Overview of recent supercomputers

Version: 2002 **Date:** July 2002

Prepared by Aad J. van der Steen,

NCF/Utrecht University

In this report we give an overview of parallel and vector computers which are currently available or will become available within a short time frame from vendors; no attempt is made to list all machines that are still in the research phase. The machines are described according to their architectural class. Shared and distributed-memory SIMD an MIMD machines are discerned. The information about each machine is kept as compact as possible. Moreover, no attempt is made to quote price information as this is often even more elusive than the performance of a system.

Please not that this is not the original report. The original report has been translated into XML. The XML version is combined with information from other supercomputing information sources that are also available in XML. This way, the report has been augemented with:

- ***For each system, all machines in the current TOP500 list of supercomputers is listed.***
- ***A summary table has been added. The average age of system architectures is calculated***
- ***A description from the company, taken from the EnterTheGrid catalogue has been added.***

XML-translation, XSLT-stylesheets and processing: Ad Emmen (Genias Benelux).

Table of Contents

Part 1

Chapter 1
Introduction and account

This is the twelfth edition of a report in which we attempt to give an overview of parallel and vector systems that are commercially available or are expected to become available within a short time frame (typically a few months to half a year). We choose the expression "attempt" deliberately because the market of parallel- and vector machines is highly evasive: the rate with which systems are introduced - and disappear again - is very high and therefore the information will probably be only approximately valid. Nevertheless, we think that such an overview is useful for those who want to obtain a general idea about the various means by which these systems strive at high performance, especially when it is updated on a regular basis.

We will try to be as up-to-date and compact as possible and on these grounds we think there is a place for this report. At this moment systems appearing on and disappearing from the market are approximately in balance. One of the reasons for this seems to be the ASCI in the USA that has given a big impulse to the HPC industry, at least in the USA. Furthermore, there is the more or less natural wave motion of older systems that are withdrawn and are replaced by newer models. Generally, one could say that the trend of the past few years in which more systems disappeared than new ones were introduced does not seem to continue. Only time can tell whether this stabilisation is permanent.

A trend that seems to emerge is that most new systems look as minor variations on the same theme: clusters of RISC-based Symmetric Multi-Processing (SMP) nodes which in turn are connected by a fast network Cull98 consider this as a natural architectural evolution. However, it may also be argued that the requirements formulated in the ASCI program has steered these systems in this direction.

The supercomputer market is a very dynamic one and this is especially true for the Beowulf clusters that have emerged at a tremendous rate in the last few years. The number of vendors that sell pre-configured clusters has boomed accordingly and, at least for this issue, we have decided *not* to include such configurations in this report: the speed with which cluster companies and systems appear and disappear makes this almost impossible. We will briefly comment on cluster characteristics and their position relative to other supercomputers in section clusters though.
For the tightly-coupled or "integrated" parallel systems, however, we can by updating this report at least follow the main trends in popular and emerging architectures. The details of the systems be reported do not allow the report to be shorter than in former years: between 40--50 pages.

As of the 11th issue we decided to introduce a section that describes the dominant processors in some detail. This seems fit as the processors are the heart of the systems. We do that in section processors .

The rule for including systems is as follows: they should be either available commercially at the time of appearance of this report, or within 6 months thereafter. This excludes interesting research systems like the ASCI systems, at the Sandia, Los Alamos, and Lawrence Livermore National Laboratories in the USA (all with a measured performance of more than 1.5 Tflop/s) because they are not marketed and only available at the institutes mentioned and, therefore, of not much benefit to the supercomputer community at large.

The rule that systems should be available within a time-span of 6 months is to avoid confusion by describing systems that are announced much too early, just for

marketing reasons and that will not be available to general users within a reasonable time. We also have to refrain from including all generations of a system that are still in use. Therefore, for instance, we do not include the IIBM SP1, the Cray T90 series anymore although these systems are still in use. Generally speaking, we include machines that are presently marketed or will be marketed within 6 months. To add to the information given in this report, we quote the Web addresses of the vendors because the information found there may be more recent than what can be provided here. On the other hand, such pages should be read with care because it will not always be clear what the status is of the products described there.
Some vendors offer systems that are identical in all respects except in the clock cycle of the nodes (examples are the SGI Origin3000 series and the Fujitsu AP3000). In these cases we always only mention the models with the fastest clock as it will be always possible to get the slower systems and we presume that the reader is primarily interested in the highest possible speeds that can be reached with these systems.

