in order to get a broader range of products out of the single chip design.

11.6.8 E7-x8xx-series “Westmere-EX”

List: List of Intel Xeon microprocessors#"Westmere-EX” (32 nm) Expandable

Westmere-EX is the follow-on to Beckton/Nehalem-EX and the first Intel Chip to have ten CPU cores. The microarchitecture is the same as in the six-core Gulftown/Westmere-EP processor, but it uses the LGA 1567 package like Beckton to support up to eight sockets.

Starting with Westmere-EX, the naming scheme has changed once again, with “E7-xxxx” now signifying the high-end line of Xeon processors using a package that supports larger than two-CPU configurations, formerly the 7xxx series. Similarly, the 3xxx uniprocessor and 5xxx dual-processor series turned into E3-xxxx and E5-xxxx, respectively, for later processors.

11.7 Sandy Bridge– and Ivy Bridge–based Xeon

11.7.1 E3-12xx-series "Sandy Bridge"

Main article: Sandy Bridge
List: List of Intel Xeon microprocessors#"Sandy Bridge" (32 nm)

The **Xeon E3-12xx** line of processors, introduced in April 2011, uses the Sandy Bridge chips that are also the base for the Core i3/i5/i7-2xxx and Celeron/Pentium Gxxx products using the same LGA 1155 socket, but with a different set of features disabled. Notably, the Xeon variants include support for ECC memory, VT-d and trusted execution that are not present on the consumer models, while only some Xeon E3 enable the integrated GPU that is present on Sandy Bridge. Like its Xeon 3400-series predecessors, the Xeon E3 only supports operation with a single CPU socket and is targeted at entry-level workstations and servers. The CPUID of this processor is 0206A7h, the product code is 80623.

11.7.2 E3-12xx v2-series "Ivy Bridge"

Main article: Ivy Bridge (microarchitecture)

Xeon E3-12xx v2 is a minor update of the Sandy Bridge based E3-12xx, using the 22 nm shrink, and providing slightly better performance while remaining backwards compatible. They were released in May 2012 and mirror the desktop Core i3/i5/i7-3xxx parts.

11.7.3 E5-14xx/24xx series "Sandy Bridge-EN" and E5-16xx/26xx/46xx-series "Sandy Bridge-EP"

Main article: Sandy Bridge-E
List: List of Intel Xeon microprocessors#"Sandy Bridge-E" (32 nm)
List: List of Intel Xeon microprocessors#"Sandy Bridge-EN" (32 nm) Entry
List: List of Intel Xeon microprocessors#"Sandy Bridge-EP" (32 nm) Efficient Performance

The **Xeon E5-16xx** processors follow the previous Xeon 3500/3600-series products as the high-end single-socket platform, using the LGA 2011 package introduced with this processor. They share the Sandy Bridge-E platform with the single-socket Core i7-38xx and i7-39xx processors. The CPU chips have no integrated GPU but eight CPU cores, some of which are disabled in the entry-level products. The **Xeon E5-26xx** line has the same features but also enables multi-socket operation like the earlier Xeon 5000-series and Xeon 7000-series processors.

11.7.4 E5-14xx v2/24xx v2 series "Ivy Bridge-EN" and E5-16xx v2/26xx v2/46xx v2 series "Ivy Bridge-EP"

Main article: Ivy Bridge-EN/EP
List: List of Intel Xeon microprocessors#Xeon E5-1xxx v2 (uniprocessor)
List: List of Intel Xeon microprocessors#Xeon E5-2xxx v2 (dual-processor)
List: List of Intel Xeon microprocessors#Xeon E5-4xxx v2 (quad-processor)

The **Xeon E5 v2** line was an update, released in September 2013 to replace the original Xeon E5 processors with a variant based on the Ivy Bridge shrink. The maximum number of CPU cores was raised to 12 per processor module and the total L3 cache was upped to 30 MB.[27][28] The consumer version of the Xeon E5-16xx v2 processor is the Core i7-48xx and 49xx.

11.7.5 E7-28xx v2/48xx v2/88xx v2 series "Ivy Bridge-EX"

Main article: Ivy Bridge-EX
List: List of Intel Xeon microprocessors#Xeon E7-28xx v2 (dual-processor)
List: List of Intel Xeon microprocessors#Xeon E7-48xx v2 (quad-processor)
List: List of Intel Xeon microprocessors#Xeon E7-88xx v2 (octa-processor)

The **Xeon E7 v2** line was an update, released in February 2014 to replace the original Xeon E7 processors with a variant based on the Ivy Bridge shrink. There was no Sandy Bridge version of these processors.

11.8 Haswell-based Xeon

Main article: Haswell (microarchitecture)
List: Haswell (microarchitecture) § Server processors

11.8.1 E3-12xx v3-series "Haswell"

Intel Xeon E3-1241 v3 CPU, sitting atop the inside part of its retail box that contains an OEM fan-cooled heatsink

Introduced in May 2013, **Xeon E3-12xx v3** is the first Xeon series based on the Haswell microarchitecture. It uses the new LGA 1150 socket, which was introduced with the desktop Core i5/i7 Haswell processors, incompatible with the LGA 1155 that was used in Xeon E3 and E3 v2. As before, the main difference between the desktop and server versions is added support for ECC memory in the Xeon-branded parts. The main benefit of the new microarchitecture is better power efficiency.

11.8.2 E5-16xx/26xx v3-series "Haswell-EP"

Intel Xeon E5-1650 v3 CPU; its retail box contains no OEM heatsink

Introduced in September 2014, **Xeon E5-16xx v3** and **Xeon E5-26xx v3** series use the new LGA 2011-v3 socket, which is incompatible with the LGA 2011 socket used by earlier Xeon E5 and E5 v2 generations based on Sandy Bridge and Ivy Bridge microarchitectures. Some of the main benefits of this generation, when compared to the previous one, are improved power efficiency, higher core counts, and bigger last level caches (LLCs). Following the already used nomenclature, Xeon E5-26xx v3 series allows multi-socket operation.

One of the new features of this generation is that Xeon E5 v3 models with more than 10 cores support cluster on die (COD) operation mode, allowing CPU's multiple columns of cores and LLC slices to be logically divided into what is presented as two non-uniform memory access (NUMA) CPUs to the operating system. By keeping data and instructions local to the "partition" of CPU which is processing them, thus decreasing the LLC access latency, COD brings performance improvements to NUMA-aware operating systems and applications.[29]

11.9 Supercomputers

By 2013 Xeon processors were ubiquitous in supercomputers—more than 80% of the Top500 machines in 2013 used them. For the very fastest machines, much of the performance comes from compute accelerators; Intel's entry into that market was the Xeon Phi, the first machines using it appeared in the June 2012 list and by June 2013 it was used in the fastest computer in the world.

- The first Xeon-based machines in the top-10 appeared in November 2002, two clusters at Lawrence Livermore National Laboratory and at NOAA.
- The first Xeon-based machine to be in the first place of the Top500 was the Chinese Tianhe-IA in November 2010, which used a mixed Xeon-nVIDIA GPGPU configuration; it was overtaken by the Japanese K computer in 2012, but the Tianhe-2 system using 12-core Xeon E5-2692 processors and Xeon Phi cards occupied the first place in both Top500 lists of 2013.
- The SuperMUC system, using 8-core Xeon E5-2680 processors but no accelerator cards, managed fourth place in June 2012 and had dropped to tenth by November 2013
- Xeon processor-based systems are among the top 20 fastest systems by memory bandwidth as measured by the STREAM benchmark.[30]
- An Intel Xeon virtual SMP system leveraging ScaleMP's Versatile SMP (vSMP) architecture with 128 cores and 1TB RAM.[31] This system aggregates 16 Stoakley platform (Seaburg chipset) systems with total of 32 Harpertown processors.

11.10 See also

- AMD Opteron

- Intel Xeon Phi, brand name for family of products using the Intel MIC architecture
- List of Intel Xeon microprocessors
- List of Intel microprocessors

11.11 References

[1] Bateman, B. et al. (February 1998). "A 450MHz 512kB Second-Level Cache with a 3.6GB/s Data Bandwidth". *ISSCC Digest of Technical Papers*. doi:10.1109/ISSCC.1998.672528.

[2] New High-End Intel Server Processors Expand Performance Leadership, Intel News Release.

[3] Intel prices up Woodcrest, Tulsa server chips, The Inquirer.

[4] "Intel drops 32-bit dual-core LV processors". TG Daily. Retrieved 2007-07-31.

[5] Intel Adds Low End Xeons to Roadmap, DailyTech

[6] Intel Readies New Xeons and Price Cuts, WinBeta.org

[7] "ARK - Your Source for Intel® Product Information". *Intel® ARK (Product Specs)*.

[8] HTN_WDP_Datasheet.book

[9] Intel bringt neue Prozessoren für den Embedded-Markt auf Basis seiner 45nm-Fertigungstechnologie

[10] Intel Hard-Launches Three New Quad-core Processors, DailyTech

[11] "Intel Clovertowns step up, reduce power". TG Daily. Retrieved 2007-09-05.

[12] Quad-core Xeon Details Unveiled, DailyTech

[13] Intel Ignites Quad-Core Era, Intel News Release.

[14] Intel Readies New Xeons and Price Cuts, DailyTech

[15] Quad-Core Intel Xeon Processor E5405, Intel ARK (Automated Relational Knowledgebase)

[16] Intel® Xeon® Processor L5408, Intel ARK (Automated Relational Knowledgebase)

[17] Intel Readies 1600 MHz Front-Side Bus Xeons, DailyTech

[18] Intel Xeons Coming With 1600MHz FSB, TrustedReviews

[19] *Intel Launches First Industry-Standard Quad-Core Products for High-End, Multi-Processor Servers*, Intel Corporation, 5 September 2007, retrieved 2010-06-25

[20] Valich, Theo (25 February 2008). "Intel six-core coming in 2008". *TG Daily* (Tigervision Media). Retrieved 2008-02-26.

[21] "Chipzilla unveils six-core 'Dunnington' Xeons". *theregister.co.uk*.

[22] "Intel® Xeon® Processor E7 Family". *Intel*.

[23] AnandTech: Intel Xeon 5570: Smashing SAP records, 16 December 2008

[24] "Intel demos Moorestown, embeds Nehalem". *theregister.co.uk*.

[25] "Nehalem-EX: 2.3 billion transistors, eight cores, one die".

[26] Intel's next bunch of fun CPUs moves to 2010

[27] Timothy Prickett Morgan (10 September 2013). "Intel carves up Xeon E5-2600 v2 chips for two-socket boxes". *The Register*. Retrieved 13 September 2013.

[28] "Intel Introduces Highly Versatile Datacenter Processor Family Architected for New Era of Services". *Press release*. 10 September 2013. Retrieved 13 September 2013.

[29] Johan De Gelas (2014-09-08). "Intel Xeon E5 Version 3, Up to 18 Haswell EP Cores: The Magic Inside the Uncore". AnandTech. Retrieved 2014-09-09.

[30] STREAM benchmark, Dr. John D. McCalpin

[31] "Stream Benchmark Results - Top 20 Set". *virginia.edu*.

11.12 External links

- Server processors at the Intel website
- Intel look inside: Xeon E5 v3 (Grantley) launch, Intel, September 2014

Chapter 12

Distributed computing

"Distributed Information Processing" redirects here. For the computer company, see DIP Research.

Distributed computing is a field of computer science that studies distributed systems. A *distributed system* is a software system in which components located on networked computers communicate and coordinate their actions by passing messages.[1] The components interact with each other in order to achieve a common goal. Three significant characteristics of distributed systems are: concurrency of components, lack of a global clock, and independent failure of components.[1] Examples of distributed systems vary from SOA-based systems to massively multiplayer online games to peer-to-peer applications.

A computer program that runs in a distributed system is called a **distributed program**, and distributed programming is the process of writing such programs.[2] There are many alternatives for the message passing mechanism, including RPC-like connectors and message queues. An important goal and challenge of distributed systems is location transparency.

Distributed computing also refers to the use of distributed systems to solve computational problems. In *distributed computing*, a problem is divided into many tasks, each of which is solved by one or more computers,[3] which communicate with each other by message passing.[4]

12.1 Introduction

The word *distributed* in terms such as "distributed system", "distributed programming", and "distributed algorithm" originally referred to computer networks where individual computers were physically distributed within some geographical area.[5] The terms are nowadays used in a much wider sense, even referring to autonomous processes that run on the same physical computer and interact with each other by message passing.[4] While there is no single definition of a distributed system,[6] the following defining properties are commonly used:

- There are several autonomous computational entities, each of which has its own local memory.[7]
- The entities communicate with each other by message passing.[8]

In this article, the computational entities are called *computers* or *nodes*.

A distributed system may have a common goal, such as solving a large computational problem.[9] Alternatively, each computer may have its own user with individual needs, and the purpose of the distributed system is to coordinate the use of shared resources or provide communication services to the users.[10]

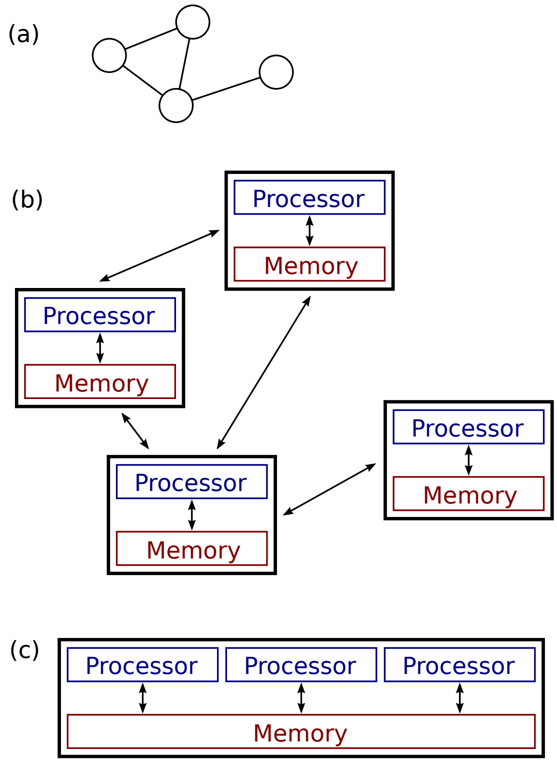
Other typical properties of distributed systems include the following:

- The system has to tolerate failures in individual computers.[11]
- The structure of the system (network topology, network latency, number of computers) is not known in advance, the system may consist of different kinds of computers and network links, and the system may change during the execution of a distributed program.[12]
- Each computer has only a limited, incomplete view of the system. Each computer may know only one part of the input.[13]

12.1.1 Architecture

Client/Server System : The Client-server architecture is a way to provide a service from a central source. There is a single server that provides a service, and many clients that communicate with the server to consume its products. In this architecture, clients and servers have different jobs. The server's job is to respond to service requests from clients, while a client's job is to use the data provided in response in order to perform some tasks.

Peer-to-Peer System : The term peer-to-peer is used to describe distributed systems in which labour is divided



(a)–(b) A distributed system.
(c) A parallel system.

among all the components of the system. All the computers send and receive data, and they all contribute some processing power and memory. As a distributed system increases in size, its capacity of computational resources increases. In a peer-to-peer system, all components of the system contribute some processing power and memory to a distributed computation.

12.2 Parallel and distributed computing

Distributed systems are groups of networked computers, which have the same goal for their work. The terms "concurrent computing", "parallel computing", and “distributed computing” have a lot of overlap, and no clear distinction exists between them.[14] The same system may be characterized both as “parallel” and “distributed"; the processors in a typical distributed system run concurrently in parallel.[15] Parallel computing may be seen as a particular tightly coupled form of distributed computing,[16] and distributed computing may be seen as a loosely coupled form of parallel computing.[6] Nevertheless, it is possible to roughly classify concurrent systems as “parallel” or “distributed” using the following criteria:

- In parallel computing, all processors may have access to a shared memory to exchange information between processors.[17]
- In distributed computing, each processor has its own private memory (distributed memory). Information is exchanged by passing messages between the processors.[18]

The figure on the right illustrates the difference between distributed and parallel systems. Figure (a) is a schematic view of a typical distributed system; as usual, the system is represented as a network topology in which each node is a computer and each line connecting the nodes is a communication link. Figure (b) shows the same distributed system in more detail: each computer has its own local memory, and information can be exchanged only by passing messages from one node to another by using the available communication links. Figure (c) shows a parallel system in which each processor has a direct access to a shared memory.

The situation is further complicated by the traditional uses of the terms parallel and distributed *algorithm* that do not quite match the above definitions of parallel and distributed *systems*; see the section Theoretical foundations below for more detailed discussion. Nevertheless, as a rule of thumb, high-performance parallel computation in a shared-memory multiprocessor uses parallel algorithms while the coordination of a large-scale distributed system uses distributed algorithms.

12.3 History

The use of concurrent processes that communicate by message-passing has its roots in operating system architectures studied in the 1960s.[19] The first widespread distributed systems were local-area networks such as Ethernet, which was invented in the 1970s.[20]

ARPANET, the predecessor of the Internet, was introduced in the late 1960s, and ARPANET e-mail was invented in the early 1970s. E-mail became the most successful application of ARPANET,[21] and it is probably the earliest example of a large-scale distributed application. In addition to ARPANET, and its successor, the Internet, other early worldwide computer networks included Usenet and FidoNet from 1980s, both of which were used to support distributed discussion systems.

The study of distributed computing became its own branch of computer science in the late 1970s and early 1980s. The first conference in the field, Symposium on Principles of Distributed Computing (PODC), dates back to 1982, and its European counterpart International Symposium on Distributed Computing (DISC) was first held in 1985.

12.4 Applications

Reasons for using distributed systems and distributed computing may include:

1. The very nature of an application may *require* the use of a communication network that connects several computers: for example, data produced in one physical location and required in another location.
2. There are many cases in which the use of a single computer would be possible in principle, but the use of a distributed system is *beneficial* for practical reasons. For example, it may be more cost-efficient to obtain the desired level of performance by using a cluster of several low-end computers, in comparison with a single high-end computer. A distributed system can provide more reliability than a non-distributed system, as there is no single point of failure. Moreover, a distributed system may be easier to expand and manage than a monolithic uniprocessor system.[22]

Ghaemi *et al.* define a **distributed query** as a query "that selects data from databases located at multiple sites in a network" and offer as an SQL example:

> SELECT ename, dname
> FROM company.emp e, company.dept@sales.goods d
> WHERE e.deptno = d.deptno[23]

12.5 Examples

Examples of distributed systems and applications of distributed computing include the following:[24]

- Telecommunication networks:
  - Telephone networks and cellular networks
  - Computer networks such as the Internet
  - Wireless sensor networks
  - Routing algorithms
- Network applications:
  - World wide web and peer-to-peer networks
  - Massively multiplayer online games and virtual reality communities
  - Distributed databases and distributed database management systems
  - Network file systems
  - Distributed information processing systems such as banking systems and airline reservation systems
- Real-time process control:
  - Aircraft control systems
  - Industrial control systems
- Parallel computation:
  - Scientific computing, including cluster computing and grid computing and various volunteer computing projects; see the list of distributed computing projects
  - Distributed rendering in computer graphics

12.6 Theoretical foundations

Main article: Distributed algorithm

12.6.1 Models

Many tasks that we would like to automate by using a computer are of question–answer type: we would like to ask a question and the computer should produce an answer. In theoretical computer science, such tasks are called computational problems. Formally, a computational problem consists of *instances* together with a *solution* for each instance. Instances are questions that we can ask, and solutions are desired answers to these questions.

Theoretical computer science seeks to understand which computational problems can be solved by using a computer (computability theory) and how efficiently (computational complexity theory). Traditionally, it is said that a problem can be solved by using a computer if we can design an algorithm that produces a correct solution for any given instance. Such an algorithm can be implemented as a computer program that runs on a general-purpose computer: the program reads a problem instance from input, performs some computation, and produces the solution as output. Formalisms such as random access machines or universal Turing machines can be used as abstract models of a sequential general-purpose computer executing such an algorithm.

The field of concurrent and distributed computing studies similar questions in the case of either multiple computers, or a computer that executes a network of interacting processes: which computational problems can be solved in such a network and how efficiently? However, it is not at all obvious what is meant by "solving a problem" in the case of a concurrent or distributed system: for example, what is the task of the algorithm designer, and what is the concurrent or distributed equivalent of a sequential general-purpose computer?

The discussion below focuses on the case of multiple computers, although many of the issues are the same for concurrent processes running on a single computer.

Three viewpoints are commonly used:

Parallel algorithms in shared-memory model

- All computers have access to a shared memory. The algorithm designer chooses the program executed by each computer.
- One theoretical model is the parallel random access machines (PRAM) that are used.[25] However, the classical PRAM model assumes synchronous access to the shared memory.
- A model that is closer to the behavior of real-world multiprocessor machines and takes into account the use of machine instructions, such as Compare-and-swap (CAS), is that of *asynchronous shared memory*. There is a wide body of work on this model, a summary of which can be found in the literature.[26][27]

Parallel algorithms in message-passing model

- The algorithm designer chooses the structure of the network, as well as the program executed by each computer.
- Models such as Boolean circuits and sorting networks are used.[28] A Boolean circuit can be seen as a computer network: each gate is a computer that runs an extremely simple computer program. Similarly, a sorting network can be seen as a computer network: each comparator is a computer.

Distributed algorithms in message-passing model

- The algorithm designer only chooses the computer program. All computers run the same program. The system must work correctly regardless of the structure of the network.
- A commonly used model is a graph with one finite-state machine per node.

In the case of distributed algorithms, computational problems are typically related to graphs. Often the graph that describes the structure of the computer network *is* the problem instance. This is illustrated in the following example.

12.6.2 An example

Consider the computational problem of finding a coloring of a given graph G. Different fields might take the following approaches:

Centralized algorithms

- The graph G is encoded as a string, and the string is given as input to a computer. The computer program finds a coloring of the graph, encodes the coloring as a string, and outputs the result.

Parallel algorithms

- Again, the graph G is encoded as a string. However, multiple computers can access the same string in parallel. Each computer might focus on one part of the graph and produce a coloring for that part.
- The main focus is on high-performance computation that exploits the processing power of multiple computers in parallel.