Until the eighth issue of this report we ordered the systems by their architectural classes as explained in section architecture. However, this distinction became more and more artificial as is explained in the same section. Therefore all systems described are simply listed alphabetically. In the header of each system description the machine type is provided. There is referred to the architectural class for as far this is relevant. We omit price information which in most cases is next to useless. If available, we will give some information about performances of systems based on user experiences instead of only giving theoretical peak performances. Here we have adhered to the following policy: We try to quote *best measured performances*, if available, thus providing a more realistic upper bound than the theoretical peak performance. We hardly have to say that the speed range of supercomputers is enormous, so the best measured performance will not always reflect the performance of the reader's favourite application. When we give performance information, it is not always possible to quote all sources and in any case if this information seems (or is) biased, this is entirely the responsibility of the author of this report. He is quite willing to be corrected or to receive additional information from anyone who is in the position to do so.

Although for the average user the appearance of new systems rapidly becomes more and more alike, it is still useful to dwell a little on the architectural classes that underlie this appearance. It gives some insight in the various ways that high performance is achieved and a feeling why machines perform as they do. This is done in the section on architecture which will be referred to repeatedly in sections that describe the various systems.
Up till the tenth issue we included a section gone some systems are listed that disappeared from the market. We reduced that section in the printed and PostScript versions from now on because it tends to take an unreasonable part of the total text. Still, because this information is of interest to a fair amount of readers and it gives insight in the field of the historical development of supercomputing over the last 12 years, this information will still be available in full at gone. In section comes.html we present some systems that are under development and have a fair chance to appear on the market. Because of the addition of the section on processors that introduces many technical terms, also a glossary.html is included.

The overview given in this report concentrates on the computational capabilities of the systems discussed. To do full justice to all assets of present days high-performance computers one should list their I/O performance and their connectivity possibilities as well. However, the possible permutations of configurations even for one model of a certain system often are so large that they would multiply the volume of this report, which we tried to limit for greater clarity. So, not all features of the systems discussed will be present. Still we think (and certainly hope) that the impressions obtained from the entries of the individual machines may be useful to many. We also omitted some systems that may be characterised as "high-performance" in the fields of database management, real-time computing, or visualisation. Therefore, as we try to give an overview for the area of general scientific and technical computing, systems that are primarily meant for database retrieval like the AT&T GIS systems or concentrate

exclusively on the real-time user community, like Concurrent Computing Systems, are not discussed in this report. Furthermore, we have set a threshold of about 10 Gflop/s for systems to appear in this report as, at least with regard to theoretical peak performance, single CPUs often exceed 1 Gflop/s although their actual performance may be an entirely other matter.

Although most terms will be familiar to many readers, we still think it is worthwhile to give some of the definitions in the architecture because some authors tend to give them a meaning that may slightly differ from the idea the reader already has acquired.

Lastly, we should point out that the WWW version is available at various places. The URLs are:
USA: http://www.netlib.org/utk/papers/advanced-computers/
Europe: http://www.nwo.nl/ncf/overview-src.
Europe: http://www.phys.uu.nl/~steen/overview/overview02.html.
Europe: http://www.euroben.nl/reports/overview02.html.

Chapter 2
The main architectural classes

Section 1

The main architectural classes

Before going on to the descriptions of the machines themselves, it is important to consider some mechanisms that are or have been used to increase the performance. The hardware structure or *architecture* determines to a large extent what the possibilities and impossibilities are in speeding up a computer system beyond the performance of a single CPU. Another important factor that is considered in combination with the hardware is the capability of compilers to generate efficient code to be executed on the given hardware platform. In many cases it is hard to distinguish between hardware and software influences and one has to be careful in the interpretation of results when ascribing certain effects to hardware or software peculiarities or both. In this chapter we will give most emphasis to the hardware architecture. For a description of machines that can be considered to be classified as "high-performance" one is referred to Cull98 and Steen95.

Since many years the taxonomy of Flynn Flynn72 has proven to be useful for the classification of high-performance computers. This classification is based on the way of manipulating of instruction and data streams and comprises four main architectural classes. We will first briefly sketch these classes and afterwards fill in some details when each of the classes is described separately.

- **SISD** machines: These are the conventional systems that contain one CPU and hence can accommodate one instruction stream that is executed serially. Nowadays many large mainframes may have more than one CPU but each of these execute instruction streams that are unrelated. Therefore, such systems still should be regarded as (a couple of) SISD machines acting on different data spaces. Examples of SISD machines are for instance most workstations like those of DEC, Hewlett-Packard, and Sun Microsystems. The definition of SISD machines is given here for completeness' sake. We will not discuss this type of machines in this report.
- **SIMD** machines: Such systems often have a large number of processing units, ranging from 1,024 to 16,384 that all may execute the same instruction on different data in lock-step. So, a single instruction manipulates many data items in parallel. Examples of SIMD machines in this class are the CPP DAP Gamma II and the Quadrics Apemille.
- Another subclass of the SIMD systems are the vectorprocessors. Vectorprocessors act on arrays of similar data rather than on single data items using specially structured CPUs. When data can be manipulated by these vector units, results can be delivered with a rate of one, two and --- in special cases --- of three per clock cycle (a clock cycle being defined as the basic internal unit of time for the system). So, vector processors execute on their data in an almost parallel way but only when executing in vector mode. In this case they are several times faster than when executing in conventional scalar mode. For practical purposes vectorprocessors are therefore mostly regarded as SIMD machines. An example of such a system is for instance the NEC SX-6i.
- **MISD** machines: Theoretically in these type of machines multiple instructions should act on a single stream of data. As yet no practical machine in this class has been constructed nor are such systems easily to conceive. We will disregard

them in the following discussions.