Distributed algorithms

- The graph G is the structure of the computer network. There is one computer for each node of G and one communication link for each edge of G. Initially, each computer only knows about its immediate neighbors in the graph G; the computers must exchange messages with each other to discover more about the structure of G. Each computer must produce its own color as output.
- The main focus is on coordinating the operation of an arbitrary distributed system.

While the field of parallel algorithms has a different focus than the field of distributed algorithms, there is a lot of interaction between the two fields. For example, the Cole–Vishkin algorithm for graph coloring[29] was originally presented as a parallel algorithm, but the same technique can also be used directly as a distributed algorithm.

Moreover, a parallel algorithm can be implemented either in a parallel system (using shared memory) or in a distributed system (using message passing).[30] The traditional boundary between parallel and distributed algorithms (choose a suitable network vs. run in any given network) does not lie in the same place as the boundary between parallel and distributed systems (shared memory vs. message passing).

12.6.3 Complexity measures

In parallel algorithms, yet another resource in addition to time and space is the number of computers. Indeed, often there is a trade-off between the running time and the number of computers: the problem can be solved faster if there are more computers running in parallel (see speedup). If a decision problem can be solved in polylogarithmic time by using a polynomial number of processors, then the problem is said to be in the class NC.[31] The class NC can be defined equally well by using the PRAM formalism or Boolean circuits – PRAM machines can simulate Boolean circuits efficiently and vice versa.[32]

In the analysis of distributed algorithms, more attention is usually paid on communication operations than computational steps. Perhaps the simplest model of distributed computing is a synchronous system where all nodes operate in a lockstep fashion. During each *communication round*, all nodes in parallel (1) receive the latest messages from their neighbours, (2) perform arbitrary local computation, and (3) send new messages to their neighbours. In such systems, a central complexity measure is the number of synchronous communication rounds required to complete the task.[33]

This complexity measure is closely related to the diameter of the network. Let *D* be the diameter of the network. On the one hand, any computable problem can be solved trivially in a synchronous distributed system in approximately 2*D* communication rounds: simply gather all information in one location (*D* rounds), solve the problem, and inform each node about the solution (*D* rounds).

On the other hand, if the running time of the algorithm is much smaller than *D* communication rounds, then the nodes in the network must produce their output without having the possibility to obtain information about distant parts of the network. In other words, the nodes must make globally consistent decisions based on information that is available in their *local neighbourhood*. Many distributed algorithms are known with the running time much smaller than *D* rounds, and understanding which problems can be solved by such algorithms is one of the central research questions of the field.[34]

Other commonly used measures are the total number of bits transmitted in the network (cf. communication complexity).

12.6.4 Other problems

Traditional computational problems take the perspective that we ask a question, a computer (or a distributed system) processes the question for a while, and then produces an answer and stops. However, there are also problems where we do not want the system to ever stop. Examples of such problems include the dining philosophers problem and other similar mutual exclusion problems. In these problems, the distributed system is supposed to continuously coordinate the use of shared resources so that no conflicts or deadlocks occur.

There are also fundamental challenges that are unique to distributed computing. The first example is challenges that are related to *fault-tolerance*. Examples of related problems include consensus problems,[35] Byzantine fault tolerance,[36] and self-stabilisation.[37]

A lot of research is also focused on understanding the *asynchronous* nature of distributed systems:

- Synchronizers can be used to run synchronous algorithms in asynchronous systems.[38]
- Logical clocks provide a causal happened-before ordering of events.[39]
- Clock synchronization algorithms provide globally consistent physical time stamps.[40]

12.6.5 Properties of distributed systems

So far the focus has been on *designing* a distributed system that solves a given problem. A complementary research problem is *studying* the properties of a given distributed system.

The halting problem is an analogous example from the field of centralised computation: we are given a computer program and the task is to decide whether it halts or runs forever. The halting problem is undecidable in the general case, and naturally understanding the behaviour of a computer network is at least as hard as understanding the behaviour of one computer.

However, there are many interesting special cases that are decidable. In particular, it is possible to reason about the behaviour of a network of finite-state machines. One example is telling whether a given network of interacting (asynchronous and non-deterministic) finite-state machines can reach a deadlock. This problem is PSPACE-complete,[41] i.e., it is decidable, but it is not likely that there is an efficient (centralised, parallel or distributed) algorithm that solves the problem in the case of large networks.

12.7 Coordinator election

Coordinator election (sometimes called **leader election**) is the process of designating a single process as the organizer of some task distributed among several computers (nodes). Before the task is begun, all network nodes are either unaware which node will serve as the “coordinator” (or leader) of the task, or unable to communicate with the current coordinator. After a coordinator election algorithm has been run, however, each node throughout the network recognizes a particular, unique node as the task coordinator.

The network nodes communicate among themselves in order to decide which of them will get into the “coordinator” state. For that, they need some method in order to break the symmetry among them. For example, if each node has unique and comparable identities, then the nodes can compare their identities, and decide that the node with the highest identity is the coordinator.

The definition of this problem is often attributed to LeLann, who formalized it as a method to create a new token in a token ring network in which the token has been lost.

Coordinator election algorithms are designed to be economical in terms of total bytes transmitted, and time. The algorithm suggested by Gallager, Humblet, and Spira [42] for general undirected graphs has had a strong impact on the design of distributed algorithms in general, and won the Dijkstra Prize for an influential paper in distributed computing.

Many other algorithms were suggested for different kind of network graphs, such as undirected rings, unidirectional rings, complete graphs, grids, directed Euler graphs, and others. A general method that decouples the issue of the graph family from the design of the coordinator election algorithm was suggested by Korach, Kutten, and Moran.[43]

In order to perform coordination, distributed systems employ the concept of coordinators. The coordinator election problem is to choose a process from among a group of processes on different processors in a distributed system to act as the central coordinator. Several central coordinator election algorithms exist.[44]

12.7.1 Bully algorithm

When using the Bully algorithm, any process sends a message to the current coordinator. If there is no response within a given time limit, the process tries to elect itself as leader.

12.7.2 Chang and Roberts algorithm

The Chang and Roberts algorithm (or "Ring Algorithm") is a ring-based election algorithm used to find a process with the largest unique identification number .

12.8 Architectures

Various hardware and software architectures are used for distributed computing. At a lower level, it is necessary to interconnect multiple CPUs with some sort of network, regardless of whether that network is printed onto a circuit board or made up of loosely coupled devices and cables. At a higher level, it is necessary to interconnect processes running on those CPUs with some sort of communication system.

Distributed programming typically falls into one of several basic architectures or categories: client–server, 3-tier architecture, *n*-tier architecture, distributed objects, loose coupling, or tight coupling.

- Client–server: Smart client code contacts the server for data then formats and displays it to the user. Input at the client is committed back to the server when it represents a permanent change.
- 3-tier architecture: Three tier systems move the client intelligence to a middle tier so that stateless clients can be used. This simplifies application deployment. Most web applications are 3-Tier.
- *n*-tier architecture: *n*-tier refers typically to web applications which further forward their requests to other enterprise services. This type of application is the one most responsible for the success of application servers.
- highly coupled (clustered): refers typically to a cluster of machines that closely work together, running a shared process in parallel. The task is subdivided in parts that are made individually by each one and then put back together to make the final result.
- Peer-to-peer: an architecture where there is no special machine or machines that provide a service or manage the network resources. Instead all responsibilities are uniformly divided among all machines, known as peers. Peers can serve both as clients and servers.
- Space based: refers to an infrastructure that creates the illusion (virtualization) of one single address-space. Data are transparently replicated according to application needs. Decoupling in time, space and reference is achieved.

Another basic aspect of distributed computing architecture is the method of communicating and coordinating work among concurrent processes. Through various message passing protocols, processes may communicate directly with one another, typically in a master/slave relationship. Alternatively, a "database-centric" architecture can enable distributed computing to be done without any form of direct inter-process communication, by utilizing a shared database.[45]

12.9 See also

- BOINC
- Plan 9 from Bell Labs
- Inferno
- Code mobility
- Decentralized computing
- Distributed algorithmic mechanism design
- Distributed cache
- Distributed operating system

- Edsger W. Dijkstra Prize in Distributed Computing
- Folding@home
- Grid computing
- Jungle computing
- Layered queueing network
- Library Oriented Architecture - LOA
- List of distributed computing conferences
- List of distributed computing projects
- List of important publications in concurrent, parallel, and distributed computing
- Parallel distributed processing
- Parallel programming model
- Service-Oriented Architecture - SOA
- Volunteer computing

12.10 Notes

[1] Coulouris, George; Jean Dollimore; Tim Kindberg; Gordon Blair (2011). *Distributed Systems: Concepts and Design (5th Edition)*. Boston: Addison-Wesley. ISBN 0-132-14301-1.

[2] Andrews (2000). Dolev (2000). Ghosh (2007), p. 10.

[3] Godfrey (2002).

[4] Andrews (2000), p. 291–292. Dolev (2000), p. 5.

[5] Lynch (1996), p. 1.

[6] Ghosh (2007), p. 10.

[7] Andrews (2000), p. 8–9, 291. Dolev (2000), p. 5. Ghosh (2007), p. 3. Lynch (1996), p. xix, 1. Peleg (2000), p. xv.

[8] Andrews (2000), p. 291. Ghosh (2007), p. 3. Peleg (2000), p. 4.

[9] Ghosh (2007), p. 3–4. Peleg (2000), p. 1.

[10] Ghosh (2007), p. 4. Peleg (2000), p. 2.

[11] Ghosh (2007), p. 4, 8. Lynch (1996), p. 2–3. Peleg (2000), p. 4.

[12] Lynch (1996), p. 2. Peleg (2000), p. 1.

[13] Ghosh (2007), p. 7. Lynch (1996), p. xix, 2. Peleg (2000), p. 4.

[14] Ghosh (2007), p. 10. Keidar (2008).

[15] Lynch (1996), p. xix, 1–2. Peleg (2000), p. 1.

[16] Peleg (2000), p. 1.

[17] Papadimitriou (1994), Chapter 15. Keidar (2008).

[18] See references in Introduction.

[19] Andrews (2000), p. 348.

[20] Andrews (2000), p. 32.

[21] Peter (2004), The history of email.

[22] Elmasri & Navathe (2000), Section 24.1.2.

[23] Ghaemi, Reza; Milani Fard, Amin; Tabatabaee, Hamid; Sadeghizadeh, Mahdi (2008). "Evolutionary Query Optimization for Heterogeneous Distributed Database Systems". *World Academy of Science, Engineering and Technology* (World Academy of Science, Engineering and Technology) (19): 43–49. Retrieved 2013-07-16.

[24] Andrews (2000), p. 10–11. Ghosh (2007), p. 4–6. Lynch (1996), p. xix, 1. Peleg (2000), p. xv. Elmasri & Navathe (2000), Section 24.

[25] Cormen, Leiserson & Rivest (1990), Section 30.

[26] Herlihy & Shavit (2008), Chapters 2-6.

[27] Lynch (1996)

[28] Cormen, Leiserson & Rivest (1990), Sections 28 and 29.

[29] Cole & Vishkin (1986). Cormen, Leiserson & Rivest (1990), Section 30.5.

[30] Andrews (2000), p. ix.

[31] Arora & Barak (2009), Section 6.7. Papadimitriou (1994), Section 15.3.

[32] Papadimitriou (1994), Section 15.2.

[33] Lynch (1996), p. 17–23.

[34] Peleg (2000), Sections 2.3 and 7. Linial (1992). Naor & Stockmeyer (1995).

[35] Lynch (1996), Sections 5–7. Ghosh (2007), Chapter 13.

[36] Lynch (1996), p. 99–102. Ghosh (2007), p. 192–193.

[37] Dolev (2000). Ghosh (2007), Chapter 17.

[38] Lynch (1996), Section 16. Peleg (2000), Section 6.

[39] Lynch (1996), Section 18. Ghosh (2007), Sections 6.2–6.3.

[40] Ghosh (2007), Section 6.4.

[41] Papadimitriou (1994), Section 19.3.

[42] R. G. Gallager, P. A. Humblet, and P. M. Spira (January 1983). "A Distributed Algorithm for Minimum-Weight Spanning Trees". *ACM Transactions on Programming Languages and Systems* **5** (1): 66–77. doi:10.1145/357195.357200.

[43] Ephraim Korach, Shay Kutten, Shlomo Moran (1990). "A Modular Technique for the Design of Efficient Distributed Leader Finding Algorithms". *ACM Transactions on Programming Languages and Systems* **12** (1): 84–101. doi:10.1145/77606.77610.

[44] Hamilton, Howard. "Distributed Algorithms". Retrieved 2013-03-03.

[45] Lind P, Alm M (2006), "A database-centric virtual chemistry system", *J Chem Inf Model* **46** (3): 1034–9, doi:10.1021/ci050360b, PMID 16711722.

12.11 References

Books

- Andrews, Gregory R. (2000), *Foundations of Multithreaded, Parallel, and Distributed Programming*, Addison–Wesley, ISBN 0-201-35752-6.
- Arora, Sanjeev; Barak, Boaz (2009), *Computational Complexity – A Modern Approach*, Cambridge, ISBN 978-0-521-42426-4.
- Cormen, Thomas H.; Leiserson, Charles E.; Rivest, Ronald L. (1990), *Introduction to Algorithms* (1st ed.), MIT Press, ISBN 0-262-03141-8.
- Dolev, Shlomi (2000), *Self-Stabilization*, MIT Press, ISBN 0-262-04178-2.
- Elmasri, Ramez; Navathe, Shamkant B. (2000), *Fundamentals of Database Systems* (3rd ed.), Addison–Wesley, ISBN 0-201-54263-3.
- Ghosh, Sukumar (2007), *Distributed Systems – An Algorithmic Approach*, Chapman & Hall/CRC, ISBN 978-1-58488-564-1.
- Lynch, Nancy A. (1996), *Distributed Algorithms*, Morgan Kaufmann, ISBN 1-55860-348-4.
- Herlihy, Maurice P.; Shavit, Nir N. (2008), *The Art of Multiprocessor Programming*, Morgan Kaufmann, ISBN 0-12-370591-6.
- Papadimitriou, Christos H. (1994), *Computational Complexity*, Addison–Wesley, ISBN 0-201-53082-1.
- Peleg, David (2000), *Distributed Computing: A Locality-Sensitive Approach*, SIAM, ISBN 0-89871-464-8.

Articles

- Cole, Richard; Vishkin, Uzi (1986), "Deterministic coin tossing with applications to optimal parallel list ranking", *Information and Control* **70** (1): 32–53, doi:10.1016/S0019-9958(86)80023-7.
- Keidar, Idit (2008), "Distributed computing column 32 – The year in review", *ACM SIGACT News* **39** (4): 53–54, doi:10.1145/1466390.1466402.
- Linial, Nathan (1992), "Locality in distributed graph algorithms", *SIAM Journal on Computing* **21** (1): 193–201, doi:10.1137/0221015.
- Naor, Moni; Stockmeyer, Larry (1995), "What can be computed locally?", *SIAM Journal on Computing* **24** (6): 1259–1277, doi:10.1137/S0097539793254571.

Web sites

- Godfrey, Bill (2002). "A primer on distributed computing".
- Peter, Ian (2004). "Ian Peter's History of the Internet". Retrieved 2009-08-04.

12.12 Further reading

Books

- Coulouris, George et al. (2011), *Distributed Systems: Concepts and Design (5th Edition)*, Addison-Wesley ISBN 0-132-14301-1.
- Attiya, Hagit and Welch, Jennifer (2004), *Distributed Computing: Fundamentals, Simulations, and Advanced Topics*, Wiley-Interscience ISBN 0-471-45324-2.
- Faber, Jim (1998), *Java Distributed Computing*, O'Reilly: Java Distributed Computing by Jim Faber, 1998
- Garg, Vijay K. (2002), *Elements of Distributed Computing*, Wiley-IEEE Press ISBN 0-471-03600-5.
- Tel, Gerard (1994), *Introduction to Distributed Algorithms*, Cambridge University Press
- Chandy, Mani et al., *Parallel Program Design*

Articles

- Keidar, Idit; Rajsbaum, Sergio, eds. (2000–2009), "Distributed computing column", *ACM SIGACT News*.
- Birrell, A. D.; Levin, R.; Schroeder, M. D.; Needham, R. M. (April 1982). "Grapevine: An exercise in distributed computing". *Communications of the ACM* **25** (4): 260–274. doi:10.1145/358468.358487.

Conference Papers

- C. Rodríguez, M. Villagra and B. Barán, Asynchronous team algorithms for Boolean Satisfiability, Bionetics2007, pp. 66–69, 2007.

12.13 External links

- Distributed computing at DMOZ
- Distributed computing journals at DMOZ

Chapter 13

Service-oriented architecture

A **service-oriented architecture** (**SOA**) is a design pattern in which application components provide services to other components via a communications protocol, typically over a network. The principles of service-orientation are independent of any vendor, product or technology.[1]

A service is a self-contained unit of functionality, such as retrieving an online bank statement.[2] By that definition, a service is a discretely invocable operation. However, in the Web Services Description Language (WSDL), a "service" is an interface definition that may list several discrete services or operations. Elsewhere, the term is used for a component that is encapsulated behind an interface.

Services can be combined to provide the functionality of a large software application.[3] SOA makes it easier for software components on computers connected over a network to cooperate. Every computer can run any number of services, and each service is built in a way that ensures that the service can exchange information with any other service in the network without human interaction and without the need to make changes to the underlying program itself.

13.1 Definitions

The OASIS group[4] and the Open Group[5] have both created formal definitions. OASIS defines SOA as:

> A paradigm for organizing and utilizing distributed capabilities that may be under the control of different ownership domains. It provides a uniform means to offer, discover, interact with and use capabilities to produce desired effects consistent with measurable preconditions and expectations.

The Open Group's definition is:

> Service-Oriented Architecture (SOA) is an architectural style that supports service-orientation.
>
> Service-orientation is a way of thinking in terms of services and service-based development and the outcomes of services.
>
> A service:
>
> Is a logical representation of a repeatable business activity that has a specified outcome (e.g., check customer credit, provide weather data, consolidate drilling reports)
>
> Is self-contained
>
> May be composed of other services
>
> Is a "black box" to consumers of the service

13.2 Overview

Services are **unassociated**, loosely coupled units of functionality that are self-contained. Each service implements at least one action, such as submitting an online application for an account, retrieving an online bank statement or modifying an online booking or airline ticket order. Within an SOA, services use defined protocols that describe how services pass and parse messages using description metadata, which in sufficient details describes not only the characteristics of these services, but also the data that drives them. Programmers have made extensive use of XML in SOA to structure data that they wrap in a nearly exhaustive description-container. Analogously, the Web Services Description Language (WSDL) typically describes the services themselves, while SOAP (originally Simple Object Access Protocol) describes the communications protocols. SOA depends on data and services that are described by metadata that should meet the following two criteria:

1. The metadata should be provided in a form that software systems can use to configure dynamically by discovery and incorporation of defined services, and also to maintain coherence and integrity. For example, metadata could be used by other applications, like a catalogue, to perform auto discovery of services without modifying the functional contract of a service.

2. The metadata should be provided in a form that system designers can understand and manage with a reasonable expenditure of cost and effort.

The purpose of SOA is to allow users to combine together fairly large chunks of functionality to form *ad hoc* applications built almost entirely from existing software services. The larger the chunks, the fewer the interfaces required to implement any given set of functionality; however, very large chunks of functionality may not prove sufficiently granular for easy reuse. Each interface brings with it some amount of processing overhead, so there is a performance consideration in choosing the granularity of services.

SOA as an architecture relies on service-orientation as its fundamental design principle. If a service presents a simple interface that abstracts away its underlying complexity, then users can access independent services without knowledge of the service's platform implementation.[6]

13.3 SOA framework

SOA-based solutions endeavour to enable business objectives while building an enterprise-quality system. SOA architecture is viewed as five horizontal layers:[7]

1. Consumer Interface Layer - These are GUI for end users or apps accessing apps/service interfaces.
2. Business Process Layer - These are choreographed services representing business use-cases in terms of applications.
3. Services - Services are consolidated together for whole enterprise in service inventory.
4. Service Components - The components used to build the services, like functional and technical libraries, technological interfaces etc.
5. Operational Systems - This layer contains the data models, enterprise data repository, technological platforms etc.

There are four cross-cutting vertical layers, each of which are applied to and supported by each of horizontal layers:

1. Integration Layer - starts with platform integration (protocols support), data integration, service integration, application integration, leading to enterprise application integration supporting B2B and B2C.
2. Quality of Service - Security, availability, performance etc. constitute the quality of service which are configured based on required SLAs, OLAs.
3. Informational - provide business information.
4. Governance - IT strategy is governed to each horizontal layer to achieve required operating and capability model.

13.4 Design concept

SOA is based on the concept of a *service*. Depending on the service design approach taken, each SOA service is designed to perform one or more activities by implementing one or more service operations. As a result, each service is built as a discrete piece of code. This makes it possible to reuse the code in different ways throughout the application by changing only the way an individual service interoperates with other services that make up the application, versus making code changes to the service itself. SOA design principles are used during software development and integration.

SOA generally provides a way for consumers of services, such as web-based applications, to be aware of available SOA-based services. For example, several disparate departments within a company may develop and deploy SOA services in different implementation languages; their respective clients will benefit from a well-defined interface to access them.

SOA defines how to integrate widely disparate applications for a Web-based environment and uses multiple implementation platforms. Rather than defining an API, SOA defines the interface in terms of protocols and functionality. An *endpoint* is the entry point for such a SOA implementation.