- **MIMD**} machines: These machines execute several instruction streams in parallel on different data. The difference with the multi-processor SISD machines mentioned above lies in the fact that the instructions and data are related because they represent different parts of the same task to be executed. So, MIMD systems may run many sub-tasks in parallel in order to shorten the time-to-solution for the main task to be executed. There is a large variety of MIMD systems and especially in this class the Flynn taxonomy proves to be not fully adequate for the classification of systems. Systems that behave very differently like a four-processor NEC SX-6 and a thousand processor SGI/Cray T3E fall both in this class. In the following we will make another important distinction between classes of systems and treat them accordingly.
    - **Shared memory systems**: Shared memory systems have multiple CPUs all of which share the same address space. This means that the knowledge of where data is stored is of no concern to the user as there is only one memory accessed by all CPUs on an equal basis. Shared memory systems can be both SIMD or MIMD. Single-CPU vector processors can be regarded as an example of the former, while the multi-CPU models of these machines are examples of the latter. We will sometimes use the abbreviations SM-SIMD and SM-MIMD for the two subclasses.
    - **Distributed memory systems**: In this case each CPU has its own associated memory. The CPUs are connected by some network and may exchange data between their respective memories when required. In contrast to shared memory machines the user must be aware of the location of the data in the local memories and will have to move or distribute these data explicitly when needed. Again, distributed memory systems may be either SIMD or MIMD. The first class of SIMD systems mentioned which operate in lock step, all have distributed memories associated to the processors. As we will see, distributed-memory MIMD systems exhibit a large variety in the topology of their connecting network. The details of this topology are largely hidden from the user which is quite helpful with respect to portability of applications. For the distributed-memory systems we will sometimes use DM-SIMD and DM-MIMD to indicate the two subclasses.

As already alluded to, although the difference between shared- and distributed memory machines seems clear cut, this is not always entirely the case from user's point of view. For instance, the late Kendall Square Research systems employed the idea of "virtual shared memory" on a hardware level. Virtual shared memory can also be simulated at the programming level: A specification of High Performance Fortran (HPF) was published in 1993 HPF93 which by means of compiler directives distributes the data over the available processors. Therefore, the system on which HPF is implemented in this case will look like a shared memory machine to the user. Other vendors of Massively Parallel Processing systems (sometimes called MPP systems), like HP and SGI, also are able to support proprietary virtual shared-memory programming models due to the fact that these physically distributed memory systems are able to address the whole collective address space. So, for the user such systems have one ***global address space*** spanning all of the memory in the system. We will say a little more about the structure of such systems in the ccNUMA.html#ccNUMA section. In addition, packages like TreadMarks (Amza95) provide a virtual shared memory environment for networks of workstations.

Distributed processing takes the DM-MIMD concept one step further: instead of many integrated processors in one or several boxes, workstations, mainframes, etc., are connected by (Gigabit) Ethernet, FDDI, or otherwise and set to work concurrently on tasks in the same program. Conceptually, this is not different from DM-MIMD computing, but the communication between processors is often orders of magnitude slower. Many packages to realise distributed computing are available. Examples of these are PVM (standing for **P**arallel **V**irtual **M**achine) Geist94, and MPI (**M**essage **P**assing **I**nterface, MPI1), MPI2). This style of programming, called the "message passing" model has becomes so much accepted that PVM and MPI have been

adopted by virtually all major vendors of distributed-memory MIMD systems and even on shared-memory MIMD systems for compatibility reasons. In addition there is a tendency to cluster shared-memory systems, for instance by HiPPI channels, to obtain systems with a very high computational power. E.g., the NEC SX-6, and the Cray SV1ex have this structure. So, within the clustered nodes a shared-memory programming style can be used while between clusters message-passing should be used.

For SM-MIMD systems we should mention OpenMP Chandra01, that can be used to parallelise Fortran and C(++) programs by inserting comment directives (Fortran 77/90/95) or pragmas (C/C++) into the code. OpenMP has quickly been adopted by the major vendors and has become a well established standard for shared memory systems.

Section 2

Shared-memory SIMD machines

This subclass of machines is practically equivalent to the single-processor vectorprocessors, although other interesting machines in this subclass have existed (viz. VLIW machines Steen90). In the block diagram in figure 1 we depict a generic model of a vector architecture.

-

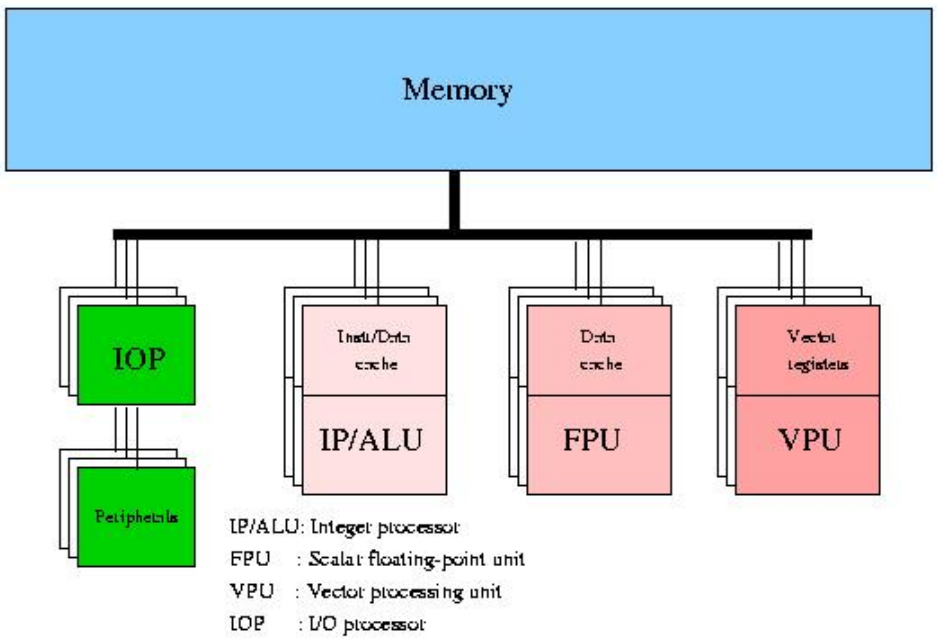

Figure 1 . Block diagram of a vector processor

The single-processor vector machine will have only one of the vectorprocessors depicted and the system may even have its scalar floating-point capability shared with the vector processor (as was the case in some gone). It may be noted that the VPU does not show a cache. The majority of vectorprocessors do not employ a cache anymore. In many cases the vector unit cannot take advantage of it and execution speed may even be unfavourably affected because of frequent cache overflow.

Although vectorprocessors have existed that loaded their operands directly from memory and stored the results again immediately in memory (CDC Cyber 205, ETA-10), all present-day vectorprocessors use vector registers. This usually does not impair the speed of operations while providing much more flexibility in gathering operands and manipulation with intermediate results.

Because of the generic nature of Figure #figvecpr no details of the interconnection between the VPU and the memory are shown. Still, these details are very important for the effective speed of a vector operation: when the bandwidth between memory and the VPU is too small it is not possible to take full advantage of the VPU because it has to wait for operands and/or has to wait before it can store results. When the ratio of arithmetic to load/store operations is not high enough to compensate for such situations, severe performance losses may be incurred.

The influence of the number of load/store paths for the dyadic vector operation $c = a + b$ (a, b, and c vectors) is depicted in figure 2.

-

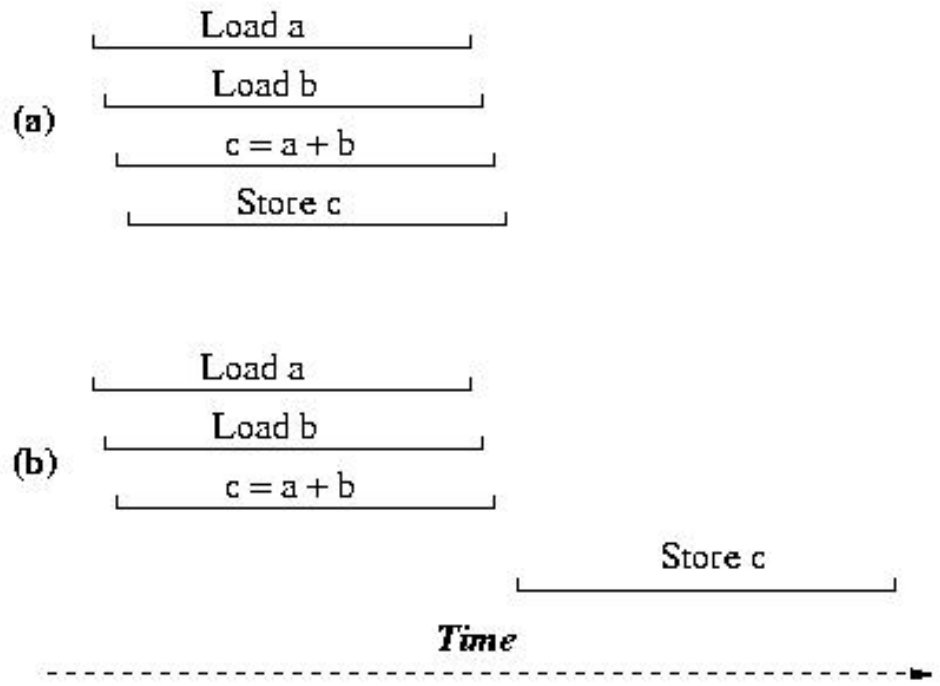

Figure 2 . Schematic diagram of a vector addition. Case (a) when two load- and one store pipe are available; case (b) when two load/store pipes are available.