Service-orientation requires *loose coupling* of services with operating systems and other technologies that underlie applications. SOA separates functions into distinct units, or services,[8] which developers make accessible over a network in order to allow users to combine and reuse them in the production of applications. These services and their corresponding consumers communicate with each other by passing data in a well-defined, shared format, or by coordinating an activity between two or more services.[9]

For some, SOA can be seen in a continuum from older concepts of distributed computing[8][10] and modular programming, through SOA, and on to current practices of mashups, SaaS, and cloud computing (which some see as the offspring of SOA).[11]

13.5 Principles

There are no industry standards relating to the exact composition of a service-oriented architecture, although many industry sources have published their own principles. Some of the principles published[12][13][14][15] include the following:

- *Standardized service contract*: Services adhere to a communications agreement, as defined collectively by one or more service-description documents.
- *Service loose coupling*: Services maintain a relationship that minimizes dependencies and only requires that they maintain an awareness of each other.
- *Service abstraction*: Beyond descriptions in the service contract, services hide logic from the outside world.
- *Service reusability*: Logic is divided into services with the intention of promoting reuse.
- *Service autonomy*: Services have control over the logic they encapsulate, from a Design-time and a Run-time perspective.
- *Service statelessness*: Services minimize resource consumption by deferring the management of state information when necessary[16]
- *Service discoverability*: Services are supplemented with communicative meta data by which they can be effectively discovered and interpreted.
- *Service composability*: Services are effective composition participants, regardless of the size and complexity of the composition.
- *Service granularity*: A design consideration to provide optimal scope and right granular level of the business functionality in a service operation.
- *Service normalization*: Services are decomposed and/or consolidated to a level of normal form to minimize redundancy. In some cases, services are denormalized for specific purposes, such as performance optimization, access, and aggregation.[17]
- *Service optimization*: All else being equal, high-quality services are generally preferable to low-quality ones.
- *Service relevance*: Functionality is presented at a granularity recognized by the user as a meaningful service.
- *Service encapsulation*: Many services are consolidated for use under the SOA. Often such services were not planned to be under SOA.
- *Service location transparency*: This refers to the ability of a service consumer to invoke a service regardless of its actual location in the network. This also recognizes the discoverability property (one of the core principle of SOA) and the right of a consumer to access the service. Often, the idea of service virtualization also relates to location transparency. This is where the consumer simply calls a logical service while a suitable SOA-enabling runtime infrastructure component, commonly a service bus, maps this logical service call to a physical service.

13.5.1 Service architecture

This is the physical design of an individual service that encompasses all the resources used by a service. This would normally include databases, software components, legacy systems, identity stores, XML schemas and any backing stores, e.g. shared directories. It is also beneficial to include any service agents employed by the service, as any change in these service agents would affect the message processing capabilities of the service.

The (standardized service contract) design principle, keeps service *contracts* independent from their implementation. The service contract needs to be documented to formalize the required processing resources by the individual service capabilities. Although it is beneficial to document details about the service architecture, the service abstraction design principle dictates that any internal details about the service are invisible to its consumers so that they do not develop any unstated couplings. The service architecture serves as a point of reference for evolving the service or gauging the impact of any change in the service.

13.5.2 Service composition architecture

One of the core characteristics of services developed using the service-orientation design paradigm is that they are composition-centric. Services with this characteristic can potentially address novel requirements by recomposing the same services in different configurations. Service composition architecture is itself a composition of the individual architectures of the participating services. In the light of the Service Abstraction principle, this type of architecture only documents the service contract and any published service-level agreement (SLA); internal details of each service are not included.

If a service composition is a part of another (parent) composition, the parent composition can also be referenced in the child service composition. The design of service composition also includes any alternate paths, such as error conditions, which may introduce new services into the current service composition.

Service composition is also a key technique in software integration, including enterprise software integration, business process composition and workflow composition.

13.5.3 Service inventory architecture

A service inventory is composed of services that automate business processes. It is important to account for the combined processing requirements of all services within the service inventory. Documenting the requirements of services, independently from the business processes that they automate, helps identify processing bottlenecks. The service inventory architecture is documented from the service inventory blueprint, so that service candidates[18] can be redesigned before their implementation.

13.5.4 Service-oriented enterprise architecture

This umbrella architecture incorporates service, composition, and inventory architectures, plus any enterprise-wide technological resources accessed by these architectures e.g. an ERP system. This can be further supplemented by including enterprise-wide standards that apply to the aforementioned architecture types. Any segments of the enterprise that are not service-oriented can also be documented in order to consider transformation requirements if a service needs to communicate with the business processes automated by such segments.SOA's main goal is to deliver agility to business

13.6 Web services approach

Web services can implement a service-oriented architecture.[19] They make functional building-blocks accessible over standard Internet protocols independent of platforms and programming languages. These services can represent either new applications or just wrappers around existing legacy systems to make them network-enabled.

Each SOA building block can play one or both of two roles:

1. *Service provider*: The service provider creates a web service and possibly publishes its interface and access information to the service registry. Each provider must decide which services to expose, how to make trade-offs between security and easy availability, how to price the services, or (if no charges apply) how/whether to exploit them for other value. The provider also has to decide what category the service should be listed in for a given broker service and what sort of trading partner agreements are required to use the service. It registers what services are available within it, and lists all the potential service recipients. The implementer of the broker then decides the scope of the broker. Public brokers are available through the Internet, while private brokers are only accessible to a limited audience, for example, users of a company intranet. Furthermore, the amount of the offered information has to be decided. Some brokers specialize in many listings. Others offer high levels of trust in the listed services. Some cover a broad landscape of services and others focus within an industry. Some brokers catalog other brokers. Depending on the business model, brokers can attempt to maximize look-up requests, number of listings or accuracy of the listings. The Universal Description Discovery and Integration (UDDI) specification defines a way to publish and discover information about Web services. Other service broker technologies include (for example) ebXML (Electronic Business using eXtensible Markup Language) and those based on the ISO/IEC 11179 Metadata Registry (MDR) standard.

2. *Service consumer*: The service consumer or web service client locates entries in the broker registry using various find operations and then binds to the service provider in order to invoke one of its web services. Whichever service the service-consumers need, they have to take it into the brokers, bind it with respective service and then use it. They can access multiple services if the service provides multiple services.

13.7 Web service protocols

See also: List of web service protocols

Implementers commonly build SOAs using web services standards (for example, SOAP) that have gained broad industry acceptance after recommendation of Version 1.2 from the W3C[20] (World Wide Web Consortium) in 2003. These standards (also referred to as web service specifications) also provide greater interoperability and some protection from lock-in to proprietary vendor software. One can, however, implement SOA using any service-based technology, such as Jini, CORBA or REST.

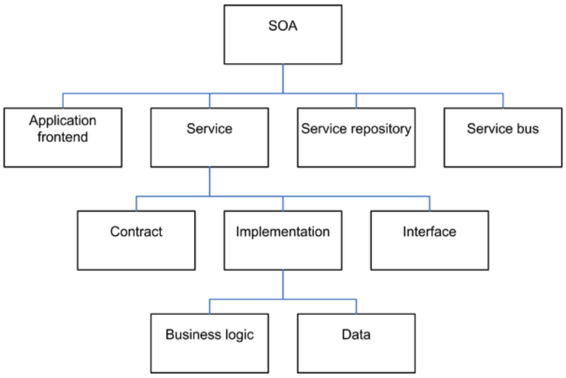
13.8 Other SOA concepts

Architectures can operate independently of specific technologies and can therefore be implemented using a wide range of technologies, including:

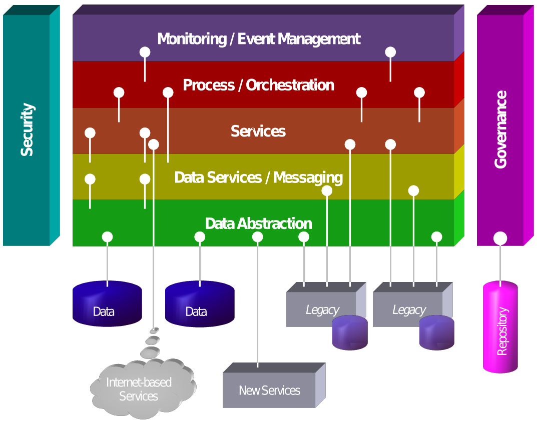
- SOAP, RPC
- REST
- DCOM
- CORBA
- Web services
- DDS
- Java RMI
- WCF (Microsoft's implementation of web services now forms a part of WCF)
- Apache Thrift

- SORCER

Implementations can use one or more of these protocols and, for example, might use a file-system mechanism to communicate data conforming to a defined interface specification between processes conforming to the SOA concept. The key is independent services with defined interfaces that can be called to perform their tasks in a standard way, without a service having foreknowledge of the calling application, and without the application having or needing knowledge of how the service actually performs its tasks.



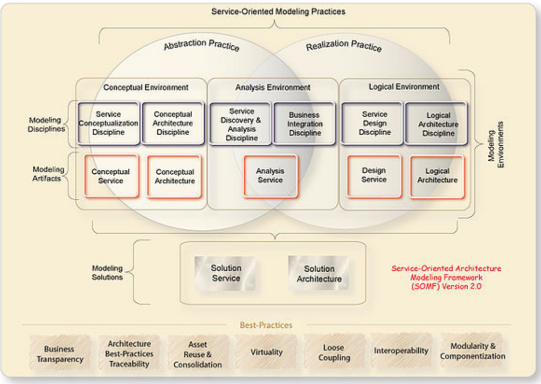
Elements of SOA, by Dirk Krafzig, Karl Banke, and Dirk Slama[21]



SOA meta-model, The Linthicum Group, 2007

SOA enables the development of applications that are built by combining loosely coupled and interoperable services.[22]

These services inter-operate based on a formal definition (or contract, e.g., WSDL) that is independent of the underlying platform and programming language. The interface definition hides the implementation of the language-specific service. SOA-based systems can therefore function independently of development technologies and platforms (such as Java, .NET, etc.). Services written in C# running on .NET platforms and services written in Java running on Java EE platforms, for example, can both be consumed by a common composite application (or client). Applications running on either platform can also consume services running on the other as web services that facilitate reuse. Managed environments can also wrap COBOL legacy systems and present them as software services. This has extended the useful life of many core legacy systems indefinitely, no matter what language they originally used.


Service-Oriented Modeling Practices
Abstraction Practice
Realization Practice
Conceptual Environment
Analysis Environment
Logical Environment
Modeling Disciplines
Service Conceptualization Discipline
Conceptual Architecture Discipline
Service Discovery & Analysis Discipline
Business Integration Discipline
Service Design Discipline
Logical Architecture Discipline
Modeling Artifacts
Conceptual Service
Conceptual Architecture
Analysis Service
Design Service
Logical Architecture
Modeling Environments
Modeling Solutions
Solution Service
Solution Architecture
Service-Oriented Architecture Modeling Framework (SOMF) Version 2.0
Best-Practices
Business Transparency
Architecture Best-Practices Traceability
Asset Reuse & Consolidation
Virtuality
Loose Coupling
Interoperability
Modularity & Componentization

Service-Oriented Modeling Framework (SOMF) Version 2.0

SOA can support integration and consolidation activities within complex enterprise systems, but SOA does not specify or provide a methodology or framework for documenting capabilities or services.

We can distinguish the Service Object-Oriented Architecture (SOOA), where service providers are network (call/response) objects accepting remote invocations, from the Service Protocol Oriented Architecture (SPOA), where a communication (read/write) protocol is fixed and known beforehand by the provider and requestor. Based on that protocol and a service description obtained from the service registry, the requestor can bind to the service provider by creating own proxy used for remote communication over the fixed protocol. If a service provider registers its service description by name, the requestors have to know the name of the service beforehand. In SOOA, a proxy—an object implementing the same service interfaces as its service provider—is registered with the registries and it is always ready for use by requestors. Thus, in SOOA, the service provider owns and publishes the proxy as the active surrogate object with a codebase annotation, e.g., URLs to the code defining proxy behavior (Jini ERI). In SPOA, by contrast, a passive service description is registered (e.g., an XML document in WSDL for Web services, or an interface description in IDL for CORBA); the requestor then has to generate the proxy (a stub forwarding calls to a provider) based on a service description and the fixed communication protocol (e.g., SOAP in Web services, IIOP in CORBA). This is referred to as a bind operation. The proxy binding operation is not required in SOOA since the requestor holds the active surrogate object obtained via the registry. The surrogate object is already bound to the provider that registered it with its appropriate network configuration and its code annotations. Web services, OGSA, RMI, and CORBA services cannot change the communication protocol between requestors and providers while the SOOA approach is protocol neutral.[23]

High-level languages such as BPEL and specifications such as WS-CDL and WS-Coordination extend the service concept by providing a method of defining and supporting orchestration of fine-grained services into more coarse-grained business services, which architects can in turn incorporate into workflows and business processes implemented in composite applications or portals.[24]

Service-oriented modeling[8] is a SOA framework that identifies the various disciplines that guide SOA practitioners to conceptualize, analyze, design, and architect their service-oriented assets. The Service-oriented modeling framework (SOMF) offers a modeling language and a work structure or "map" depicting the various components that contribute to a successful service-oriented modeling approach. It illustrates the major elements that identify the "what to do" aspects of a service development scheme. The model enables practitioners to craft a project plan and to identify the milestones of a service-oriented initiative. SOMF also provides a common modeling notation to address alignment between business and IT organizations.

13.9 Organizational benefits

Some enterprise architects believe that SOA can help businesses respond more quickly and more cost-effectively to changing market conditions.[25] This style of *architecture* promotes reuse at the macro (service) level rather than micro (classes) level. It can also simplify interconnection to—and usage of—existing IT (legacy) assets.

With SOA, the idea is that an organization can look at a problem holistically. A business has more overall control. Theoretically there would not be a mass of developers using whatever tool sets might please them. But rather they would be coding to a standard that is set within the business. They can also develop enterprise-wide SOA that encapsulates a business-oriented infrastructure. SOA has also been illustrated as a highway system providing efficiency for car drivers. The point being that if everyone had a car, but there was no highway anywhere, things would be limited and disorganized, in any attempt to get anywhere quickly or efficiently. IBM Vice President of Web Services Michael Liebow says that SOA "builds highways".[26]

In some respects, SOA could be regarded as an architectural evolution rather than as a revolution. It captures many of the best practices of previous software architectures. In communications systems, for example, little development of solutions that use truly static bindings to talk to other equipment in the network has taken place. By formally embracing a SOA approach, such systems can position themselves to stress the importance of well-defined, highly inter-operable interfaces.[27]

Some have questioned whether SOA simply revives concepts like modular programming (1970s), event-oriented design (1980s), or interface/component-based design (1990s). SOA promotes the goal of separating users (consumers) from the service implementations. Services can therefore be run on various distributed platforms and be accessed across networks. This can also maximize reuse of services.

A service comprises a stand-alone unit of functionality available only via a formally defined interface. Services can be some kind of "nano-enterprises" that are easy to produce and improve. Also services can be "mega-corporations" constructed as the coordinated work of subordinate services.

A mature rollout of SOA effectively defines the API of an organization.

Reasons for treating the implementation of services as separate projects from larger projects include:

1. Separation promotes the concept to the business that services can be delivered quickly and independently from the larger and slower-moving projects common in the organization. The business starts understanding systems and simplified user interfaces calling on services. This advocates agility. That is to say, it fosters business innovations and speeds up time-to-market.[28]

2. Separation promotes the decoupling of services from consuming projects. This encourages good design insofar as the service is designed without knowing who its consumers are.

3. Documentation and test artifacts of the service are not embedded within the detail of the larger project. This is important when the service needs to be reused later.

An indirect benefit of SOA involves dramatically simplified testing. Services are autonomous, stateless, with fully documented interfaces, and separate from the cross-cutting concerns of the implementation.

If an organization possesses appropriately defined test data, then a corresponding stub is built that reacts to the test data when a service is being built. A full set of regression tests, scripts, data, and responses is also captured for the service. The service can be tested as a 'black box' using existing stubs corresponding to the services it calls. Test environments can be constructed where the primitive and out-of-scope services are stubs, while the remainder of the mesh is test deployments of full services. As each interface is fully documented with its own full set of regression test documentation, it becomes simple to identify problems in test services. Testing evolves to merely validate that the test service operates according to its documentation, and finds gaps in documentation and test cases of all services within the environment. Managing the data state of idempotent services is the only complexity.

Examples may prove useful to aid in documenting a service to the level where it becomes useful. The documentation of some APIs within the Java Community Process provide good examples. As these are exhaustive, staff would typically use only important subsets. The 'ossjsa.pdf' file within JSR-89 exemplifies such a file.[29]

13.10 Challenges

One obvious and common challenge faced involves managing *services metadata*. SOA-based environments can include many services that exchange messages to perform tasks. Depending on the design, a single application may generate millions of messages. Managing and providing information on how services interact can become complex. This becomes even more complicated when these services are delivered by different organizations within the company or even different companies (partners, suppliers, etc.). This creates huge trust issues across teams; hence SOA Governance comes into the picture.

Another challenge involves the *lack of testing* in SOA space. There are no sophisticated tools that provide testability of all headless services (including message and database services along with web services) in a typical architecture. Lack of horizontal trust requires that both producers and consumers test services on a continuous basis. SOA's main goal is to deliver *agility to businesses*. Therefore it is important to invest in a testing framework (build it or buy it) that would provide the visibility required to find the culprit in the architecture. Business agility requires SOA services to be controlled by the business goals and directives as defined in the business Motivation Model (BMM).[30]

Another challenge relates to providing *appropriate levels of security*. Security models built into an application may no longer suffice when an application exposes its capabilities as services that can be used by other applications. That is, application-managed security is not the right model for securing services. A number of new technologies and standards have started to emerge and provide more appropriate models for security in SOA.

Finally, the impact of changing a service that touches multiple business domains will require a *higher level of change management governance*

As SOA and the WS-* specifications practitioners expand, update and refine their output, they encounter a *shortage of skilled people* to work on SOA-based systems, including the integration of services and construction of services infrastructure.

Interoperability becomes an important aspect of SOA implementations. The WS-I organization has developed basic profile (BP) and basic security profile (BSP) to enforce compatibility.[31] WS-I has designed testing tools to help assess whether web services conform to WS-I profile guidelines. Additionally, another charter has been established to work on the Reliable Secure Profile.

Significant *vendor hype* surrounds SOA, which can create exaggerated expectations. Product stacks continue to evolve as early adopters test the development and runtime products with real-world problems. SOA does not guarantee reduced IT costs, improved systems agility or shorter time to market. Successful SOA implementations may realize some or all of these benefits depending on the quality and relevance of the system architecture and design.[32][33]

Internal IT delivery organizations routinely initiate SOA efforts, and some do a poor job of introducing SOA concepts to a business with the result that SOA remains misunderstood within that business. The adoption of SOA starts to meet IT delivery needs instead of those of the business, resulting in an organization with, for example, superlative laptop provisioning services, instead of one that can quickly respond to market opportunities. Business leadership also frequently becomes convinced that the organization is executing well on SOA.

One of the most important benefits of SOA is its ease of reuse. Therefore accountability and funding models must ultimately evolve within the organization. A business unit needs to be encouraged to create services that other units will use. Conversely, units must be encouraged to reuse services. This requires a few new governance components:

- Each business unit creating services must have an appropriate support structure in place to deliver on its service-level obligations, and to support enhancing existing services strictly for the benefit of others. This is typically quite foreign to business leaders.

- Each business unit consuming services accepts the apparent risk of reusing services outside their own control, with the attendant external project dependencies, etc.

- An innovative funding model is needed as incentive to drive these behaviors above. Business units normally pay the IT organization to assist during projects and then to operate the environment. Corporate incentives should discount these costs to service providers and create internal revenue streams from consuming business units to the service provider.[34] These streams should be less than the costs of a consumer simply building it the old-fashioned way. This is where SOA deployments can benefit from the SaaS monetization architecture.[35]

13.11 Criticisms

Some criticisms of SOA depend on conflating SOA with Web services.[36] For example, some critics claim SOA results in the addition of XML layers, introducing XML parsing and composition.[37] In the absence of native or binary forms of remote procedure call (RPC), applications could run more slowly and require more processing power, increasing costs. Most implementations do incur these overheads, but SOA can be implemented using technologies (for example, Java Business Integration (JBI), Windows Communication Foundation (WCF) and data distribution service (DDS)) that do not depend on remote procedure calls or translation through XML. At the same time, emerging open-source XML parsing technologies (such as VTD-XML) and various XML-compatible binary formats promise to significantly improve SOA performance.[38][39][40]

Stateful services require both the consumer and the provider to share the same consumer-specific context, which is either included in or referenced by messages exchanged between the provider and the consumer. This constraint has the drawback that it could reduce the overall scalability of the service provider if the service-provider needs to retain the shared context for each consumer. It also increases the coupling between a service provider and a consumer and makes switching service providers more difficult.[41] Ultimately, some critics feel that SOA services are still too constrained by applications they represent.[42]

Another concern relates to the ongoing evolution of WS-* standards and products (e. g., transaction, security), and SOA can thus introduce new risks unless properly managed and estimated with additional budget and contingency for additional proof-of-concept work. There has even been an attempt to parody the complexity and sometimes-oversold benefits of SOA, in the form of a 'SOA Facts' site that mimics the 'Chuck Norris Facts' meme.

Some critics regard SOA as merely an obvious evolution of currently well-deployed architectures (open interfaces, etc.).

IT system designs sometimes overlook the desirability of modifying systems readily. Many systems, including SOA-based systems, hard-code the operations, goods and services of the organization, thus restricting their online service and business agility in the global marketplace.

The next step in the design process covers the definition of a service delivery platform (SDP) and its implementation. In the SDP design phase one defines the business information models, identity management, products, content, devices, and the end-user service characteristics, as well as how agile the system is so that it can deal with the evolution of the business and its customers.

13.12 SOA Manifesto

In October 2009, at the *2nd International SOA Symposium*, a mixed group of 17 independent SOA practitioners and vendors, the “SOA Manifesto Working Group”, announced the publication of the SOA Manifesto.[43] The SOA Manifesto is a set of objectives and guiding principles that aim to provide a clear understanding and vision of SOA and service-orientation. Its purpose is rescuing the SOA concept from an excessive use of the term by the vendor community and “a seemingly endless proliferation of misinformation and confusion”.

The manifesto provides a broad definition of SOA, the values it represents for the signatories and some guiding principles. The manifesto prioritizes:

- Business value over technical strategy
- Strategic goals over project-specific benefits
- Intrinsic interoperability over custom integration
- Shared services over specific-purpose implementations
- Flexibility over optimization
- Evolutionary refinement over pursuit of initial perfection

As of September 2010, the SOA Manifesto had been signed by more than 700 signatories and had been translated to nine languages.