Because of the high costs of implementing these datapaths between memory and the VPU, often compromises are sought and the number of systems that have the full required bandwidth (i.e., two load operations and one store operation at the *same* time) is limited. In fact, in the vector systems marketed today this high bandwidth thus not occur any longer. Vendors rather rely on additional caches and other tricks to hide the lack of bandwidth.

The VPUs are shown as a single block in Figure #figvecpr. Yet, again there is a considerable diversity in the structure of VPUs. Every VPU consists of a number of

vector functional units, or "pipes" that fulfill one or several functions in the VPU. Every VPU will have pipes that are designated to perform memory access functions, thus assuring the timely delivery of operands to the arithmetic pipes and of storing the results in memory again. Usually there will be several arithmetic functional units for integer/logical arithmetic, for floating-point addition, for multiplication and sometimes a combination of both, a so-called compound operation. Division is performed by an iterative procedure, table look-up, or a combination of both using the add and multiply pipe. In addition, there will almost always be a mask pipe to enable operation on a selected subset of elements in a vector of operands. Lastly, such sets of vector pipes can be replicated within one VPU (2- up to 16-fold replication are occurs). Ideally, this will increase the performance per VPU by the same factor provided the bandwidth to memory is adequate.

Section 3

Distributed-memory MIMD machines

The class of DM-MIMD machines is undoubtly the fastest growing part in the family of high-performance computers. Although this type of machines is more difficult to deal with than shared-memory machines and DM-SIMD machines. The latter type of machines are processor-array systems in which the data structures that are candidates for parallelisation are vectors and multi-dimensional arrays that are laid out automatically on the processor array by the system software. For shared-memory systems the data distribution is completely transparant to the user. This is quite different for DM-MIMD systems where the user has to distribute the data over the processors and also the data exchange between processors has to be performed explicitely. The initial reluctance to use DM-MIMD machines seems to have been decreased. Partly this is due to the now existing standard for communication software (Geist94) and partly because, at least theoretically, this class of systems is able to outperform all other types of machines.

The advantages of DM-MIMD systems are clear: the bandwidth problem that haunts shared-memory systems is avoided because the bandwidth scales up automatically with the number of processors. Furthermore, the speed of the memory which is another critical issue with shared-memory systems (to get a peak performance that is comparable to that of DM-MIMD systems, the processors of the shared-memory machines should be very fast and the speed of the memory should match it) is less important for the DM-MIMD machines, because more processors can be configured without the afore mentioned bandwidth problems.

Of course, DM-MIMD systems also have their disadvantages: The communication between processors is much slower than in SM-MIMD systems, and so, the synchronisation overhead in case of communicating tasks is generally orders of magnitude higher than in shared-memory machines. Moreover, the access to data that are not in the local memory belonging to a particular processor have to be obtained from non-local memory (or memories). This is again on most systems very slow as compared to local data access. When the structure of a problem dictates a frequent exchange of data between processors and/or requires many processor synchronisations, it may well be that only a very small fraction of the theoretical peak speed can be obtained. As already mentioned, the data- and task decomposition are factors that mostly have to be dealt with explicitly, which may be far from trivial.

It will be clear from the paragraph above that also for DM-MIMD machines both the topology and the speed of the datapaths are of crucial importance for the practical usefulness of a system. Again, as in the section on sm-mimd.html, the richness of the connection structure has to be balanced against the costs. Of the many conceivable interconnection structures only a few are popular in practice. One of these is the so-called hypercube topology as depicted in figure 3.