13.13 Extensions

13.13.1 SOA, Web 2.0, services over the messenger, and mashups

Web 2.0, a perceived “second generation” of web activity, primarily features the ability of visitors to contribute information for collaboration and sharing. Web 2.0 applications often use RESTful web APIs and commonly feature AJAX based user interfaces, utilizing web syndication, blogs, and wikis. While there are no set standards for Web 2.0, it is characterized by building on the existing Web server architecture and using services. Web 2.0 can therefore be regarded as displaying some SOA characteristics.[44][45][46]

Some commentators also regard mashups as Web 2.0 applications. The term "business mashups" describes web applications that combine content from more than one source into an integrated user experience that shares many of the characteristics of service-oriented business applications (SOBAs). SOBAs are applications composed of services in a declarative manner. There is ongoing debate about “the collision of Web 2.0, mashups, and SOA," with some stating that Web 2.0 applications are a realization of SOA composite and business applications.[47]

13.13.2 Web 2.0

Tim O'Reilly coined the term "Web 2.0" to describe a perceived, quickly growing set of web-based applications.[48] A topic that has experienced extensive coverage involves the relationship between Web 2.0 and Service-Oriented Architectures (SOAs).

SOA is the philosophy of encapsulating application logic in services with a uniformly defined interface and making these publicly available via discovery mechanisms. The notion of complexity-hiding and reuse, but also the concept of loosely coupling services has inspired researchers to elaborate on similarities between the two philosophies, SOA and Web 2.0, and their respective applications. Some argue Web 2.0 and SOA have significantly different elements and thus can not be regarded "parallel philosophies", whereas others consider the two concepts as complementary and regard Web 2.0 as the global SOA.[45]

The philosophies of Web 2.0 and SOA serve different user needs and thus expose differences with respect to the design and also the technologies used in real-world applications. However, as of 2008, use-cases demonstrated the potential of combining technologies and principles of both Web 2.0 and SOA.[45]

In an "Internet of Services", all people, machines, and goods will have access via the network infrastructure of tomorrow. The Internet will thus offer services for all areas of life and business, such as virtual insurance, online banking and music, and so on. Those services will require a complex services infrastructure including service-delivery platforms bringing together demand and supply. Building blocks for the Internet of Services include SOA, Web 2.0 and semantics on the technology side; as well as novel business models, and approaches to systematic and community-based innovation.[49]

Even though Oracle indicates that Gartner is coining a new term, Gartner analysts indicate that they call this *advanced SOA* and refer to it as "SOA 2.0".[50] Most of the major middleware vendors (e. g., Red Hat, webMethods, TIBCO Software, IBM, Sun Microsystems, and Oracle) have had some form of SOA 2.0 attributes for years.

13.13.3 Digital nervous system

SOA implementations have been described as representing a piece of the larger vision known as the digital nervous system[51][52] or the Zero Latency Enterprise.[53]

13.14 See also

- Architecture of Interoperable Information Systems
- Autonomous decentralized system
- Business-agile enterprise
- Business-driven development
- Business Intelligence 2.0
- Business-oriented architecture
- Cloud computing
- Communications-enabled application
- Comparison of business integration software
- Component business model
- Enterprise Mashup Markup Language (EMML)
- Enterprise messaging system
- Enterprise service bus
- Event-driven programming
- HATEOAS (Hypermedia as the Engine of Application State)
- iLAND project
- Library Oriented Architecture
- Message-oriented middleware
- Microservices
- Open ESB
- Platform as a service
- Resource-oriented architecture
- Robot as Service
- Search-oriented architecture
- Semantic service-oriented architecture
- Service layer
- Service-oriented modeling
- Service-oriented architecture implementation framework
- Service (systems architecture)
- Service virtualization
- SOA governance
- SOALIB
- SORCER
- Web-oriented architecture

13.15 References

[1] Chapter 1: Service Oriented Architecture (SOA). Msdn.microsoft.com. Retrieved on 2014-05-30.

[2] "What Is SOA?". opengroup. Retrieved 2013-08-19.

[3] Velte, Anthony T. (2010). *Cloud Computing: A Practical Approach*. McGraw Hill. ISBN 978-0-07-162694-1.

[4] SOA Reference Model definition

[5] "Service Oriented Architecture : What Is SOA?". opengroup.

[6] Channabasavaiah, Holley and Tuggle, Migrating to a service-oriented architecture, *IBM DeveloperWorks*, 16 December 2003.

[7] "SOA Reference Architecture Technical Standard : Basic Concepts". opengroup. Retrieved 2014-10-10.

[8] Bell, Michael (2008). "Introduction to Service-Oriented Modeling". *Service-Oriented Modeling: Service Analysis, Design, and Architecture*. Wiley & Sons. p. 3. ISBN 978-0-470-14111-3.

[9] Bell_, Michael (2010). *SOA Modeling Patterns for Service-Oriented Discovery and Analysis*. Wiley & Sons. p. 390. ISBN 978-0-470-48197-4.

[10] Erl, Thomas. *About the Principles*. Serviceorientation.org, 2005–06

[11] "Application Platform Strategies Blog: SOA is Dead; Long Live Services". Apsblog.burtongroup.com. 2009-01-05. Retrieved 2012-08-13.

[12] Yvonne Balzer Improve your SOA project plans, *IBM*, 16 July 2004

[13] Microsoft Windows Communication Foundation team (2012). "Principles of Service Oriented Design". *msdn.microsoft.com*. Retrieved September 3, 2012.

[14] Principles by Thomas Erl of SOA Systems Inc. eight specific service-orientation principles

[15] M. Hadi Valipour, Bavar AmirZafari, Kh. Niki Maleki, Negin Daneshpour, A Brief Survey of Software Architecture Concepts and Service Oriented Architecture, in Proceedings of 2nd IEEE International Conference on Computer Science and Information Technology, ICCSIT'09, pp 34-38, Aug 2009, China.

[16] Services Oriented Architecture (SOA) - Jargon Buster. Lansa.com. Retrieved on 2014-05-30.

[17] Tony Shan, "Building a Service-Oriented eBanking Platform", scc, pp.237-244, First IEEE International Conference on Services Computing (SCC'04), 2004

[18] "Service Candidate". ServiceOrientation.com. Retrieved 17 October 2014.

[19] E. Oliveros et al. (2012), *Web Service Specifications Relevant for Service Oriented Infrastructures*, Achieving Real-Time in Distributed Computing: From Grids to Clouds, IGI Global, pp. 174–198, doi:10.4018/978-1-60960-827-9.ch010

[20] "SOAP Version 1.2 □□□□□□□ (W3C □□)" (in Japanese). W3.org. Retrieved 2012-08-13.

[21] *Enterprise SOA*. Prentice Hall, 2005

[22] Cardoso, Jorge; Sheth, Amit P. (2006). "Foreword". *Semantic Web Services, Processes and Applications*. SEMANTIC WEB AND BEYOND: Computing for Human Experience. Foreword by Frank Leymann. Springer. xxi. ISBN 978-0-387-30239-3. The corresponding architectural style is called "service-oriented architecture": fundamentally, it describes how service consumers and service providers can be decoupled via discovery mechanisms resulting in loosely coupled systems. Implementing a service-oriented architecture means to deal with heterogeneity and interoperability concerns.

[23] Waldo, Jim (2002). "The Source". Sum Microsystems. Retrieved 2013-12-11. |chapter= ignored (help)

[24] "Service selection and workflow mapping for Grids: an approach exploiting quality-of-service information". *Concurrency and Computation: Practice and Experience* (Wiley) **21** (6): 739–766. 22 July 2008. doi:10.1002/cpe.1343.

[25] Christopher Koch A New Blueprint For The Enterprise, *CIO Magazine*, March 1, 2005

[26] Elizabeth Millard. "Building a Better Process". *Computer User*. January 2005. Page 20.

[27] Bieberstein et al., Service-Oriented Architecture (SOA) Compass: Business Value, Planning, and Enterprise Roadmap (The developerWorks Series) (Hardcover), IBM Press books, 2005, 978-0131870024

[28] Brayan Zimmerli Business Benefits of SOA, *University of Applied Science of Northwestern Switzerland, School of Business*, 11 November 2009

[29] https://cds.sun.com/is-bin/INTERSHOP.enfinity/WFS/CDS-CDS_Developer-Site/en_US/-/USD/ViewProductDetail-Start?ProductRef=7854-oss_service_activation-1.0-fr-spec-oth-JSpec@CDS-CDS_Developer

[30] "From The Business Motivation Model (BMM) To Service Oriented Architecture (SOA)". Jot.fm. Retrieved 2013-06-15.

[31] WS-I Basic Profile

[32] Is There Real Business Value Behind the Hype of SOA?, *Computerworld*, June 19, 2006.

[33] See also: WS-MetadataExchange OWL-S

[34] "4CaaSt marketplace: An advanced business environment for trading cloud services". *Future Generation Computer Systems* (Elsevier) **41**: 104–120. 2014. doi:10.1016/j.future.2014.02.020.

[35] The Overlapping Worlds of SaaS and SOA

[36] McKendrick, Joe. "Bray: SOA too complex; 'just vendor BS'". ZDNet.

[37] M. Riad, Alaa; E. Hassan, Ahmed; F. Hassan, Qusay (2009). "Investigating Performance of XML Web Services in Real-Time Business Systems". *Journal of Computer Science & Systems Biology* **02** (05): 266–271. doi:10.4172/jcsb.1000041.

[38] Index XML documents with VTD-XML

[39] The Performance Woe of Binary XML

[40] Manipulate XML Content the Ximple Way

[41] "The Reason SOA Isn't Delivering Sustainable Software". jpmorgenthal.com. 2009-06-19. Retrieved 2009-06-27.

[42] "SOA services still too constrained by applications they represent". zdnet.com. 2009-06-27. Retrieved 2009-06-27.

[43] SOA Manifesto Official Website Date Accessed: 02 October 2010.

[44] Dion Hinchcliffe Is Web 2.0 The Global SOA?, *SOA Web Services Journal*, 28 October 2005

[45] Schroth, Christoph ; Janner, Till; (2007). "Web 2.0 and SOA: Converging Concepts Enabling the Internet of Services". IT Professional 9 (2007), Nr. 3, pp. 36-41, IEEE Computer Society. Retrieved 2008-02-23.

[46] Hoyer, Volker ; Stanoesvka-Slabeva, Katarina; Janner, Till; Schroth, Christoph; (2008). "Enterprise Mashups: Design Principles towards the Long Tail of User Need". Proceedings of the 2008 IEEE International Conference on Services Computing (SCC 2008). Retrieved 2008-07-08.

[47] Jason Bloomberg Mashups and SOBAs: Which is the Tail and Which is the Dog?, *Zapthink*

[48] "What Is Web 2.0". Tim O'Reilly. 2005-09-30. Retrieved 2008-06-10.

[49] Ruggaber, Rainer; (2007). "Internet of Services—A SAP Research Vision". IEEE Computer Society. Retrieved 2008-02-23.

[50] Yefim Natis & Roy Schulte Advanced SOA for Advanced Enterprise Projects, *Gartner*, July 13, 2006

[51] "From Web to Boarding Area: Delta's SOA is Ready". Retrieved 2009-05-02.

[52] "The Value of An Enterprise Architecture". Retrieved 2009-05-02.

[53] "Moving Toward the Zero Latency Enterprise". Retrieved 2009-05-02.

13.16 External links

- A comparison of SOA standards carried out for Ministry of Defence (United Kingdom) in 2010
- SOA in the real world - Microsoft Developer network
- SOA reference architecture from IBM
- SOA Practitioners Guide Part 2: SOA Reference Architecture
- SOA Practitioners Guide Part 3: Introduction to Services Lifecycle

Chapter 14

Massively multiplayer online game

A **massively multiplayer online game** (also called **MMO** and **MMOG**) is a multiplayer video game which is capable of supporting large numbers of players simultaneously. By necessity, they are played on the Internet.[1] MMOs usually have at least one persistent world, however some games differ. These games can be found for most network-capable platforms, including the personal computer, video game console, or smartphones and other mobile devices.

MMOGs can enable players to cooperate and compete with each other on a large scale, and sometimes to interact meaningfully with people around the world. They include a variety of gameplay types, representing many video game genres.

14.1 History

Main article: History of massively multiplayer online games

The most popular type of MMOG, and the subgenre that pioneered the category,which was launched in late April 1999, is the massively multiplayer online role-playing game (MMORPG), which descended from university mainframe computer MUD and adventure games such as *Rogue* and *Dungeon* on the PDP-10. These games predate the commercial gaming industry and the Internet, but still featured persistent worlds and other elements of MMOGs still used today.

The first graphical MMOG, and a major milestone in the creation of the genre, was the multiplayer flight combat simulation game *Air Warrior* by Kesmai on the GEnie online service, which first appeared in 1986. Kesmai later added 3D graphics to the game, making it the first 3D MMO.

Commercial MMORPGs gained acceptance in the late 1980s and early 1990s. The genre was pioneered by the GemStone series on GEnie, also created by Kesmai, and *Neverwinter Nights*, the first such game to include graphics, which debuted on AOL in 1991.[2]

As video game developers applied MMOG ideas to other computer and video game genres, new acronyms started to develop, such as MMORTS. *MMOG* emerged as a generic term to cover this growing class of games.

The debuts of *The Realm Online*, *Meridian 59* (the first 3D MMORPG), *Ultima Online*, *Underlight* and *EverQuest* in the late 1990s popularized the MMORPG genre. The growth in technology meant that where Neverwinter Nights in 1991 had been limited to 50 simultaneous players (a number that grew to 500 by 1995), by the year 2000 a multitude of MMORPGs were each serving thousands of simultaneous players and led the way for games such as World of Warcraft and *EVE Online*.

Despite the genre's focus on multiplayer gaming, AI-controlled characters are still common. NPCs and mobs who give out quests or serve as opponents are typical in MMORPGs. AI-controlled characters are not as common in action-based MMOGs.

The popularity of MMOGs was mostly restricted to the computer game market until the sixth-generation consoles, with the launch of *Phantasy Star Online* on *Dreamcast* and the emergence and growth of online service Xbox Live. There have been a number of console MMOGs, including *EverQuest Online Adventures* (PlayStation 2), and the multiconsole *Final Fantasy XI*. On PCs, the MMOG market has always been dominated by successful fantasy MMORPGs.

MMOGs have only recently begun to break into the mobile phone market. The first, Samurai Romanesque set in feudal Japan, was released in 2001 on NTT DoCoMo's iMode network in Japan.[3] More recent developments are CipSoft's TibiaME and Biting Bit's *MicroMonster* which features online and bluetooth multiplayer gaming. SmartCell Technology is in development of Shadow of Legend, which will allow gamers to continue their game on their mobile device when away from their PC.

Science fiction has also been a popular theme, featuring games such as *Mankind*, *Anarchy Online*, *Eve Online*, *Star Wars Galaxies* and *The Matrix Online*.

MMOGs emerged from the hard-core gamer community to the mainstream strongly in December 2003 with an analysis in the *Financial Times* measuring the value of the virtual property in the then-largest MMOG, Everquest, to result in a per-capita GDP of 2,266 dollars which would have placed the virtual world of Everquest as the 77th wealthiest nation, on par with Croatia, Ecuador, Tunisia or Vietnam.

Happy Farm is the most popular MMOG with 228 million active users, and 23 million daily users (daily active users logging onto the game with a 24-hour period).[4]

World of Warcraft is a dominant MMOG in the world with more than 50% of the subscribing player base,[5] and with 8-9 million monthly subscribers worldwide. The subscriber base dropped by 1 million after the expansion Wrath of the Lich King, bringing it to 9 million subscribers,[6] though it remains the most popular Western title among MMOGs. In 2008, Western consumer spending on *World of Warcraft* represented a 58% share of the subscription MMOG market.[7] The title has generated over $2.2 billion in cumulative consumer spending on subscriptions since 2005.[7]

14.2 Virtual economies

Main article: Virtual economy

Within a majority of the MMOGs created, there is virtual currency where the player can earn and accumulate money. The uses for such virtual currency are numerous and vary from game to game. The virtual economies created within MMOGs often blur the lines between real and virtual worlds. The result is often seen as an unwanted interaction between the real and virtual economies by the players and the provider of the virtual world. This practice (economy interaction) is mostly seen in this genre of games. The two seem to come hand in hand with even the earliest MMOGs such as *Ultima Online* having this kind of trade, real money for virtual things.

The importance of having a working virtual economy within an MMOG is increasing as they develop. A sign of this is CCP Games hiring the first real-life economist for its MMOG Eve Online to assist and analyze the virtual economy and production within this game.

The results of this interaction between the virtual economy, and our real economy, which is really the interaction between the company that created the game and the third-party companies that want to share in the profits and success of the game. This battle between companies is defended on both sides. The company originating the game and the intellectual property argue that this is in violation of the terms and agreements of the game as well as copyright violation since they own the rights to how the online currency is distributed and through what channels. The case that the third-party companies and their customers defend, is that they are selling and exchanging the time and effort put into the acquisition of the currency, not the digital information itself. They also express that the nature of many MMOGs is that they require time commitments not available to everyone. As a result, without external acquisition of virtual currency, some players are severely limited to being able to experience certain aspects of the game.

The practice of acquiring large volumes of virtual currency for the purpose of selling to other individuals for tangible and real currency is called gold farming. Many players who have poured in all of their personal effort resent that there is this exchange between real and virtual economies since it devalues their own efforts. As a result, the term 'gold farmer' now has a very negative connotation within the games and their communities. This slander has unfortunately also extended itself to racial profiling and to in-game and forum insulting.

The reaction from many of the game companies varies. In games that are substantially less popular and have a small player base, the enforcement of the elimination of 'gold farming' appears less often. Companies in this situation most likely are concerned with their personal sales and subscription revenue over the development of their virtual economy, as they most likely have a higher priority to the games viability via adequate funding. Games with an enormous player base, and consequently much higher sales and subscription income, can take more drastic actions more often and in much larger volumes. This account banning could also serve as an economic gain for these large games, since it is highly likely that, due to demand, these 'gold farming' accounts will be recreated with freshly bought copies of the game. In December 2007, Jagex Ltd., in a successful effort to reduce real world trading levels enough so they could continue using credit cards for subscriptions, introduced highly controversial changes to its MMOG RuneScape to counter the negative effects gold sellers were having on the game on all levels.[8]

The virtual goods revenue from online games and social networking exceeded US$7 billion in 2010.[9]

In 2011, it was estimated that up to 100,000 people in China and Vietnam are playing online games to gather gold and other items for sale to Western players.[10]

However single player in MMOs is quite viable, especially in what is called 'player vs environment' gameplay. This may result in the player being unable to experience all content, as many of the most significant and potentially rewarding game experiences are events which require large and coordinated teams to complete.

Most MMOGs also share other characteristics that make them different from other multiplayer online games. MMOGs host a large number of players in a single game world, and all of those players can interact with each other at any given time. Popular MMOGs might have thousands of players online at any given time, usually on a company owned servers. Non-MMOGs, such as *Battlefield 1942* or *Half-Life* usually have fewer than 50 players online (per server) and are usually played on private servers. Also, MMOGs usually do not have any significant mods since the game must work on company servers. There is some debate if a high head-count is the requirement to be an MMOG. Some say that it is the size of the game world and its capability to support a large number of players that should matter. For example, despite technology and content constraints, most MMOGs can fit up to a few thousand players on a single game server at a time.

To support all those players, MMOGs need large-scale game worlds, and servers to connect players to those worlds. Some games have all of their servers connected so all players are connected in a shared universe. Others have copies of their starting game world put on different servers, called “shards”, for a sharded universe. Shards got their name from Ultima Online, where in the story, the shards of Mondain’s gem created the duplicate worlds.

Still others will only use one part of the universe at any time. For example, *Tribes* (which is not an MMOG) comes with a number of large maps, which are played in rotation (one at a time). In contrast, the similar title *PlanetSide* allows all map-like areas of the game to be reached via flying, driving, or teleporting.

MMORPGs usually have sharded universes, as they provide the most flexible solution to the server load problem, but not always. For example, the space simulation *Eve Online* uses only one large cluster server peaking at over 60,000 simultaneous players.

There are also a few more common differences between MMOGs and other online games. Most MMOGs charge the player a monthly or bimonthly fee to have access to the game’s servers, and therefore to online play. Also, the game state in an MMOG rarely ever resets. This means that a level gained by a player today will still be there tomorrow when the player logs back on. MMOGs often feature in-game support for clans and guilds. The members of a clan or a guild may participate in activities with one another, or show some symbols of membership to the clan or guild.

14.2.1 Technical aspect

It is challenging to develop the database engines that are needed to run a successful MMOG with millions of players.[11] Many developers have created their own, but attempts have been made to create *middleware*, software that would help game developers concentrate on their games more than technical aspects. One such piece of middleware is called BigWorld.

An early, successful entry into the field was VR-1 Entertainment whose Conductor platform was adopted and endorsed by a variety of service providers around the world including Sony Communications Network in Japan; the Bertelsmann Game Channel in Germany; British Telecom’s Wireplay in England; and DACOM and Samsung SDS in South Korea.[12] Games that were powered by the Conductor platform included Fighter Wing, Air Attack, Fighter Ace, EverNight, Hasbro Em@ail Games (Clue, NASCAR and Soccer), Towers of Fallow, The SARAC Project, VR1 Crossroads and Rumble in the Void.

One of the bigger problems with the engines has been to handle the vast number of players. Since a typical server can handle around 10,000–12,000 players, 4000–5000 active simultaneously, dividing the game into several servers has up until now been the solution. This approach has also helped with technical issues, such as lag, that many players experience. Another difficulty, especially relevant to real-time simulation games, is time synchronization across hundreds or thousands of players. Many games rely on time synchronization to drive their physics simulation as well as their scoring and damage detection.

14.3 Game types

There are several types of massively multiplayer online games.

14.3.1 Role-playing

A group photo of a “Linkshell” guild in the roleplaying game Final Fantasy XI.

Main article: Massively multiplayer online role-playing game
See also: List of MMORPGs

Massively multiplayer online role-playing games, known as **MMORPG**s, are the most common type of MMOG. Some MMORPGs are designed as a multiplayer browser game in order to reduce infrastructure costs and utilise a thin client that most users will already have installed. The acronym *BBMMORPG*s has sometimes been used to describe these as *browser-based*.

Bulletin board role-playing games

A large number of games categorize under MMOBBG, massively multiplayer online bulletin board game, can also be called MMOBBRPG. These particular type of games are primarily made up of text and descriptions, although images are often used to enhance the game.

14.3.2 First-person shooter

Main article: Massively multiplayer online first-person shooter game
See also: List of MMOFPSs

MMOFPS is an online gaming genre which features a persistent world and a large number of simultaneous players in a first-person shooter fashion.[13][14] These games provide large-scale, sometimes team-based combat. The addition of persistence in the game world means that these games add elements typically found in RPGs, such as experience points. However, MMOFPS games emphasize player skill more than player statistics, as no number of in-game bonuses will compensate for a player's inability to aim and think tactically.[15]

14.3.3 Real-time strategy

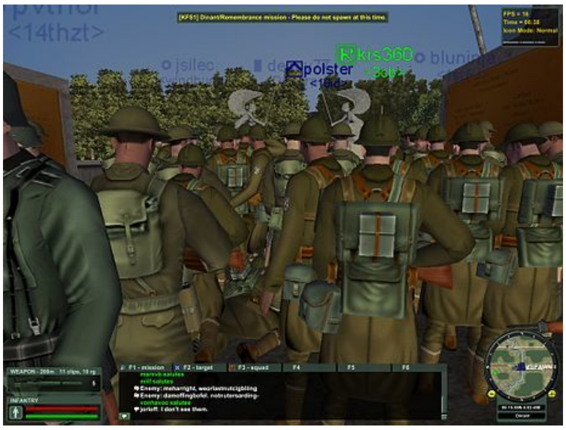
Main article: Massively multiplayer online real-time strategy game
See also: List of MMORTSs

Massively multiplayer online real-time strategy games, also known as "MMORTS", combine real-time strategy (RTS) with a persistent world. Players often assume the role of a general, king, or other type of figurehead leading an army into battle while maintaining the resources needed for such warfare. The titles are often based in a sci-fi or fantasy universe and are distinguished from single or small-scale multiplayer RTSes by the number of players and common use of a persistent world, generally hosted by the game's publisher, which continues to evolve even when the player is offline.

14.3.4 Turn-based strategy

Steve Jackson Games' UltraCorps is an example of a MMO turn-based strategy game.[16] Hundreds of players share the same playing field of conquest. In a "mega" game, each turn fleets are built and launched to expand one's personal empire. Turns are usually time-based, with a "tick" schedule usually daily. All orders are processed, and battles resolved, at the same time during the tick. Similarly, in *Darkwind: War on Wheels*, vehicle driving and combat orders are submitted simultaneously by all players and a "tick" occurs typically once per 30 seconds. This allows each player to accurately control multiple vehicles and pedestrians in racing or combat.

14.3.5 Simulations



World War II Online *simulation game showing the numbers of players during a special event in June 2008. Some 400 people had spawned in for this gathering in this location in the game.*

Some MMOGs have been designed to accurately simulate certain aspects of the real world. They tend to be very specific to industries or activities of very large risk and huge potential loss, such as rocket science, airplanes, trucks, battle tanks, submarines etc. Gradually as simulation technology is getting more mainstream, so too various simulators arrive into more mundane industries.

The initial goal of *World War II Online* was to create a map (in north western Europe) that had real world physics (gravity, air/water resistance, etc.), and ability for players to have some strategic abilities to its basic FPS/RPG role. While the current version is not quite a true simulated world, it is very complex and contains a large persistent world.

The MMOG genre of air traffic simulation is one example, with networks such as VATSIM and IVAO striving to provide rigorously authentic flight-simulation environments to players in both pilot and air traffic controller roles. In this category of MMOGs, the objective is to create duplicates of the real world for people who cannot or do not wish to undertake those experiences in real life. For example, flight simulation via an MMOG requires far less expenditure of time and money, is completely risk-free, and is far less restrictive (fewer regulations to adhere to, no medical exams to pass, and so on).

Another specialist area is mobile telecoms operator (carrier) business where billion-dollar investments in networks are needed but marketshares are won and lost on issues from segmentation to handset subsidies. A specialist simulation was developed by Nokia called Equilibrium/Arbitrage to have over a two-day period five teams of top management of one operator/carrier play a "wargame" against each other, under extremely realistic conditions, with one operator an incumbent fixed and mobile network operator, another a new entrant mobile operator, a third a fixed-line/internet operator etc. Each team is measured by outperforming their rivals by market expectations of that type of player. Thus each player has drastically different goals, but within the simulation, any one team can win. Also to ensure maximum intensity, only one team can win. Telecoms senior executives who have taken the Equilibrium/Arbitrage simulation say it is the most intense, and most useful training they have ever experienced. It is typical of business use of simulators, in very senior management training/retraining.

Other online simulation games include *War Thunder*, *Motor City Online*, *The Sims Online*, and *Jumpgate*.

Sports

A massively multiplayer online sports game is a title where players can compete in some of the more traditional major league sports, such as football (soccer), basketball, baseball, hockey, golf or American football. According to GameSpot.com, Baseball Mogul Online was "the world's first massively multiplayer online sports game".[17] Other titles that qualify as MMOSG have been around since the early 2000s, but only after 2010 did they start to receive the endorsements of some of the official major league associations and players.

Racing

MMOR means **massively multiplayer online racing**. Currently there are only a small number of racing based MMOGs, including *Kart Rider*, *Upshift StrikeRacer*, *Test Drive Unlimited*, *Project Torque*, *Drift City*, *The Crew*, *Race or Die (iPhone)* and *Need for Speed: World*. The *Trackmania* series is the world's largest MMO racing game and holds the world record for "Most Players in a Single Online Race". Although *Darkwind: War on Wheels* is more combat based than racing, it is also considered an MMOR.

14.3.6 Casual

Many types of MMO games can be classified as casual, because they are designed to appeal to all computer users (as opposed to subgroup of frequent game buyers), or to fans of another game genre (such as collectible card games). Such games are easy to learn and require a smaller time commitment than other game types. One example is *Racing Frogs*, an MMOG that can be played with only a small amount of time every day. Other popular casual games include simple management games such as *The Sims Online*, *Monopoly City Streets*, *Virtonomics*, or *Kung Fu Panda World*.

MMOPGs, or massively multiplayer puzzle games, are games based entirely on puzzle elements. It is usually set in a world where the players can access the puzzles around the world. Most games that are MMOPGs are hybrids with other genres. *Castle Infinity* was the first MMOG developed for children. Its gameplay falls somewhere between puzzle and adventure.

There are also massively multiplayer collectible card games: *Alteil*, *Astral Masters* and *Astral Tournament*. Other MMOCCGs might exist (*Neopets* has some CCG elements) but are not as well known.

Alternate reality games (ARGs) can be massively multiplayer, allowing thousands of players worldwide to cooperate in puzzle trails and mystery solving. ARGs take place in a unique mixture of online and real-world play that usually does not involve a persistent world, and are not necessarily multiplayer, making them different from MMOGs.

Music/Rhythm

Massively multiplayer online music/rhythm games (**MMORG**s), sometimes called massively multiplayer online dance games (**MMODG**s), are MMOGs that are also music video games. This idea was influenced by *Dance Dance Revolution*. *Audition Online* is another casual massively multiplayer online game and it is produced by T3 Entertainment.

Just Dance 2014 has a game mode called World Dance Floor, which also structures like a MMORPG.

Social

Massively multiplayer online social games focus on socialization instead of objective-based gameplay. There is a great deal of overlap in terminology with "online communities" and "virtual worlds". One example that has garnered widespread media attention is Linden Lab's *Second Life*, emphasizing socializing, world-building and an in-world virtual economy that depends on the sale and purchase of user-created content. It is technically an MMOSG or Casual Multiplayer Online (CMO) by definition, though its stated goal was to realize the concept of the Metaverse from Neal Stephenson's novel *Snow Crash*. Instead of being based around combat, one could say that it was based around the creation of virtual objects, including models and scripts. In practice, it has more in common with *Club Caribe* than *Everquest*. It was the first MMO of its kind to achieve widespread success (including attention from mainstream media); however, it was not the first (as *Club Caribe* was released in 1988). Competitors in this subgenre (non-combat-based MMORPG) include *Active Worlds*, *There*, *SmallWorlds*, *Furcadia*, *Whirled* and *IMVU*.

Many browser based Casual MMOs have begun to spring up. This has been made easier because of maturing of *Adobe Flash* and the popularity of *Club Penguin*, *Growtopia*, and *The Sims Online*.

14.4 Research

Some recent attempts to build peer-to-peer (P2P) MMOGs have been made. *Outback Online* may be the first commercial one,[18] however, so far most of the efforts have been academic studies.[19] A P2P MMOG may potentially be more scalable and cheaper to build, but notable issues with P2P MMOGs include security and consistency control, which can be difficult to address given that clients are easily hacked. Some MMOGs such as Vindictus use P2P networking and client-server networking together.

In April 2004, the United States Army announced that it was developing a massively multiplayer training simulation called *AWE* (asymmetric warfare environment). The purpose of *AWE* is to train soldiers for urban warfare and there are no plans for a public commercial release. Forterra Systems is developing it for the Army based on the *There* engine.[20]

In 2010, Bonnie Nardi published an ethnographic study on World of Warcraft examined with Lev Vygotsky's activity theory.

As the field of MMOs grows larger each year, research has also begun to investigate the socio-informatic bind the games create for their users. In 2006, researchers Constance A. Steinkuehler and Dmitri Williams initiated research on such topics. The topic most intriguing to the pair was to further understand the gameplay, as well as the virtual world serving as a social meeting place, of popular MMOs.

To further explore the effects of social capital and social relationships on MMOs, Steinkuehler and Williams combined conclusions from two different MMO research projects: sociocultural perspective on culture and cognition, and the other on media effects of MMOs. The conclusions of the two studies explained how MMOs function as a new form of a “third place” for informal social interactions much like coffee shops, pubs, and other typical hangouts. Many scholars, however, such as Oldenburg (1999), refute the idea of a MMOs serving as a “third place” due to inadequate bridging social capital. His argument is challenged by Putnam (2000) who concluded that MMOs are well suited for the formation of bridging social capital, tentative relationships that lack in depth, because it is inclusive and serves as a sociological lubricant that is shown across the data collected in both of the research studies.[21]

MMOs can also move past the “lubricant” stage and into the “superglue” stage known as bonding social capital, a closer relationship that is characterized by stronger connections and emotional support. The study concludes that MMOs function best as a bridging mechanism rather than a bonding one, similar to a “third place”. Therefore, MMOs have the capacity and the ability to serve as a community that effectively socializes users just like a coffee shop or pub, but conveniently in the comfort of their own home.[21]

14.5 Spending

British online gamers are outspending their German and French counterparts according to a recently released study commissioned by Gamesindustry.com and TNS. The UK MMO-market is now worth £195 million in 2009 compared to the £165 million and £145 million spent by German and French online gamers.[22]

The US gamers spend more, however, spending about $3.8 billion overall on MMO games. $1.8 billion of that money is spent on monthly subscription fees. The money spent averages out to $15.10 between both subscription and free-to-play MMO gamers. The study also found that 46% of 46 million players in the US pay real money to play MMO games.[23]

Today’s Gamers MMO Focus Report, published in March 2010, was commissioned by TNS and gamesindustry.com. A similar study for the UK market-only (*UK National Gamers Survey Report*)[24] was released in February 2010 by the same groups.

14.6 See also

- List of massively multiplayer online games
- Multiplayer online game
- Online game
- Social network game
- Virtual world

14.7 References

[1] What is Massively Multiplayer Online Game (MMOG)? - Definition from. Techopedia. Retrieved on 2014-05-25.

[2] “The Oldest MMORPGs in Gaming History”. Retrieved 2015-03-09.

[3] Krikke, J. (2003). “Samurai Romanesque, J2ME, and the battle for mobile cyberspace”. *IEEE Computer Graphics and Applications* **23**: 16–00. doi:10.1109/MCG.2003.1159608.

[4] "□□□□□□□□:□□□□□□"□□□□" - □□□□". Game.dayoo.com. Retrieved 2013-08-28.

[5] Mmogchart.com

[6] "Blizzard Entertainment:Press Releases". Us.blizzard.com. 2010-10-07. Retrieved 2013-08-28.

[7] Hardy-Rolls, Piers (2009). *Subscription MMOGs: Life Beyond World of Warcraft* (PDF). Lyndon, UK: Screen Digest. Retrieved 30 March 2009.

[8] Runescape.com

[9] Kevin Kwang (12 July 2011). "Online games, social networks drive virtual goods". *ZDNet*. Retrieved 27 November 2014.

[10] "Virtual sales provide aid to poorer nations". *BBC*. 8 April 2011. Retrieved 27 November 2014.

[11] Building a simple yet powerful MMO game architecture, Part 1: Introduction. Ibm.com (2008-09-08). Retrieved on 2014-05-25.

[12] "VR-1 Conductor Technology Suite Endorsed by LeadingOnline Service Providers. | Consumer Products > Computing Products from". AllBusiness.com. 28 April 1998. Retrieved 16 March 2009.

[13] "IGN: The Worlds First MMOFPS is nearly complete". Uk.games.ign.com. Retrieved 16 March 2009.

[14] Thierry 'Scooter' Nguyen (20 March 2007). "GameSpy: Huxley Preview". *GameSpy*. Retrieved 16 March 2009.

[15] Leon Ryan (May 2007). *Beyond the Looking Glass of MMOG's*. GameAxis Unwired. pp. 27–31. Retrieved 15 March 2009.

[16] "UltraCorps is recommended to anyone interested in a simplistic, slow-paced sci-fi strategy game."

[17] "Baseball Mogul Online: The World's First MMOSG". gamespot.com. Retrieved 23 July 2001.

[18] "Yoicks! It's another virtual world - BizTech - Technology". theage.com.au. 26 February 2007. Retrieved 16 March 2009.

[19] "VAST - Related Work". Vast.sourceforge.net. 26 November 2008. Retrieved 16 March 2009.

[20] Gonzalez, Lauren (17 January 2008). "Spot On: The US Army's There-based simulation - News at GameSpot". Gamespot.com. Retrieved 16 March 2009.

[21] Steinkuehler, C. A. and Williams, D. (2006), Where Everybody Knows Your (Screen) Name: Online Games as "Third Places". Journal of Computer-Mediated Communication, 11: 885–909. doi: 10.1111/j.1083-6101.2006.00300.x

[22] "MMO Big in Britain But US Still No 1". Agencynews.net. 20 October 2003.

[23] US Gamers spent $3.8 billion on MMO Gaming in 2009. "US Gamers spent $3.8 billion on MMO Gaming in 2009 - MMO Hut". Mmohuts.com. Retrieved 2013-08-28.

[24] "Millions Spent on Casual Gaming". BBC UK. 10.02.2010. Check date values in: |date= (help)

14.8 External links

- Massive Multiplayer Online at DMOZ

14.9 Text and image sources, contributors, and licenses

14.9.1 Text

- **Multiprocessing** *Source:* http://en.wikipedia.org/wiki/Multiprocessing?oldid=650855157 *Contributors:* Szopen, The Anome, Andre Engels, Aldie, Maury Markowitz, Edward, Wapcaplet, Dori, Ahoerstemeier, Ronz, Wernher, Raul654, Chuunen Baka, AlexPlank, Iain.mcclatchie, DavidCary, AlistairMcMillan, Pgan002, Phe, Dyl, CanisRufus, R. S. Shaw, Guy Harris, Derbeth, FlaBot, BMF81, Chobot, Mortenoesterlundjoergensen, Borgx, Hairy Dude, Tony1, Rwwww, SmackBot, Hydrogen Iodide, Henriok, Agateller, Paxswill, Cfallin, Allan McInnes, Nakon, VegaDark, Zac67, Harryboyles, Euchiasmus, Jurohi, Quaeler, Beno1000, Deice, Nczempin, Xaariz, Kubanczyk, James086, Towopedia, Dgies, TexMurphy, JAnDbot, Arch dude, Hexdump, David Eppstein, GermanX, R'n'B, VolkovBot, Umar420e, VasilievVV, Slserpent, Chickyfuzz14, SieBot, Gorptree, ClueBot, Rilak, Excirial, Goodone121, PixelBot, HumphreyW, Asafshelly, Чръный человек, G7a, Addbot, Leikfaz, Download, Jasper Deng, Luckas-bot, Amirobot, Goldenthree, MulticoreNerd, Peter Flass, AnomieBOT, HughesJohn, Xqbot, Julle, Miym, Cyfraw, Glenn Maddox, EmausBot, WikitanvirBot, Ferry24.Milan, Topeil, MaGa, Heyjoy, Cswierkowski, ClueBot NG, Satellizer, بهاء حجازي, Dipankan001, Mrt3366, Tagremover, Someone not using his real name, Sagarsurani and Anonymous: 96

- **Computer multitasking** *Source:* http://en.wikipedia.org/wiki/Computer%20multitasking?oldid=650586294 *Contributors:* Damian Yerrick, Bryan Derksen, The Anome, Ap, Rjstott, Dachshund, Pit, Rp, Theanthrope, Ahoerstemeier, Salsa Shark, Magnus.de, Jessel, Tpbradbury, Wernher, Robbot, Tobias Bergemann, Jyril, Lupin, Peruvianllama, AJim, AlistairMcMillan, Nayuki, Edcolins, Wmahan, Tipiac, Starblue, Stephan Leclercq, Zhuuu, Beland, QuiTeVexat, Simson, Burschik, Ulmanor, Abdull, Mormegil, CALR, Rich Farmbrough, Florian Blaschke, Djce, Dmeranda, Bender235, Lou Crazy, CanisRufus, Drhex, Bobo192, R. S. Shaw, Csabo, Wtshymanski, H2g2bob, Mikenolte, Nuno Tavares, Gimboid13, Palica, Pabix, Guinness2702, Bhadani, GeorgeBills, FlaBot, Chris Purcell, RexNL, YurikBot, Wavelength, DMahalko, Piet Delport, Zimbricchio, CarlHewitt, Dianne Hackborn, Misza13, DeadEyeArrow, PS2pcGAMER, JakkoWesterbeke, Ketsuekigata, Drable, Thomas Blomberg, GrinBot, SmackBot, Reedy, Ixtli, Sparking Spirit, Andy M. Wang, Stevage, Nbarth, Rrelf, Tsca.bot, AntiVan, Zvar, Jsavit, Radagast83, FormerUser1, Esb, Ozhiker, Harryboyles, BlindWanderer, Loadmaster, Dicklyon, EdC, Peyre, Iridescent, Tawkerbot2, FatalError, Unixguy, Ale jrb, Phatom87, Xaariz, Kubanczyk, Wikid77, Marek69, AntiVandalBot, Ad88110, Alphachimpbot, JAnDbot, MER-C, Arch dude, Magioladitis, JNW, Faizhaider, Web-Crawling Stickler, Glen, Manticore, Maurice Carbonaro, Public Menace, Jesant13, Silas S. Brown, Cometstyles, Jcea, VolkovBot, Mazarin07, HiDrNick, Toyalla, SieBot, RucasHost, Theaveng, Yerpo, Ixe013, Nskillen, Frappucino, Anchor Link Bot, Mhouston, ClueBot, Ndenison, M4gnum0n, PixelBot, Estirabot, Sun Creator, Oliverbell99, Fiskegalen92, Andy16666, DumZiBoT, Joel Saks, Kwjbot, Dsimic, Arkantospurple, Deineka, Akshatdabralit, Cst17, CarsracBot, Lightbot, דוד שי, Gail, Zorrobot, John.St, Legobot, Luckas-bot, ArchonMagnus, Peter Flass, AnomieBOT, TwistedText, Materialscientist, GB fan, Xqbot, Julle, Almabot, 399man, RibotBOT, Alan.A.Mick, Rat2, D'ohBot, Mfwitten, Atlantia, EmausBot, WikitanvirBot, Dewritech, Tolly4bolly, L Kensington, ChuispastonBot, 28bot, ClueBot NG, Scicluna93, Toastyking, Mesoderm, ScottSteiner, Theopolisme, Compilation finished successfully, Jimperio, GGShinobi, Venera Seyranyan, ANI MARTIROSYAN, Shaun, BattyBot, FizzixNerd, Wrmattison, Mogism, Jamesx12345, Sriharsh1234, Vcfahrenbruck, OMPIRE, Benjamintf1, 7stone7, AyanMazumdar91 and Anonymous: 192