-

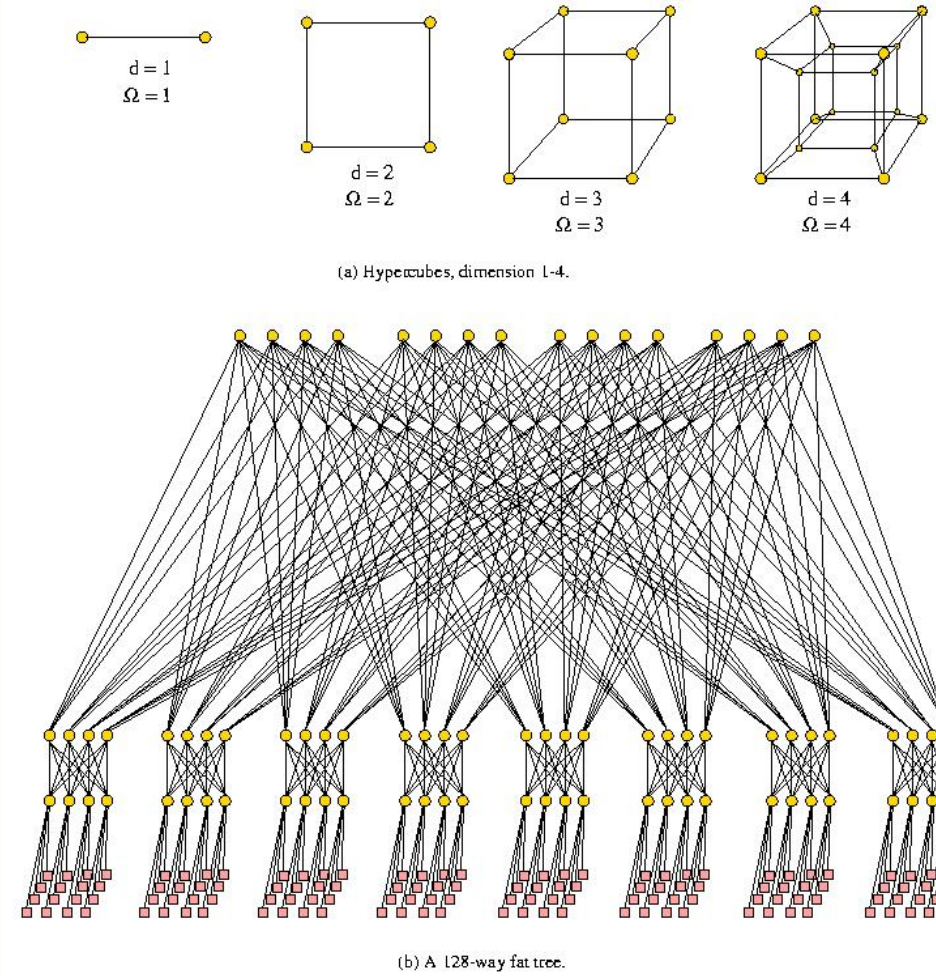

Figure 3 . Some often used networks for DM machine types

A nice feature of the hypercube topology is that for a hypercube with 2^d nodes the number of steps to be taken between any two nodes is at most *d*. So, the dimension of the network grows only logarithmically with the number of nodes. In addition, theoretically, it is possible to simulate any other topology on a hypercube: trees, rings, 2-D and 3-D meshes, etc. In practice, the exact topology for hypercubes does not matter too much anymore because all systems in the market today employ what is called "wormhole routing". This means that a message is send from *i* to node *j* a header message is sent from *i* to *j*, resulting in a direct connection between these nodes. As soon this connection is established, the data proper is sent through this connection without disturbing the operation of the intermediate nodes. Except for a small amount of time in setting up the connection between nodes, the communication time has become virtually independent of the distance between the nodes. Of course, when several messages in a busy network have to compete for the same paths, waiting times are incurred as in any network that does not directly connect any processor to all others and often rerouting strategies are employed to circumvent busy links.

Another cost-effective way to connect a large number of processors is by means of a *fat tree*. In principle a simple tree structure for a network is sufficient to connect all nodes in a computer system. However, in practice it turns out that near the root of the

tree congestion occurs because of the the concentration of messages that first have to traverse the higher levels in the tree structure before they can descend again to their target nodes. The fat tree amends this shortcoming by providing more bandwidth (mostly in the form of multiple connections) in the higher levels of the tree. An example of a fat tree with a bandwidth in the highest level that is doubled with respect to the lower levels is shown in Figure #netw2.

A number of massively parallel DM-MIMD systems seem to favour a 2-D or 3-D mesh (torus) structure. The rationale for this seems to be that most large-scale physical simulations can be mapped efficiently on this topology and that a richer interconnection structure hardly pays off. However, some systems maintain (an) additional network(s) besides the mesh to handle certain bottlenecks in data distribution and retrieval Hori91.

A large fraction of systems in the DM-MIMD class employ crossbars. For relatively small amounts of processors (in the order of 64) this may be a direct or 1-stage crossbar, while to connect larger numbers of nodes multi-stage crossbars are used, i.e., the connections of a crossbar at level 1 connect to a crossbar at level 2, etc., instead of directly to nodes at more remote distances in the topology. In this way it is possible to connect in the order of a few thousands of nodes through only a few switching stages. In addition to the hypercube structure, other logarithmic complexity networks like Butterfly-, Omega-, or shuffle-exchange networks are often employed in such systems.