- **Symmetric multiprocessing** *Source:* http://en.wikipedia.org/wiki/Symmetric%20multiprocessing?oldid=652732259 *Contributors:* The Anome, Maury Markowitz, Edward, Alfio, Ronz, Zoicon5, Maximus Rex, Furrykef, Pakaran, AnthonyQBachler, Robbot, RedWolf, Chris Roy, Rdash, AlistairMcMillan, Jabowery, Neilc, Beland, Vina, Yggdræsil, Deleting Unnecessary Words, Tietew, Bjacob, Qiq, Urhixidur, Twajn, EagleOne, Pavel Vozenilek, Dyl, CanisRufus, Spearhead, Matt Britt, Ricsi, Bad Byte, Homerjay, Cncxbox, Pearle, Poweroid, Guy Harris, 119, Rcbarnes, Hns, Wikicaz, Inarius, Nuno Tavares, SKopp, Fred J, Mr Bill, Smithfarm, Hathawayc, FlaBot, Margosbot, Pathoschild, Ewlyahoocom, BMF81, Butros, Anrie Nord, RobotE, Splash, Gardar Rurak, Hymyly, Aaron Brenneman, Qviri, Amwebb, Richard Russell, Cffrost, JLaTondre, Rwwww, Plouiche, Jsnx, SmackBot, Henriok, Yuyudevil, Brianski, Chris the speller, Artoftransformation, Bluebot, Ctrlfreak13, Stephanvaningen, Thumperward, BBCWatcher, Kraljja, Tsca.bot, Solarix, JonHarder, Easwarno1, Soosed, Wybot, Vertigo Acid, Autopilot, Harryboyles, Aqualize, Joffeloff, Bjankuloski06en, Jec, DagErlingSmørgrav, Quaeler, CRGreathouse, Raysonho, Mr Echo, Evilgohan2, Myasuda, Cydebot, Dr unix, GSTQ21C, Kubanczyk, Fatfool, Jayrvox, Hcobb, NocNokNeo, Dgies, Stannered, JAnDbot, Arch dude, Jtrob704, Magioladitis, Realg187, Iam8up, Allstarecho, Vinograd19, R'n'B, Szeder, Daniel.robert.franklin, Khazadum, Alainkaa, ParallelWolverine, AlleborgoBot, EwokiWiki, Gerakibot, Edans.sandes, Jerryobject, MarkMLl, Kntkadosh, Garysixpack, Brian Geppert, Martarius, Rilak, JWA74, Elizium23, Wwskier, DumZiBoT, JRWoodwardMSW, Jabberwoch, Addbot, Ghettoblaster, Tothwolf, John Sauter, Тивирополник, Victor.gattegno, Yobot, SabbaZ, Goldenthree, AnomieBOT, Rubinbot, LilHelpa, Xqbot, DataWraith, DSisyphBot, LAMurakami, RibotBOT, W Nowicki, Xcvista, Rocketshiporion, Jesse V., Slon02, WikitanvirBot, Ecapoccia, Ferry24.Milan, Ipsign, Kalpase, Eemars, Rezabot, MerlIwBot, Wbm1058, BattyBot, Harnisch 54667, Npx122sy, Raulizahi and Anonymous: 181

- **Asymmetric multiprocessing** *Source:* http://en.wikipedia.org/wiki/Asymmetric%20multiprocessing?oldid=601204820 *Contributors:* SimonP, Delirium, Joy, Giftlite, EagleOne, Sdwood, Sietse Snel, Giraffedata, Guy Harris, Atlant, Firsfron, Woohookitty, Tabletop, Rjwilmsi, Mbutts, Martin Hinks, YurikBot, SmackBot, Arny, Chris the speller, JennyRad, Nixeagle, Thegathering, CapitalR, CRGreathouse, Arrenlex, Cydebot, Xaariz, Thijs!bot, Widefox, RainbowCrane, Sir Link, Cocytus, R'n'B, Player 03, DancesWithPeons, Milan Keršláger, Azimout, MarkMLl, Huku-chan, Rilak, Чръный человек, Skarebo, TheMoMonez, G7a, Addbot, Mephiston999, John Sauter, Yobot, AnomieBOT, Pontificalibus, Chatul, Skyerise, Pierre5018, OnePt618, Tech.kyle, MerlIwBot, Helpful Pixie Bot, بهاء حجازي, Bvidulich and Anonymous: 43

- **Non-uniform memory access** *Source:* http://en.wikipedia.org/wiki/Non-uniform%20memory%20access?oldid=641981689 *Contributors:* AxelBoldt, The Anome, Maury Markowitz, Hfastedge, (, Mulad, Emperorbma, Jharrell, Pedant17, RedWolf, Iain.mcclatchie, Rchandra, Isidore, Beland, MJA, Togo, Bjacob, Hellisp, NegativeK, Plugwash, CanisRufus, Pion, Suruena, SteinbDJ, Kinema, Kglavin, RadioActive, BenjaminHare, Margosbot, Ewlyahoocom, Jagjag, YurikBot, RussBot, Gardar Rurak, Moop2000, Długosz, Ravedave, Lkcl, Entheta, SmackBot, Gilliam, Thumperward, Letdorf, BBCWatcher, JonHarder, Fuhghettaboutit, Luís Felipe Braga, Cdills, Pawlex, Benjy613, Raysonho, CCFreak2K, Cydebot, Dr unix, Thijs!bot, Xycaleth, Arch dude, PhilKnight, .anacondabot, Raanoo, KenSharp, Bigfoot00, Thad McCotter, Ken g6, Sander van Malssen, GL1zdA, Khazadum, Pleeplee, BotMultichill, VVVBot, Jerryobject, Lightmouse, COBot, Trustable, Tanvir Ahmmed, Rilak, Arjayay, Dsimic, Addbot, Ramu50, Luckas-bot, Yobot, Vincnet, Amirobot, Zadneram, Winterst, Jonesey95, MastiBot, FoxBot, DixonDBot, EmausBot, WikitanvirBot, Wikipelli, Jan Traspe, Zalnas, Ferry24.Milan, Etorus, KentOlsen, SecureDBA, ChrisGualtieri, Electricmuffin11, Becker553, JailaBailaBoo, Psantimauro, K0zka and Anonymous: 73

- **Multi-core processor** *Source:* http://en.wikipedia.org/wiki/Multi-core%20processor?oldid=654003153 *Contributors:* Edward, Mahjongg, Nixdorf, Ixfd64, 7265, CesarB, Ronz, Julesd, Charles Matthews, Dragons flight, Furrykef, Bevo, Mazin07, Jakohn, Donreed, Altenmann, Nurg, Auric, Bkell, Ancheta Wis, Centrx, Giftlite, DavidCary, Gracefool, Solipsist, Falcon Kirtaran, Kiteinthewind, Ludootje, Cynical, Qiq, Ukexpat, GreenReaper, Alkivar, Real NC, MattKingston, Monkeyman, Reinthal, Archer3, Rich Farmbrough, Florian Blaschke, Sapox, SECProto, Berkut, Dyl, Bender235, Narcisse, RoyBoy, Dennis Brown, Neilrieck, WhiteTimberwolf, Bobo192, Fir0002, SnowRaptor, Matt Britt, Hectoruk, Gary, Liao, Polarscribe, Guy Harris, Hoary, Evil Prince, Lerdsuwa, Bsadowski1, Gene Nygaard, Marasmusine, Simetrical, Woohookitty, Henrik, Mindmatrix, Aaron McDaid, Splintax, Pol098, Vossanova, Qwertyus, Haikupoet, JIP, Ketiltrout, SMC, Smithfarm, CQJ, Bubba73, Yamamoto Ichiro, Skizatch, Ian Pitchford, Master Thief Garrett, Crazycomputers, Superchad, Dbader, DaGizza, SirGrant, Hairy Dude, TheDoober, Epolk, Stephenb, Rsrikanth05, NawlinWiki, VetteDude, Thiseye, Rbarreira, Anetode, DGJM, Falcon9x5, Addps4cat, Closedmouth, Fram, Andyluciano, JLaTondre, Carlosguitar, Mark hermeling, SmackBot, Mmernex, Stux, Henriok, JPH-FM, Jagged 85, Powo, Pinpoint23, Thumperward, Swanner, Hibernian, JagSeal, E946, Shalom Yechiel, Frap, KaiserbBot, AcidPenguin9873, JonHarder, Kcordina, Aldaron, Fuhghettaboutit, Letowskie, DWM, Natamas, Kellyprice, Fitzhugh, Sonic Hog, A5b, Homo sapiens, Lambiam, Kyle wood, JzG, Pgk1, Littleman TAMU, Ulner, WhartoX, Disavian, Danorux, Soumyasch, Joffeloff, Gorgalore, Guy2007, Fernando S. Aldado, 16@r, JHunterJ, NJA, Peyre, Vincecate, Hu12, Quaeler, Iridescent, Pvsuresh, Tawkerbot2, CmdrObot, Plasticboob, CBM, Nczempin, Jesse Viviano, Michaelbarreto, Shandris, Evilgohan2, Neelix, Babylonfive, ScorpSt, Cydebot, Myscrnnm, Steinj, Drtechmaster, Kozuch, Thijs!bot, Hervegirod, Mwastrod, Bahnpirat, Squater, Dawnseeker2000, Sherbrooke, AlefZet, AntiVandalBot, Widefox, Seaphoto, Shalewagner, DarthShrine, Chaitanya.lala, Leuko, Od1n, Gamer2325, Arch dude, IanOsgood, Stylemaster, Aviadbd, Geniac, Coffee2theorems, Ramurf, Vintei, JamesBWatson, CattleGirl, CountingPine, Midgrid, EagleFan, David Eppstein, DerHexer, Gimpy530, Gwern, Gjd001, Oren0, Bdsatish, Red66, Ehoogerhuis, Sigmajove, Felipe1982, J.delanoy, Jspiegler, GuitarFreak, NerdyNSK, Techedgeezine, Acalamari, Barts1a, Thucydides411, EMG Blue, Chrisforster, Mikael Häggström, Hubbabridge, LA Songs, SlightlyMad, Haoao, Remi0o, 28bytes, Monkeyegg, Imperator3733, Smkoehl, Gwib, Klower, Taurius, HuskyHuskie, ITMADOG, Haseo9999, Cmbay, Nono1234, ParallelWolverine, Mike4ty4, JasonTWL, Vjardin, Winchelsea, Rockstone35, Caltas, Jerryobject, Flyer22, FSHL, Rogergummer, Lord British, Rupert baines, Ddxc, Ttrevers, Noelhurley, Radical.bison, WikipedianMarlith, ClueBot, Binksternet, GorillaWarfare, Starkiller88, Rilak, Czarkoff, Taroaldo, Wikicat, LizardJr8, Cirt, DragonBot, Drewster1829, Goodone121, Karlhendrikse, Coralmizu, Alejandrocaro35, Time2zone, Jeffmeisel, Msrill, Versus22, Un Piton, DumZiBoT, Чръный человек, Parallelized, Zodon, Airplaneman, Dsimic, Thebestofall007, Addbot, Proofreader77, Hcucu, Scientus, CanadianLinuxUser, MrOllie, LaaknorBot, Markuswiki, Jasper Deng, IOLJeff, Imirman, Tide rolls, Jarble, Ettrig, Legobot, Yobot, SabbaZ, TaBOT-zerem, Jack Boyce, Goldenthree, Danperryy, SwisterTwister, Mdegive, AnomieBOT, Enisbayramoglu, VanishedUser sdu9aya9fasdsopa, Jim1138, Piano non troppo, DaveRunner, GFauvel, Flewis, Materialscientist, RobertEves92, Threadman, Gnumer, Joshxyz, MacintoshWriter, LilHelpa, TheAMmollusc, Miym, Abce2, Bikeman333, Barfolomio, Adavis444, Gordonrox24, Elemesh, Gastonhillar, Prari, Hemant wikikosh, FrescoBot, Picklecolor2, StaticVision, RaulMetumtam, MBbjv, Winterst, Elockid, UkillaJJ, Skyerise, Meaghan, FoxBot, Sulomania, Ellwd, Glenn Maddox, Gal872875, Sreven.Nevets, NagabhushanReddy, Gg7777, Jesse V., DARTH SIDIOUS 2, Truthordaretoblockme, Beyond My Ken, WildBot, Virtimo, Helwr, EmausBot, Az29, Keithathaide, Super48paul, P3+J3^u!, Tommy2010, Serketan, Erpert, Alpha Quadrant (alt), NGPriest, Bmmxc damo, Steedhorse, L Kensington, Donner60, Jsanthara, DASHBotAV, Rmashhadi, Cswierkowski, ClueBot NG, Jeff Song, Gilderien, Chharper1, Braincricket, Widr, MerlIwBot, Nodulation, OpenSystemsPublishing, Minadasa, Cdog44, Hz.tiang, Charlie6WIND, WinampLlama, Op47, Trevayne08, Harizotoh9, Nolansdad95120, Glacialfox, Simonriley, DigitalMediaSage, Sha-256, Michael Anon, Snippy the heavily-templated snail, NimbusNiner, DavidLeighEllis, Koza1983, Christian CHABRERIE, Geoyo, Aniru919, Lagoset, Sofia Koutsouveli, RoninDusette, Kyle1009, ComsciStudent, DorothyGAlvarez and Anonymous: 643

- **Intel Atom (CPU)** *Source:* http://en.wikipedia.org/wiki/Intel%20Atom%20(CPU)?oldid=653270591 *Contributors:* David spector, Frecklefoot, Llywrch, Cyde, KAMiKAZOW, Julesd, Brigman, Conti, Ehn, Andrewman327, Taxman, Thue, Nurg, Mervyn, Zzyzx, Somercet, Andries, Digital infinity, Rchandra, Macrakis, Bo102010, Uzume, Chowbok, Elektron, Moxfyre, Imroy, NeuronExMachina, YUL89YYZ, Roo72, Bender235, Plugwash, ThierryVignaud, PatrikR, Giraffedata, MARQUIS111, Towel401, Jakew, Alansohn, Walter Görlitz, Guy Harris, Quatermass, Kenyon, Jkt, Richard Arthur Norton (1958-), Poppafuze, Scjessey, Pol098, Scootey, GregorB, MiG, Rjecina, JamesHenstridge, Laurinkus, Pmj, Rjwilmsi, Koavf, PHenry, Jlundell, Traut, Gelma, DirkvdM, X1987x, TheDJ, Jamessnell, Srleffler, Imnotminkus, Jpfagerback, Hairy Dude, DMahalko, Jengelh, Manop, Gaius Cornelius, XX55XX, TDogg310, Blitterbug, SixSix, Lucasreddinger, Hugo 87, Huangcjz, E Wing, Luk, SmackBot, Henriok, GeneralAntilles, SmackEater, Brianski, Amux, Thumperward, Onesimos, Mdwh, Aleutia, Xaxxon, Colonies Chris, Jeffreyarcand, Dethme0w, Frap, Glloq, Sh0dan, Rrburke, Metageek, Ne0Freedom, Morio, Daniel.Cardenas, Curly Turkey, Anss123, Joffeloff, JonT, Dr.K., LaMenta3, Dl2000, Hu12, Skorp, Flaming-tubesock, Andrew Hampe, Sfm 7, Fernvale, Artemgy, FleetCommand, HenkeB, Bungalowbill, CCFreak2K, Steel, Myscrnnm, Rhe br, Clovis Sangrail, Christian75, DumbBOT, Alaibot, Kozuch, Thijs!bot, Electron9, Humble Scribe, Hcobb, Oldmanbiker, Seaphoto, Lordmetroid, Smartse, CairoTasogare, Plainchips, RedWyverex, Tvbinstinct, Dimawik, JAnDbot, MER-C, Mark Grant, BrotherE, Inspector 2211, Peremen, Yosh3000, Brownout, KJRehberg, Mdulcey, Chocobogamer, Bjornen, Nicsterr, AVRS, Yura.hm, Keith D, R'n'B, CommonsDelinker, Pacdude9, Jayayess1190, J.delanoy, Whitebox, R!SC, McSly, Austin512, Ef3, Mazemode, Dubhe.sk, Ken g6, Jaimeastorga2000, My wing hk, Mrmanny09, Mlorimer, DpuTiger, Imperator3733, Bjquinn, Allan kuan1992, TXiKiBoT, Someguy1221, Delv0n2, Bojan PLOJ, Tomaxer, Winterspan, Andromeda451, Ham Pastrami, Behind The Wall Of Sleep, Oomgosh, Jhdezjr, Sasawat, Callidior, Jfromcanada, Jeo100, WikipedianMarlith, Chris D Heath, Dsalt, Jetstreak150, ClueBot, Pistazienfresser, Stygiansonic, Rilak, Boing! said Zebedee, Edknol, DerekMorr, Ykhwong, Technorj, Mewtu, Thingg, Aronzak, HumphreyW, Crowsnest, DumZiBoT, Jdwinx, Nnewton, Bobbozzo, Zodon, Jabberwoch, ZackMulgrew, Airplaneman, Deineka, Addbot, E-Tiggy, Mortense, Ocdnctx, JoshuaKuo, Scientus, LaaknorBot, Glane23, Martin0641, Jasper Deng, Abisys, Tide rolls, Lightbot, Cesiumfrog, Chinavation, Z897623, Ajoao4444, Luckas-bot, Yobot, Nghtwlkr, Helena srilowa, Bryan.burgers, Ganimoth, AnomieBOT, 1exec1, Versus785, Materialscientist, E2eamon, Undefinedsymbol, Jeremonkey, LilHelpa, Ziggy3055, Xqbot, Iadrian yu, Capricorn42, TecCougar, Trontonic, Locos epraix, User931, Narunetto, DriverDan, DooMMeeR, Cyfraw, CaptainMorgan, Demysc, 33rogers, Uusijani, MrAronnax, Brunocip, Mark Renier, Wikiphile1603, BrownsRock10, Mikesteele81, Arndbergmann, Kwns, Dsavi, Krating tong, Hgb asicwizard, Boobarkee, Jus dreamboy, Sailing ralf, Logipedia, Pklec, Wikk3d1, Blardie, Bobbyus, Txhorw, Mappy wp, MoreNet, Fitoschido, GoingBatty, ExilorX, ValC, Werieth, Cogiati, Algh0022, Rvrcopy, Tomy9510, Gsarwa, UnknownFork, Ipsign, Uziel302, ClueBot NG, Matthiaspaul, Logic2000, Gilderien, Cntras, RanmaKei, Dein Gregor, Nodulation, Technical 13, Frantz369, Zapper067, Trilemma2, Ianteraf, Deepender singla, Manguene, Kwokangus, BattyBot, Sallynice, Drbendanillo, Codename Lisa, PhelperRF, The.power.p, Pragmatool, Rotlink, Noasshats, Shelbystripes, FDMS4, Aeroid, John.Donoghue, Some Gadget Geek, Pancho507 and Anonymous: 419

- **Intel Core** *Source:* http://en.wikipedia.org/wiki/Intel%20Core?oldid=651296137 *Contributors:* Caltrop, Atlan, Cherkash, Christopherwoods, Xanzzibar, Thv, Brouhaha, Rchandra, Chowbok, SimonLyall, Thomas Bohn, DmitryKo, Apalsola, Discospinster, Hydrox, Bender235, Guy Harris, PatrickFisher, Velella, Woohookitty, Pol098, Ruud Koot, GregorB, Bubba73, Fivemack, Riki, WikiWikiPhil, DVdm, Aluvus, MGodwin, Manop, Mike411, Kvn8907, Hunnyhiteshseth, ItsDrewMiller, SmackBot, Reedy, Gilliam, Ohnoitsjamie, Vvarkey, Smileyborg, TheDamian, Gobonobo, Hvn0413, Agent007bond, Blehfu, Fernvale, MoHaG, HDCase, Augustlilleaas, FleetCommand, Pseudo-Richard, Fletcher, Yaris678, Dominicanpapi82, Dmws, JAnDbot, DuncanHill, MER-C, Bebo2good1, Trickv, Magioladitis, Kheider, Tracer9999, Rettetast, R'n'B, Tgeairn, Joeinwap, Black Kite, Hammersoft, Lexein, Sbjf, Nxavar, Echrei, Anna Lincoln, Jim Peak, Jaqen, Zuchinni one, ObjectivismLover, CavalloRazzo, Fnagaton, Dawn Bard, Bentogoa, Lord British, Jimthing, Svick, ImageRemovalBot, Martarius, ClueBot, EoGuy, Boing! said Zebedee, Rockfang, Ktr101, Pzoxicuvybtnrm, Antti29, Avoided, Badgernet, Airplaneman, Dsimic, Addbot, P4p5, Some jerk on the Internet, Damiens.rf, Melwade, Aunva6, Jasper Deng, Yobot, Legobot II, AnomieBOT, Ciphers, Hairhorn, 1exec1, Flinders Petrie, Intelati, Addihockey10, Johnny Bin, DSisyphBot, Shadowjams, Spazturtle, FrescoBot, Arndbergmann, Gligoran, Winterst, Notedgrant, LittleWink, Jschnur, Kuyamarco123, Full-date unlinking bot, Cnwilliams, Edo248, TBloemink, The Assassin047, Visite fortuitement prolongée, Specs112, Craigdsthompson, Lopifalko, Mmm333k, John of Reading, Heymid, K6ka, ZéroBot, Badrury, Cogiati, Najazjalal, Dr. A. J. Olivier, Sakurai Harumi, Donner60, Zee991, Ppw0, Azul120, Llightex, 28bot, Alcazar84, ClueBot NG, Smtchahal, Alegh, Luraj, Thompson james, Widr, Zackaback, 42girls, Wolfmmiii, Titodutta, Popish, Killswitch125, Indexallan, Gdowding, Mark Arsten, BenjaminBrock, BattyBot, Hang9417, Chemya, Rally24, Dexbot, Mogism, Pop99999, Etotheb123, TwoTwoHello, PhelperRF, Isarra (HG), Npx122sy, Frosty, Intelinside13core, OsCeZrCd, MountSynai, Applist, Haminoon, JoshChoi4121, Random23423423523423523, Studcameo, JaconaFrere, Boby2000, Vivekvigian, DiscantX, Anonguest, Computerpc0710 and Anonymous: 271

- **List of Intel Core i5 microprocessors** *Source:* http://en.wikipedia.org/wiki/List%20of%20Intel%20Core%20i5%20microprocessors?oldid=651592968 *Contributors:* F. Delpierre, Alereon, Mindmatrix, Asav, Sdschulze, Vossanova, Bubba73, Mdeegan, G-smooth2k, SmackBot, Aqualize, Wislam, Briguychau, Strohel, Inspector 2211, Paranoidmage, KenSharp, Gu1dry, CrackDragon, Addbot, Matushorvath, Luckas-bot, Yobot, AnomieBOT, 1exec1, Erik9bot, Uusijani, FrescoBot, Ralf König, Arndbergmann, Wpguy, Thilinawijesooriya, Blound, Xplosionist, واژه دای, LOTG, Tc80, Blinkiz, Ravbr, Adam J Richter, Aularon, Vitkovskiy Roman, Sakurai Harumi, SporkBot, Sbmeirow, Asdfsfs, Mystery70, Xxxxxls2, Azul120, Everlasting enigma, Frietjes, Killswitch125, Frze, Trevayne08, Hang9417, Mahadri, Mhdmj, Latias1290, Extec286 and Anonymous: 150

- **Pentium Dual-Core** *Source:* http://en.wikipedia.org/wiki/Pentium%20Dual-Core?oldid=634778879 *Contributors:* Uzume, Ukexpat, Djyang, M1ss1ontomars2k4, Guy Harris, Amram99, Denniss, Angelic Wraith, Paul1337, Pauli133, GregorB, Coneslayer, FlaBot, Gurch, Wrightbus, Chobot, Aluvus, Manop, Mike411, Cybergoth, Spliffy, SmackBot, MalafayaBot, O keyes, Letdorf, Ceecookie, BWCNY, MureninC, Thomasyen, Morio, Luigi.a.cruz, DHR, Radiant chains, Fernvale, JForget, Nathanbarlow, Alaibot, Ike-bana, Suigi, Jdlowery, Okki, Myanw, JAnDbot, Mufffin man, Jaakobou, Alex Spade, RP88, Cspan64, Cmichael, STBotD, Pcordes, VolkovBot, Imperator3733, Rafiko77, Hotdog111, Hellcat fighter, Dauthiwatchman, Iammrysh, André Pessoa, Deepakrawal121084, AlphaPyro, BBOPOPOS, VVVBot, Ham Pastrami, Yongweiru, Jhdezjr, Anakin101, ClueBot, Supertouch, Czarkoff, Boing! said Zebedee, Otolemur crassicaudatus, Sinaboy06, Geografia75, DumZiBoT, P40K, C. A. Russell, Airplaneman, Addbot, JoshuaKuo, Belekkaya, LaaknorBot, LAAFan, Luckas-bot, Yobot, 4th-otaku, Rubinbot, Piano non troppo, Xqbot, Yhljjang, Daram.G, BrownsRock10, Arndbergmann, John of Reading, Spacetronaut, H3llBot, Demonkoryu, ClueBot NG, HMSSolent, Jphill19, Henrytang194, Chepny, Emaifunction1, Gindylow, Julianprescott2604juuly, Smashmeerkat and Anonymous: 120

- **Xeon** *Source:* http://en.wikipedia.org/wiki/Xeon?oldid=653591763 *Contributors:* Pnm, Ixfd64, Stevenj, Julesd, Conti, Ehn, Anuchit, Magnus.de, Tempshill, Wernher, Pengo, David Gerard, DocWatson42, Levin, EJDyksen, Bobblewik, Chowbok, Pgan002, DaveJB, Kiteinthewind, Mormegil, Imroy, Wfaulk, Rich Farmbrough, Hydrox, Jpk, Jaap, Alistair1978, Bender235, Plugwash, MARQUIS111, MPerel, Ehurtley, Typobox, Guy Harris, Keyser Söze, CyberSkull, Water Bottle, Denniss, Schapel, Paul1337, Pauli133, Gene Nygaard, Dan100, Forderud, PdDemeter, Morkork, Jannex, Mindmatrix, LOL, Timharwoodx, Megaslow, ^demon, GregorB, Sega381, Alecv, FreplySpang, Yurik, BorgHunter, Rjwilmsi, Jgp, Bubba73, Tommy Kronkvist, Baryonic Being, Fivemack, RexNL, Aluvus, YurikBot, Borgx, RobotE, Charles Gaudette, Wengier, Yuhong, Ksyrie, Txuspe, Bovineone, Big Brother 1984, NawlinWiki, AugieWest, Seegoon, ThrashedParanoid, Tony1, Alex43223, Phandel, Chris S, Jayamohan, Cloudbound, Cffrost, VodkaJazz, Pádraic MacUidhir, KnightRider, SmackBot, Saihtam, Henriok, Ayocee, Gary Kirk, Darklock, SigurdMagnusson, Bluebot, Keegan, Big.P, Thumperward, SchfiftyThree, Letdorf, Kungming2, ABACA, TurboTad, Dethme0w, Drkirkby, Frap, Jacob Poon, A, OzymandiasX, Midnightcomm, Metageek, Aditsu, Iridescence, Morio, Ugur Basak Bot, Autopilot, DHR, Pgk1, Hectorpal, Kuru, Joffeloff, Nashif, Slakr, Beetstra, Peyre, Kvng, Dl2000, Dominicliam, Lee Carre, Coldpower27, Chovain, A1b2c345, Fernvale, CmdrObot, Mix Bouda-Lycaon, Jesse Viviano, Kamujin, CarrerCrytharis, Christian75, Jdlowery, Hardchemist, Headbomb, Ldoron, Jpchatt, Scepia, Fellix, Deflective, Davewho2, Arch dude, Belg4mit, Bongwarrior, Banzai!, Paranoidmage, Indon, Bernd vdB, TimSmall, Retroneo, Edward321, Ftiercel, CommonsDelinker, NerdyNSK, Regani, Sulimo, AntiSpamBot, GHanson, Sollosonic, DpuTiger, Joeinwap, Gerrit C. Groenenboom, Hammersoft, VolkovBot, KJSatz, Imperator3733, PNG crusade bot, Marskuzz, ScriptedGhost, KP-Adhikari, Thunderbird2, Quantpole, Aednichols, P2501, JonnyJD, Roadrunner gs, SieBot, Tinkoniko, Vexorg, Pxma, Flyer22, Sherlockindo, Jhdezjr, Lightmouse, Hobartimus, Jamiebalfour04, Anchor Link Bot, ImageRemovalBot, Loren.wilton, Martarius, Sfan00 IMG, ClueBot, Mustafa Mustamann, Rune-san, Mooo993, Rilak, Dogtoy, Paperlan, Fr3d org, TheTruthForSure, Theodric aethelfrith, BlueLikeYou, MatthieuV, Ykhwong, Pluknet, EPIC MASTER, Wdustbuster, AeonHun, HumphreyW, DumZiBoT, C. A. Russell, NellieBly, Ebbex, Dsimic, Jeff.science, Addbot, Tmacfan4321, JoshuaKuo, Hardwarefreak, Munkeevegetable, Tide rolls, Yobot, Koman90, Erikb5, Rvt1000r, ArthurBot, Victor.na, Obersachsebot, Winrarlawl, Miahac, J04n, Magnuspg, Wfischer, W Nowicki, Schafdog, Arndbergmann, Jcheckler, Rick.weber.iii, Jonesey95, A412, Xcvista, HappyCamp23, Perversus, GoneIn60, SkyMachine, Txt.file, Ecaveh, Rx5674, WCLL HK, Wire306, Dewritech, Minael, ZéroBot, Caseybutt, SCF0, Owenmann, MainFrame, Vanished 1850, ChuispastonBot, Azul120, ClueBot NG, FeiTeng1000, Snotbot, Xeonisedg, Widr, Gavinstubbs09, Satusguy, FlashSWT, Johnadamy, Andrzej w k 2, Lancededcena, Andyspam, BattyBot, Xehroz, Oranjblud, Cerabot, Kingaustin42, Velinath, Taino19xx, I am One of Many, MaXintoshPro, Applist, Monkbot, Filedelinkerbot, Unician, Sarr Cat and Anonymous: 429

- **Distributed computing** *Source:* http://en.wikipedia.org/wiki/Distributed%20computing?oldid=648678556 *Contributors:* Damian Yerrick, TwoOneTwo, Szopen, Koyaanis Qatsi, Snorre, Greg Lindahl, SimonP, Kurt Jansson, Heron, Formulax, Metz2000, Bernfarr, DennisDaniels, Edward, Derek, Marvinfreeman, Lexor, Nixdorf, Kku, Wapcaplet, Ixfd64, Dori, Anonymous56789, SebastianHelm, Alfio, Nanshu, CatherineMunro, Darkwind, Glenn, Whkoh, Rotem Dan, EdH, Rob Hooft, Ghewgill, Adam Conover, Timwi, Reddi, Ww, Hao2lian, David Shay, Dbabbitt, Optim, Raul654, Frazzydee, Jni, Donarreiskoffer, Robbot, MrJones, Brent Gulanowski, Fredrik, Kizor, R3m0t, Kristof vt, Vespristiano, Nurg, Kuszi, Hadal, Wikibot, Dbroadwell, Michael2, Giftlite, Dbenbenn, Thv, Herbee, Curps, Waxmop, Dawidl, Mboverload, Khalid hassani, Matt Crypto, Edcolins, Lupine1647, Barneyboo, Beland, Jacob grace, APH, Maximaximax, Nickptar, M1ss1ontomars2k4, Eggstasy, D6, AlexChurchill, Freakofnurture, Spiffy sperry, Mark Zinthefer, Vague Rant, Chrischan, YUL89YYZ, Bender235, S.K., Darkness Productions, Evice, Bobdoe, El C, Walden, Drektor2oo3, Gyll, Stesmo, BrokenSegue, Viriditas, Cmdrjameson, Mrdude, Haham hanuka, Paullaw, Mdd, Wayfarer, Autopilots, Ellisonch, Redxiv, Guy Harris, Atlant, Andrewpmk, DLJessup, Flata, InShaneee, Irdepesca572, Stephan Leeds, Suruena, Evil Monkey, 4c27f8e656bb34703d936fc59ede9a, SimonHova, Nigini, Nuno Tavares, Daira Hopwood, Decrease789, JonH, Ruud Koot, Jeff3000, AlbertCahalan, Qwertyus, Kbdank71, Iflipti, Rjwilmsi, Indiedan, Karmachrome, Vary, John Nixon, Pascal76, Brighterorange, AlisonW, Fred Bradstadt, FayssalF, FlaBot, JFromm, Ewlyahoocom, Bihzad, Glenn L, Chobot, DVdm, Garas, Bgwhite, YurikBot, Wavelength, Mdsmedia, Spl, Bhny, Gaius Cornelius, Bovineone, CarlHewitt, SEWilcoBot, Mkbnett, Ichatz, Neum, Voidxor, Amwebb, Nethgirb, Jeh, Tim Watson, Georgewilliamherbert, Cdiggins, Ninly, Juliano, Wsiegmund, Wikiant, Joysofpi, JoanneB, Rwwww, SmackBot, Ariedartin, David Woolley, Powo, CrypticBacon, Gilliam, RD-

Brown, PrimeHunter, EncMstr, LaggedOnUser, Scwlong, Allan McInnes, Cybercobra, Rajrajmarley, Kasperd, Dreadstar, Bejnar, Howdoesthiswo, Kuru, Sosodank, Iosef aetos, Statsone, Codepro, Bjankuloski06en, Hazzeryoda, MikeHearn, Beetstra, Trey56, Skabraham, Lee Carre, Quaeler, Buyya, Tawkerbot2, Flubeca, Gangesmaster, Page Up, Only2sea, Pmerson, WeggeBot, Ezrakilty, SuperMidget, Gortsack, CaptainMooseInc, Markov12, Stevag, Vanished User jdksfajlasd, D104, Thijs!bot, Hervegirod, Andyjsmith, Ideogram, Hala54, Papipaul, Alphachimpbot, Sorry Go Fish, JAnDbot, CosineKitty, Magioladitis, Greg Ward, Nyq, SirDuncan, Geozapf, Tedickey, Cic, David Eppstein, Jacobko, Unfactual POV, Sahkuhnder, Chocmah, Chtfn, Softguyus, LedgendGamer, Cadence-, Cpiral, McSly, Akiezun, Aervanath, Shamjithkv, DorganBot, LightningDragon, VolkovBot, Lee.Sailer, AlnoktaBOT, Philip Trueman, DragonLord, ChuckPheatt, VanishedUserABC, Spinningspark, Palaeovia, Kbrose, YonaBot, EwokiWiki, Ajtouchstone, Monkeypooisgood, Flyer22, JCLately, Hello71, Xe7al, Dust Filter, WikiLaurent, Vladrassvet, Tanvir Ahmmed, The Thing That Should Not Be, TallMagic, Nonlinear149, Worldwidegrid, Dominikiii, Alexbot, Ahmed abbas helmy, WalterGR, Warrior4321, Aitias, SoxBot III, DumZiBoT, Darkicebot, XLinkBot, Wonderfulsnow, WikHead, Slashem, RealityDysfunction, NonNobisSolum, Addbot, Proofreader77, Ramu50, Some jerk on the Internet, DOI bot, Maria C Mosak, EjsBot, AkhtaBot, MrOllie, Kisbesbot, Jarble, Frehley, Legobot, PlankBot, Luckas-bot, Yobot, AnomieBOT, Nit634, Citation bot, ArthurBot, Hahutch298, Xqbot, Capricorn42, Julianhyde, Mika au, Gilo1969, GrabBrain, Miguel in Portugal, Miym, Wizardist, AreThree, Felix.rivas, Toonsuperlove, Doulos Christos, D'ohBot, Sae1962, Wifione, Jomifica, Neilksomething, Citation bot 1, Guarani.py, I dream of horses, Jonesey95, Sohil it, RedBot, Île flottante, Jandalhandler, Trappist the monk, Yunshui, Diannaa, Tbhotch, Jesse V., Shafigoldwasser, TjBot, Eng azza, EmausBot, Kinshuk jpr19, Janakan86, Goudron, JordiGH, Cincybluffa, Wakebrdkid, Unobjectionable, ClueBot NG, Matthiaspaul, MelbourneStar, Gilderien, Advuser14, Widr, ساجد امجد ساجد, Helpful Pixie Bot, Mellorf, BG19bot, Cognitivecarbon, PlasmaTime, Kitisco, Riley Huntley, Khiladi 2010, Boufal, ChrisGualtieri, Mtriana, Catclock, Frosty, Malhelo, Maxcommejesus, Mario.virtu, Dudewhereismybike, Narendra22, Dtngo, Ma2369, Spmeu, Meyerjo and Anonymous: 365

- **Service-oriented architecture** *Source:* http://en.wikipedia.org/wiki/Service-oriented%20architecture?oldid=653416322 *Contributors:* Shd, Christian List, Frecklefoot, Edward, Michael Hardy, Repton, Kku, MartinSpamer, Haakon, Ronz, Arthur Frayn, Julesd, Stefan, MasterDirk, Chprasad, Fuzheado, Nickg, Rednblu, Pedant17, Maximus Rex, Sboehringer, Bevo, Nnh, Gentgeen, Rfc1394, Texture, Feigling, Caknuck, Bkell, Pengo, Dave6, Jonabbey, Varlaam, Khalid hassani, Pne, Matt Darby, Wmahan, Neilc, Stevietheman, Gadfium, Metlin, Beland, OwenBlacker, Usrnme h8er, Axelangeli, Lipton, Ziroby, Kustere, Breedlov, Discospinster, William Pietri, Rich Farmbrough, Rhobite, Avriette, ArnoldReinhold, ESkog, S.K., JoeSmack, Marx Gomes, Kyz, Shanes, Felagund, Simon South, Nigelj, Iain Cheyne, Polluks, Giraffedata, Cem Basman, BlueNovember, Mdd, Iolar, Kraupu, Pinar, Walter Görlitz, Dethron, Diego Moya, WTGDMan1986, M@, Warpsmith, Tayal01, Velella, Bugg, Adoble, SteveLetwin, H2g2bob, Saxifrage, Zntrip, Stuartyeates, Weyes, Angr, Jacobolus, Timosa, Cnb, Stefan Ivanovich, Varco, Mattmcc, GregorB, MauriceKA, Lastorset, ThomasOwens, Rogerd, Erebus555, Phantast, MZMcBride, Oblivious, Scorpiuss, FayssalF, Kmorozov, Penedo, RobyWayne, Chobot, Sharkface217, Gwernol, YurikBot, Whoisjohngalt, RussBot, Bovineone, SamJohnston, Alexliu, SAE1962, Mccready, Brandon, Jpbowen, Tony1, Everyguy, Sneakymoose, LeonardWalstad, LandNav, Zzuuzz, WebWombat, Hadipedia, Tvarnoe, JuJube, GraemeL, TBadger, Aarsanjani, LeonardoRob0t, Fram, Back ache, JLaTondre, DEng, Mbruggen, Halhelms, Rohitcool, Tom Morris, Binuraghavan, Riverawynter, That Guy, From That Show!, DocendoDiscimus, SmackBot, Reedy, McGeddon, Dlwl, Blueshark, Basil.bourque, Elwood j blues, Renesis, Rajah9, Netsql, Ordinant, Roma emu, Gilliam, Senfo, Ohnoitsjamie, Aamironline, Mdhmdh31, Anwar saadat, Chris the speller, Dlafont, Jjalexand, Thumperward, Philoserf, MalafayaBot, Sadads, Wykis, BBCWatcher, DHN-bot, A. B., Mogman1, Peter Campbell, Jeff DLB, Atr0x, Ashishvaid, Cybercobra, Hslayer, Enarceus, Dreadstar, Iskatel, Warren, SeanAhern, Luís Felipe Braga, RayGates, Zbalai, Mrego, Moejorris, Kuru, Spir, Ebertelsen, Soumyasch, JorisvS, Aleenf1, Martin Wagenleiter, Dfass, Nutcracker, Joe.jackson, Hulmem, Jstrachan, Kompere, Beetstra, George The Dragon, Ehheh, Evmako, Afrab null, Zorabi, Akademy, Olimpiu.pop, ShakingSpirit, Hu12, Meitar, BananaFiend, JHP, Jamehealy, IvanLanin, Vocaro, Paul Foxworthy, Igoldste, Dp462090, AGK, Linkspamremover, Tawkerbot2, JeffAdkins, CmdrObot, Libin v, Raysonho, Kensall, Nunquam Dormio, Anil Kumar Kartham, Wsibob, Cybernetic, IanDBailey, Bbarthel, Billhunt, Slp1, Gogo Dodo, ST47, Solidpoint, Odie5533, Tawkerbot4, Torc2, DumbBOT, Dblanchard, Kckid, ErnstRohlicek, Ambitus, Assentt, IvoTotev, Corycasanave, Thijs!bot, Shuumass, Hervegirod, SReddy, HappyInGeneral, Sprhodes, Z10x, Mschneid@agentos.net, RichardVeryard, Nick Number, Bob.gourley@comcast.net, Nobar, Peashy, AntiVandalBot, RobotG, Rohit Sood, Nickull, Randybardwell, Fayenatic london, Spencer, Ozgod, Wayiran, Fbahr, Barek, Yocto42, MER-C, Hasalaka, Michig, Chiya07, BrotherE, Technologyvoices, Kirrages, Daveh1, Shigdon, Magioladitis, VoABot II, MiguelMunoz, Transcendence, HouKid, Digitalfunda, Akmani, Jen2000, Tedickey, Maard, Schastain, Snowded, Straxus, Think4yourself, Kristen puckett, Wagnermr14, Hamiltonstone, Wwmbes, Allstarecho, The Rapper, Cander0000, Muzikman67, ErikTownsend, Curtbeckmann, MartinBot, Sciagent, Davidjcmorris, Axlq, Lmxspice, Ahmadar, Zzglenm, Exostor, Fpbear, Tr0lll, NetManage, Pharaoh of the Wizards, Mange01, Fioranoweb, Samiam70002, Helon, JVersteeg, Salliesatt, Robert Illes, Monster1218, Marioct, Betswiki, Grshiplett, Tonyshan, Midnight Madness, Arms & Hearts, SJP, Kmittal, Yogishpai, Doug4, Christopher norton, Cometstyles, Fkroon, Jzupan, Davidslinthicum, Sbvb, Fogartis, HighKing, Sempertinal, Stwomack, Tomerfiliba, Malik fairose, Enderminh, Erpgenie, Bramschoenmakers, Hominidx, Digerateur, Dataprit, Parker007, A4bot, Stevers1, SynAsha, Gabhala, Tiago simoes, Oxfordwang, Lradrama, Q Chris, Graham Berrisford, Maxim, Bansipatel, Hajoworldwide, Jvlock, Jaideco, Fassihi, Badja, Paladin1979, Rajeevgm, Kbrose, Colm.mcmullan, SieBot, Jzhang2007, LarsHolmberg, Yadoo86, YonaBot, Kgoarany, Y2ksw, Flyer22, CIOGuru, JCLately, Jojalozzo, CutOffTies, EnOreg, Smet, Svick, Ncw0617, Odcdtd45, Youngfu, HighFlyer12, ManOnPipes, SanderEvers, Escape Orbit, Jvlock527, Dancinggoat, Stevepolyak, ImageRemovalBot, Slwatmough, Sathya sh, Martarius, Cmaguruaj, Sfan00 IMG, Spa015, Techwonder, GarbagEcol, ClueBot, Radharajgopal, Brandguru, Gnanaguru, Nklata, MRqtH2, Mmichaelbell, Prov519, The Thing That Should Not Be, H.E. Hall, Alksentrs, Malharbarai, Sbono, Drphallus, Gunaseelapandian, SamTheButcher, Vatsakatta, StigBot, Ericsenunas, ITBusinessEdge, Rose Booth, Carusus, Dlelash, ShachiDube, Lacomoedia, NuclearWarfare, MkuhbocK, Swtechwr, ChrisKalt, Razorflame, Bruno rodin, El bot de la dieta, John at Lancelotlinc, Aprock, Bjdehut, Djmackenzie, Mymallandnews, Defragyourmind, Bpmsoa, GeoffMacartney, Expertjohn, DumZiBoT, Spinner4321, Abrobitroy, Milais, XLinkBot, Xagronaut, Heather.taylor, OWV, AngelaMartin2008, DamsonDragon, Wintaborn, AnitaRogel, ZooFari, Cesar.obach, Richard.McGuire88, Addbot, Qiuzhuang.Lian, Sean R Fox, Betterusername, Maria C Mosak, DanielGEllis, Raypereda, AroundLAin80Days, Ddhanasekharan, Kobakhet, MrOllie, LaaknorBot, Jmilgram, Empathy321, Williamglasby, Ashwineek, Manishpk, Lightbot, OlEnglish, Richard R White, Ravisriv, Jarble, Blablablob, Yobot, Themfromspace, Bunnyhop11, TaBOT-zerem, Billing geek, Danperryy, Ricky123tony, Mojei, Dactyllic, Lelguea, AncientArk, AnomieBOT, Azambataro, Aaronaberg, HelloSOAWorld, Galoubet, Kingpin13, Sz-iwbot, ItchyDE, Materialscientist, Markcowan, JamesLWilliams2010, Citation bot, Haleyga, Capricorn42, Boaworm, Kiran.kaipa, Yetasoli, Alvin Seville, Soalib, Mangst, Aligilman, Prari, FrescoBot, Riventree, Mark Renier, Ibspublishing, Sidna, Sae1962, Feravoon, BusinessMashups, HamburgerRadio, Citation bot 1, Nav141174, Winterst, Momergil, JLRedperson, Alinabalint, Gedimin, MHPSM, DavidInCambridge, E.s.cohenlevy, MertyWiki, Vinaysingla, Taylorrx, Soa101, LogAntiLog, Orangesodakid, Hanay, Darshana.jayasinghe, Vrenator, Allen4names, Crysb, Randimae, Rjabate, Starfishjones, Nottrobin, Beantowngal, Soaguy, Obankston, Lozw, Projectmbt, Bangsanegara, Jann.poppinga, Fiftytwo thirty, Quaysea, John of Reading, Faolin42, Sam Tomato, Biskup2010, Jiri Pavelka, Thecheesykid, Bollyjeff, Traxs7, GZ-Bot, Ramachandra20, ISresearcher, BioPupil, Kirby2010, Yveschaix, Sunrisesunset12345, ClueBot NG, Ptrb, Roubo, Xeroboxer, Sudhanshuss, Satellizer, Luis Mailhos, Wambugunet, Anshuman.dwivedi, Madhavan93, Carl presscott, George392, Helpful Pixie Bot, Wbm1058, GuySh, Taibah U, BG19bot, Wasbeer, Hvh22, Compfreak7, AdventurousSquirrel, Pkkao2, BattyBot, Softwareqa, DucerGraphic, Mtriana, Cj211, Czaftan, Jots and graphs, Baringmo, Mwsobol, Gabby Merger, Lemnaminor, Marinac93, Lasith011, GreggRock, Maura Driscoll, Comp.arch, Dnader90, Wtsai36, K0zka, Nishsvn, Jpmunz, Beavercreekful, Gorohoroh, Rabdill, Amenychtas, Monkbot, Theenterprisearchitect, Aerosteak, Juanchristy, Haloedscape, Tsfto01, Yinongchen and Anonymous: 960