As with SM-MIMD machines, a node may in principle consist of any type of processor (scalar or vector) for computation or transaction processing together with local memory (with or without cache) and, in almost all cases, a separate communication processor with links to connect the node to its neighbours. Nowadays, the node processors are mostly off-the-shelf RISC processors sometimes enhanced by vector processors. A problem that is peculiar to these DM-MIMD systems is the mismatch of communication vs. computation speed that may occur when the node processors are upgraded, without also speeding up the intercommunication. In some cases this may result in turning computational-bound problems into communication-bound problems.

Section 4

ccNUMA machines

As already mentioned in the introduction, a trend can be observed to build systems that have a rather small (up to 16) number of RISC processors that are tightly integrated in a cluster, a Symmetric Multi-Processing (SMP) node. The processors in such a node are virtually always connected by a 1-stage crossbar while these clusters are connected by a less costly network. Such a system may look as depicted in figure 4. Note that in Figure 6 all CPUs in a cluster are connected to a common part of the memory.

-

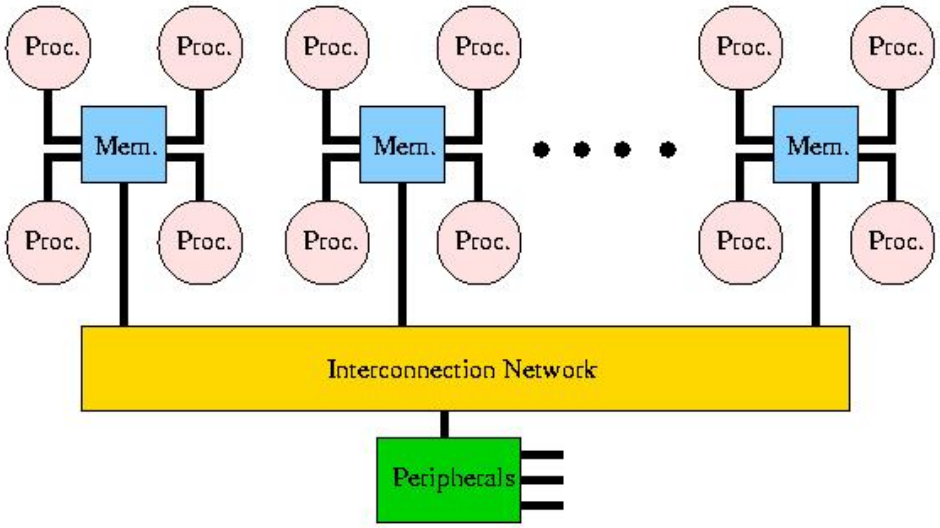

Figure 4 . Block diagram of a system with a 'hybrid' network: clusters of four CPUs are connected by a crossbar. The clusters are connected by a less expensive network, e.g., a Butterfly network.

This is similar to the policy mentioned for large vectorprocessor ensembles mentioned above but with the important difference that all of the processors can access all of the address space if necessary. The most important ways to let the SMP nodes share their memory are S-COMA **S**imple **C**ache-**O**nly **M**emory **A**rchitecture) and ccNUMA, which stands for **C**ache **C**oherent **N**on-**U**niform **M**emory **A**ccess. Therefore, such systems can be considered as SM-MIMD machines. On the other hand, because the memory is physically distributed, it cannot be guaranteed that a data access operation always will be satisfied within the same time. In S-COMA systems the cache hierarchy of the local nodes is extended to the memory of the other nodes. So, when data is required that does not reside in the local node's memory it is retrieved from the memory of the node where it is stored. In ccNUMA this concept is further extended in that all memory in the system is regarded (and addressed) globally. So, a data item may not be physically local but logically it belongs to one shared address space. Because the data can be physically dispersed over many nodes, the access time for different data items may well be different which explains the term non-uniform data access. The term "Cache Coherent" refers to the fact that for all CPUs any variable that is to be used must have a consistent value. Therefore, is must be assured that the caches that provide these variables are also consistent in this respect. There are various ways to ensure that the caches of the CPUs are coherent. One is the ***snoopy bus protocol*** in which the caches listen in on transport of variables to any of the CPUs and update their own copies of these variables if they have them. Another way is the ***directory memory***, a special part of memory which enables to keep track of the all copies of variables and of their validness.
Presently, no commercially available machine uses the S-COMA scheme. By contrast, there are several popular ccNUMA systems (HP SuperDome, SGI Origin3000) commercially available.

For all practical purposes we can classify these systems as being SM-MIMD machines also because special assisting hardware/software (such as a directory memory) has been incorporated to establish a single system image although the memory is physically distributed.