- **Massively multiplayer online game** *Source:* http://en.wikipedia.org/wiki/Massively%20multiplayer%20online%20game?oldid=652933076 *Contributors:* William Avery, DrBob, Mrwojo, Frecklefoot, Yaroslavvb, Booyabazooka, Fred Bauder, Kidburla, Pnm, Liftarn, Gabbe, CesarB, Haakon, Netsnipe, Evercat, Conti, Furrykef, K1Bond007, Raul654, Johnleemk, Nufy8, Robbot, MrJones, SmileyChris, Havatcha, Rasmus Faber, Pengo, Dave6, Centrx, DocWatson42, Jacoplane, Brian Kendig, Everyking, WolfenSilva, Rick Block, Edcolins, Golbez, Chowbok, Geni, DarkLordSeth, Tzarius, Ratiocinate, Mattb90, Esperant, Orange Goblin, Poccil, Discospinster, Fvdham, Ponder, WikiPediaAid, Duemellon, Kbh3rd, JoeSmack, Dcrt888, TOR, CanisRufus, MBisanz, Mwanner, Sietse Snel, RoyBoy, Cretog8, Smalljim, Colorfast, John Vandenberg, ZayZayEM, Kx1186, Pearle, Gsklee, NickCatal, Alansohn, LtNOWIS, Khaim, Jtalledo, Ricky81682, Hohum, Circuitloss, Super-Magician, TheDotGamer, Drat, Marasmusine, Jeffrey O. Gustafson, OwenX, Carlos Porto, LOL, WadeSimMiser, Sega381, Toussaint, DeweyQ, Ajshm, SLWK, Alastair, Qwertyus, Rjwilmsi, Gudeldar, Egberts, Brighterorange, Bryan H Bell, Yamamoto Ichiro, Algebra, Ian Pitchford, Fyrn, Richdiesal, Coll7, SouthernNights, CarolGray, JYOuyang, RexNL, Intgr, D.brodale, Terrx, Chobot, Kakurady, NTBot, Caliah, Richard Slater, RadioFan2 (usurped), Cryptic, Bovineone, Wimt, NawlinWiki, Madcoverboy, Wahooker, Dialectric, Grafen, Jaxl, Matticus78, Mechanized, SFC9394, Wknight94, TheSeer, Blueyoshi321, K.Nevelsteen, Laibcoms, GraemeL, Vicarious, Fram, Emc2, Lando242, Draconus, JDspeeder1, SmackBot, MattieTK, Numsgil, Bobet, McGeddon, Pgk, Jagged 85, J.J.Sagnella, Premchai, Dhochron, Yellowbounder, Drttm, Carl.bunderson, Keegan, Bangy boongy, Thomasjs, MalafayaBot, Thybag, Sadads, Viewfinder, ACupOfCoffee, Darth Panda, Kotra, Can't sleep, clown will eat me, Mrn71, Allemannster, Frap, Size J Battery, Onorem, Spacefed, Xyzzyplugh, Addshore, Mosca, Nayl, Tiliqua, Radagast83, E. Sn0 =31337=, Decltype, Mr Minchin, Squashua, Eagleguy125, Weregerbil, Kalathalan, Funkywizard, Expedion, Dux0r, IonStorm, Dreslough, Bodobodot1, Scientizzle, Heimstern, Aviusoraculum, Scetoaux, Aleenf1, Wyxel, Chrisch, MarkSutton, EddieVanZant, SQGibbon, Grandpafootsoldier, Ehheh, Meco, Ryulong, Zapvet, EmmaPeel007, Kyellan, Nabeth, Tythesly, Dl2000, SubSeven, Gunsmith, Woodroar, Iridescent, Akkhima, Znitrx, Beno1000, Az1568, Courcelles, Danlev, FairuseBot, Tawkerbot2, Powerslide, Cpryd001, Stupidlegs13, CT456568, SkyWalker, CmdrObot, Mattbr, The Missing Piece, Zarex, Ric36, Tomto, Lesgainous, Nczempin, Mika1h, Leevanjackson, Green caterpillar, Some P. Erson, Wratmann, Erencexor, Nilfanion, Jonursenbach, Cagecrawler, Crossmr, Gogo Dodo, Beefnut, Tawkerbot4, Quibik, Apolaine, Smokeyblues, Lazyboyholla, Legendps, Kozuch, ErrantX, Omicronpersei8, JodyB, DJBullfish, VPliousnine, Thijs!bot, Biruitorul, Qwyrxian, Jedibob5, N5iln, Gamer007, ClosedEyesSeeing, Join, Sir Simon Tolhurst, ThirdEchelon, Hcobb, AgentPeppermint, Free2flow8, AntiVandalBot, Luna Santin, Desdewd, MrNoblet, Charlo blade123, TexMurphy, Jhsounds, Spencer, Oddity-, ChrisLamb, Gökhan, JAnDbot, Husond, MER-C, CaptSkubba, Xeno, Dream Focus, Filnik, Gogs4795, Tomst, Rocktacity, TylerStacie2005, Azzer007, VoABot II, CattleGirl, God-Emperor, Dragor, SineWave, Supsupwatz, BrianGV, PIrish, Animum, Ben Ram, Atheria, DerHexer, Kaneva, BTMendenhall, Gwern, MartinBot, MLeeson, Notmyhandle, Ultraviolet scissor flame, Sm8900, R'n'B, PrestonH, Itanius, J.delanoy, Pharaoh of the Wizards, Trusilver, Reorion, Uncle Dick, RSnook, Xangel, H4xx0r, 1mujin22, SharkD, Dared111, Uranium grenade, Masky, NewEnglandYankee, SJP, Greeves, LHOO, Akhel, TepesX, Inomyabcs, Stoneface1970, Vanished user 39948282, Bonadea, Useight, Izno, Darthnader37, RJASE1, Mmovsg, Rd556212, WebAXiSS, Dani210, Bennerhingl, Vranak, Bebietdibibi, VolkovBot, Ugriffin, Yobb, Masaruemoto, Chaos5023, VasilievVV, Soliloquial, Stealthnh, Havocbean, Minister of loco, Philip Trueman, Stateofshock, TXiKiBoT, Boxkarr, AudiS4BiTurbo, Subspace Continuum, Kww, Jameseditor, JayC, Rhettdec, Melsaran, LizardPariah, Noformation, Snowbot, Ambard, Butterscotch, Mouse is back, Kilmer-san, GAM3OV3R, Haseo9999, BlKat, Emodinol, Falcon8765, Celosia, AlleborgoBot, Bob the Rabbid, Hcagri, Hitamaru, ElsaHZT, Spascu, Amoreno65, Tibullus, SieBot, Duality98765, Balthazar, Sorarules7, Addit, Winchelsea, BloodDoll, Matt Brennen, CatalinStroescu, Flyer22, FlyxCutlery, Ezh, Allmightyduck, Kaptns, Drf0053, Svick, Baron9, Mebalzer, JohnnyMrNinja, Order of the Nameless, Blake, Anchor Link Bot, Xonnex, Cocoapropo, JL-Bot, Empire Of Sports, Chewjacka, Triple3D, Roxorius, Kanonkas, Xxe47, Elathiel, Martarius, Tylercraig85, ClueBot, LAX, Jordc1991, Snigbrook, Londium, The Thing That Should Not Be, Jordzallmighty, Creovex, Dyee, Frmorrison, Mofuggin bob, CounterVandalismBot, PMDrive1061, Excirial, Alexbot, Flightsoffancy, MatttK, Powerleveling, TaintedZebra, Carsonius, NuclearWarfare, Dirt Tyrant, OekelWm, Razorflame, Teokk, Randomran, Ark25, Muro Bot, Wyrm419, Thingg, DumZiBoT, Bridies, XLinkBot, Nepenthes, Little Mountain 5, Lstanley1979, Pirateer011, SilvonenBot, Jkbena6212, MystBot, Amazone09, Fiskbil, CharlesGunnLol, Kestry, RyanCross, Thatguyflint, Ramu50, ConCompS, Apex Glide, Pete35887, Lovepopp, Lord of Ra, EjsBot, NubXPRESS, EvanVolm, Devrit, Cst17, MrOllie, Chamal N, Glane23, Bassbonerocks, Valkyrx, Reddragonbones, RaumVogel, Say nesh, Strattonbrazil, Sammyhammy23, Tide rolls, Lightbot, Luckas Blade, PlankBot, Yobot, Taleofyouth, Ptbotgourou, Legobot II, II MusLiM HyBRiD II, Sandybremer, Groveislet, Bryan.burgers, AnomieBOT, Andrewrp, Cheetah255, Piano non troppo, Kingpin13, Xqbot, Xepra, Jojoyee, Ecstacy666, Capricorn42, TechBot, Millahnna, Nasnema, DSisyphBot, Anna Frodesiak, Nerrojin, 7h3 3L173, RibotBOT, Locobot, Shadowjams, Omgtyman, BoomerAB, Paavope, Flapperistic, LSG1-Bot, LDGB1337, Dcdannyf, Sidna, HurricanesFan5, Salta10, Wetbird15, CoreyDavis787, Jongittens, Evilmidget38, Arasaylaurel, ThatRunescaper, Myrtok, Amadsen80, Seftinho, BRUTE, Jschnur, RedBot, Writelabor, Leonardo801, Bgiff, Irbisgreif, Lotje, GamerScourge, Dinamik-bot, Ieriv, Savybones, Eshaneman, Ratatatoo, DarkScizorMaster, Reach Out to the Truth, Levininja, Faraz43, Amiteli, DARTH SIDIOUS 2, Bulgun, Vannoboy, Slonenok, Charliemcckidd, EmausBot, John of Reading, Wolhound, Tallungs, Kumul, Samredfern, RenamedUser01302013, K6ka, Agencyinc, Xubuntero, IVANis1, Fæ, MithrandirAgain, Wikignom, Zamscob3, Vgmddg, Trimutius, Erianna, KazekageTR, Omomom1, Polisher of Cobwebs, Boxlid, Longwayaway, ClueBot NG, IndigoMertel, Evergreen17, Wiki helper guy, O.Koslowski, Widr, JaybeeTheHutt, Shovan Luessi, Jeraphine Gryphon, Mikikian, Oraklus, Kttp3, Run4YourLife, Peresans, Muneil, Mark Arsten, Compfreak7, Coolio665, Workmuch, Andy-roo, BattyBot, Saígúrun, Khazar2, MadGuy7023, Mogism, Katzand, ABunnell, Graphium, HariBhol, Ganymede2222, Farbodebrahimi, Epicgenius, NewzooHQ, Hanamanteo, Betino, Tasowiki, Jacobdude5672, Glaisher, Dimitry.grossman, Jccuaki, Skr15081997, Melcous, BlitzGreg, Lubieni2, Vasinoz98, Imays76, Poseidon4231, Mediavalia, Krzykoala, MMOPhilosopher, Coolkidlol221 and Anonymous: 895

14.9.2 Images

- **File:Ambox_current_red.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/9/98/Ambox_current_red.svg *License:* CC0 *Contributors:* self-made, inspired by Gnome globe current event.svg, using Information icon3.svg and Earth clip art.svg *Original artist:* Vipersnake151, penubag, Tkgd2007 (clock)
- **File:Ambox_important.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/b/b4/Ambox_important.svg *License:* Public domain *Contributors:* Own work, based off of Image:Ambox scales.svg *Original artist:* Dsmurat (talk · contribs)
- **File:Ambox_rewrite.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/1/1c/Ambox_rewrite.svg *License:* Public domain *Contributors:* self-made in Inkscape *Original artist:* penubag

- **File:Asmp_2.gif** *Source:* http://upload.wikimedia.org/wikipedia/en/0/0b/Asmp_2.gif *License:* PD *Contributors:*
  Own work
  Original artist:
  G7a (talk) (Uploads)
- **File:Athlon64x2-6400plus.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/f/fb/Athlon64x2-6400plus.jpg *License:* CC BY 3.0 *Contributors:* Own work *Original artist:* Babylonfive David W. Smith
- **File:Closeup_of_intel_pentium_E2180.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/7/73/Closeup_of_intel_pentium_E2180.jpg *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Emaifunction1
- **File:Commons-logo.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/4/4a/Commons-logo.svg *License:* ? *Contributors:* ? *Original artist:* ?
- **File:Desktop-Linux-Mint.png** *Source:* http://upload.wikimedia.org/wikipedia/commons/4/41/Desktop-Linux-Mint.png *License:* CC BY-SA 4.0 *Contributors:* Own work *Original artist:* Benjamintf1
- **File:Distributed-parallel.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/c/c6/Distributed-parallel.svg *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Miym
- **File:Dual_Core_Generic.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/e/ec/Dual_Core_Generic.svg *License:* Public domain *Contributors:* Transferred from en.wikipedia; transferred to Commons by User:Liftarn using CommonsHelper. *Original artist:* Original uploader was CountingPine at en.wikipedia
- **File:E6750bs8.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/a/af/E6750bs8.jpg *License:* Public domain *Contributors:* Transferred from en.wikipedia; transferred to Commons by User:Liftarn using CommonsHelper. *Original artist:* Original uploader was GuitarFreak at en.wikipedia
- **File:Edit-clear.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/f/f2/Edit-clear.svg *License:* Public domain *Contributors:* The *Tango! Desktop Project*. *Original artist:*
  The people from the Tango! project. And according to the meta-data in the file, specifically: “Andreas Nilsson, and Jakub Steiner (although minimally).”
- **File:En-Service-oriented_architecture.ogg** *Source:* http://upload.wikimedia.org/wikipedia/commons/0/09/En-Service-oriented_architecture.ogg *License:* CC BY-SA 3.0 *Contributors:*
- Derivative of Service-oriented architecture *Original artist:* **Speaker:** Mangst
  Authors of the article
- **File:Folder_Hexagonal_Icon.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/4/48/Folder_Hexagonal_Icon.svg *License:* Cc-by-sa-3.0 *Contributors:* ? *Original artist:* ?
- **File:Hwloc.png** *Source:* http://upload.wikimedia.org/wikipedia/commons/9/95/Hwloc.png *License:* BSD *Contributors:* Raysonho @ Open Grid Scheduler / Grid Engine *Original artist:* The Portable Hardware Locality (hwloc) Project.
  (Screenshot by the Open Source Grid Engine Project)
- **File:Intel_Atom.png** *Source:* http://upload.wikimedia.org/wikipedia/en/e/e1/Intel_Atom.png *License:* Fair use *Contributors:*
  The logo is from the www.intel.com/personal/our-technology/mids/ website. *Original artist:* ?
- **File:Intel_Atom_2009.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/5/50/Intel_Atom_2009.svg *License:* Fair use *Contributors:*
  The logo is from the http://download.intel.com/products/atom/322868.pdf website. *Original artist:* ?
- **File:Intel_PentiumDC_2008.png** *Source:* http://upload.wikimedia.org/wikipedia/en/6/6f/Intel_PentiumDC_2008.png *License:* Fair use *Contributors:*
  The logo is from the http://www.intel.com/products/desktop/processors/pentium.htm website. *Original artist:* ?
- **File:Intel_Pentium_Dual-Core_Overclock.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/9/9d/Intel_Pentium_Dual-Core_Overclock.jpg *License:* CC BY-SA 4.0 *Contributors:* Own work *Original artist:* Julianprescott2604juuly
- **File:Intel_Xeon_DP_5110_Woodcrest.jpeg** *Source:* http://upload.wikimedia.org/wikipedia/commons/0/01/Intel_Xeon_DP_5110_Woodcrest.jpeg *License:* CC BY-SA 3.0 de *Contributors:* Own work *Original artist:* Frank Zheng
- **File:Intel_Xeon_E3-1241_v3_CPU.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/f/fe/Intel_Xeon_E3-1241_v3_CPU.jpg *License:* CC BY-SA 4.0 *Contributors:* Own work *Original artist:* Dsimic
- **File:Intel_Xeon_E5-1650_v3_CPU.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/c/c6/Intel_Xeon_E5-1650_v3_CPU.jpg *License:* CC BY-SA 4.0 *Contributors:* Own work *Original artist:* Dsimic
- **File:Intel_Xeon_E7440_open_die_at_heat_spreader.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/f/fa/Intel_Xeon_E7440_open_die_at_heat_spreader.jpg *License:* CC BY-SA 4.0 *Contributors:* Own work *Original artist:* Andrzej w k 2
- **File:Internet_map_1024.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/d/d2/Internet_map_1024.jpg *License:* CC BY 2.5 *Contributors:* Originally from the English Wikipedia; description page is/was here. *Original artist:* The Opte Project
- **File:LSPic2Smaller.png** *Source:* http://upload.wikimedia.org/wikipedia/en/d/de/LSPic2Smaller.png *License:* Fair use *Contributors:*
  Created with in-game screen capture mechanic
  Original artist: ?
- **File:Logo_Pentium_DualCore_thumb2.jpg** *Source:* http://upload.wikimedia.org/wikipedia/en/c/cd/Logo_Pentium_DualCore_thumb2.jpg *License:* Fair use *Contributors:*
  The logo may be obtained from Pentium Dual-Core.
  Original artist: ?
- **File:Memorial_gathering-WW2_Online.jpg** *Source:* http://upload.wikimedia.org/wikipedia/en/d/d1/Memorial_gathering-WW2_Online.jpg *License:* Fair use *Contributors:*
  Author Flightsoffancy *Original artist:* ?

- **File:NUMA.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/1/1f/NUMA.svg *License:* Public domain *Contributors:* Own work *Original artist:* Moop2000
- **File:Pentium_II_Xeon_450_512.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/6/66/Pentium_II_Xeon_450_512.jpg *License:* CC BY-SA 3.0 *Contributors:* Transferred from en.wikipedia
Original artist: MARQUIS111 (talk). Original uploader was MARQUIS111 at en.wikipedia
- **File:Portal-puzzle.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/f/fd/Portal-puzzle.svg *License:* Public domain *Contributors:* ? *Original artist:* ?
- **File:Question_book-new.svg** *Source:* http://upload.wikimedia.org/wikipedia/en/9/99/Question_book-new.svg *License:* Cc-by-sa-3.0 *Contributors:*

  Created from scratch in Adobe Illustrator. Based on Image:Question book.png created by User:Equazcion *Original artist:*
  Tkgd2007
- **File:SMP_-_Symmetric_Multiprocessor_System.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/1/1c/SMP_-_Symmetric_Multiprocessor_System.svg *License:* CC BY-SA 3.0 *Contributors:* Own work *Original artist:* Ferry24.Milan
- **File:SOA_Elements.png** *Source:* http://upload.wikimedia.org/wikipedia/commons/d/d4/SOA_Elements.png *License:* CC BY 2.5 *Contributors:* Dirk Krafzig, Karl Banke, and Dirk Slama. Enterprise SOA. Prentice Hall, 2005 *Original artist:* Florian Lindner (designer)
- **File:SOA_Metamodel.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/0/06/SOA_Metamodel.svg *License:* GFDL *Contributors:* self-made, based on SOA Meta Model.jpg by David S. Linthicum *Original artist:* Loïc Corbasson, created with en:OOo Draw (ODG source file available on request)
- **File:SOMF_V_2.0.jpg** *Source:* http://upload.wikimedia.org/wikipedia/en/7/77/SOMF_V_2.0.jpg *License:* PD *Contributors:* ? *Original artist:* ?
- **File:Shared_memory.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/f/f2/Shared_memory.svg *License:* CC-BY-SA-3.0 *Contributors:* en:Image:Shared Memory.jpeg *Original artist:* en:User:Khazadum, User:Stannered
- **File:Sound-icon.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/4/47/Sound-icon.svg *License:* LGPL *Contributors:* Derivative work from Silsor's versio *Original artist:* Crystal SVG icon set
- **File:Wiki_letter_w_cropped.svg** *Source:* http://upload.wikimedia.org/wikipedia/commons/1/1c/Wiki_letter_w_cropped.svg *License:* CC-BY-SA-3.0 *Contributors:*
- Wiki_letter_w.svg *Original artist:* Wiki_letter_w.svg: Jarkko Piiroinen
- **File:Xeon_X5355_Clovertown.jpg** *Source:* http://upload.wikimedia.org/wikipedia/commons/8/84/Xeon_X5355_Clovertown.jpg *License:* CC BY-SA 3.0 de *Contributors:* Own work *Original artist:* Frank Zheng

14.9.3 Content license

- Creative Commons Attribution-Share Alike 3.0