Section 5

Clusters

The adoption of clusters, collections of workstations/PCs connected by a local network, has virtually exploded since the introduction of the first Beowulf cluster in 1994. The attraction lies in the (potentially) low cost of both hardware and software and the control that builders/users have over their system. The interest for clusters can be seen for instance from the active IEEE Task Force on Cluster Computing (TFCC) which regularly issues a White Paper in which the current status of cluster computing is reviewed TFCC. Also books how to build and maintain clusters have greatly added to their popularity (see, e.g.,Ster99 and . As the cluster scene becomes relatively mature and an attractive market, large HPC vendors as well as many start-up companies have entered the field and offer more or less ready out-of-the-box cluster solutions for those groups that do not want to build their cluster from scratch.

The number of vendors that sell cluster configurations has become so large that it is not sensible to include all these products in this report. In addition, there is generally a large difference in the usage of clusters and their more integrated counterparts that we discuss in the following sections: clusters are mostly used for ***capability computing*** while the integrated machines primarily are used for ***capacity computing***. The first mode of usage meaning that the system is employed for one or a few programs for which no alternative is readily available in terms of computational capabilities. The second way of operating a system is in employing it to the full by using the most of its available cycles by many, often very demanding, applications and users. Traditionally, vendors of large supercomputer systems have learned to provide for this last mode of operation as the precious resources of their systems were required to be used as effectively as possible. By contrast, Beowulf clusters are mostly operated through the Linux operating system (a small minority using Microsoft Windows) where these operating systems either miss the tools or these tools are relatively immature to use a cluster well for capacity computing. However, as clusters become on average both larger and more stable, there is a trend to use them also as computational capacity servers. In Steen00 is looked at some of the aspects that are necessary conditions for this kind of use like available cluster management tools and batch systems. In the same study also the performance on an application workload was assessed, both on a RISC (Compaq Alpha) based configuration and on Intel Pentium III based systems. An important, but not very surprising conclusion was that the speed of the network is very important in all but the most compute bound applications. Another notable observation was that using compute nodes with more than 1 CPU may be attractive from the point of view of compactness and (possibly) energy and cooling aspects, but that the performance can be severely damaged by the fact that more CPUs have to draw on a common node memory. The bandwidth of the nodes is in this case not up to the demands of memory intensive applications.

Fortunately, there is nowadays a fair choice of communication networks available in clusters. Of course 100 Mb/s Ethernet is always possible, which is attractive for economic reasons, but has the drawbacks of a very modest maximum bandwidth (about 10 MB/s) and a high latency (about 100 µs). Gigabit Ethernet has a maximum bandwidth that is 10 times higher but has about the same latency. Alternatively, there are for instance networks that operate from user space, like Myrinet Myr00, Giganet cLAN Gigan01, and SCI JaLa90. The first two have maximum bandwidths in the order of 100 MB/s and a latency in the range of 15--20 µs. SCI has a higher bandwidth (400--500 MB/s theoretically) and a latency under 10 µs. The latter solution is more costly but is nevertheless employed in some cluster configurations. The network speeds as shown by Myrinet, cLAN, and, certainly, SCI is more or less on par with some integrated parallel systems as discussed later. So, possibly apart from the speed of the processors and of the software that is provided by the vendors of DM-MIMD supercomputers, the distinction between clusters and this class of

machines becomes rather small and will undoubtly decrease in the coming years.

The best starting point for the state-of-the-art in cluster computing is given in the TFCC White Paper TFCC already mentioned. It gives an pointers to available products, both hardware and software, open questions and the focus of the present research regarding these questions.

Section 6

Processors

In comparison to 10 years ago the processor scene has become drastically different. While in the period 1980--1990, the proprietary processors and in particular the vectorprocessors were the driving forces of the supercomputers of that period, today that role has been taken on by common off-the-shelf RISC processors. In fact there are only three companies left that produce vector systems while all other systems that are offered are based on RISC CPUs (except the Cray MTA-2). We think, therefore, that it is useful to give a brief description of the main processors that populate the present supercomputers and look a little ahead to the processors that will follow in the coming year.

The modern RISC processors generally have a clock frequency that is lower than that of the Intel Pentium 3/4 processors or the corresponding AMD Intel look-alikes. However, they have a number of facilities that put them ahead in the speed of floating-point oriented applications. Firstly, all RISC processors are able to deliver 2 or more 64-bit floating-point results in one clock cycle. Secondly, all of them feature out-of-order instruction execution, which enhances the number of instructions per cycle that can be processed (although the newer AMD processors also have 2-way floating-point instruction issuing and out-of-order execution, they are limited by their adherence to the Intel x86 instruction set). Thirdly, the bandwidth from the processor to the memory, in case of a cache miss, is larger than that of the Intel(-like) processors. Notwithstanding these commonalities between the various RISC processors, there are also differences in instruction latencies, number of instructions processed, etc., which we will address below. We provide block diagrams for each of the processors to give a schematic idea of their structure. However, these figures do not reflect the actual layout of the devices on the respective chips.

Section 1

Test processors

Section 1.1

Compaq Alpha EV7

-

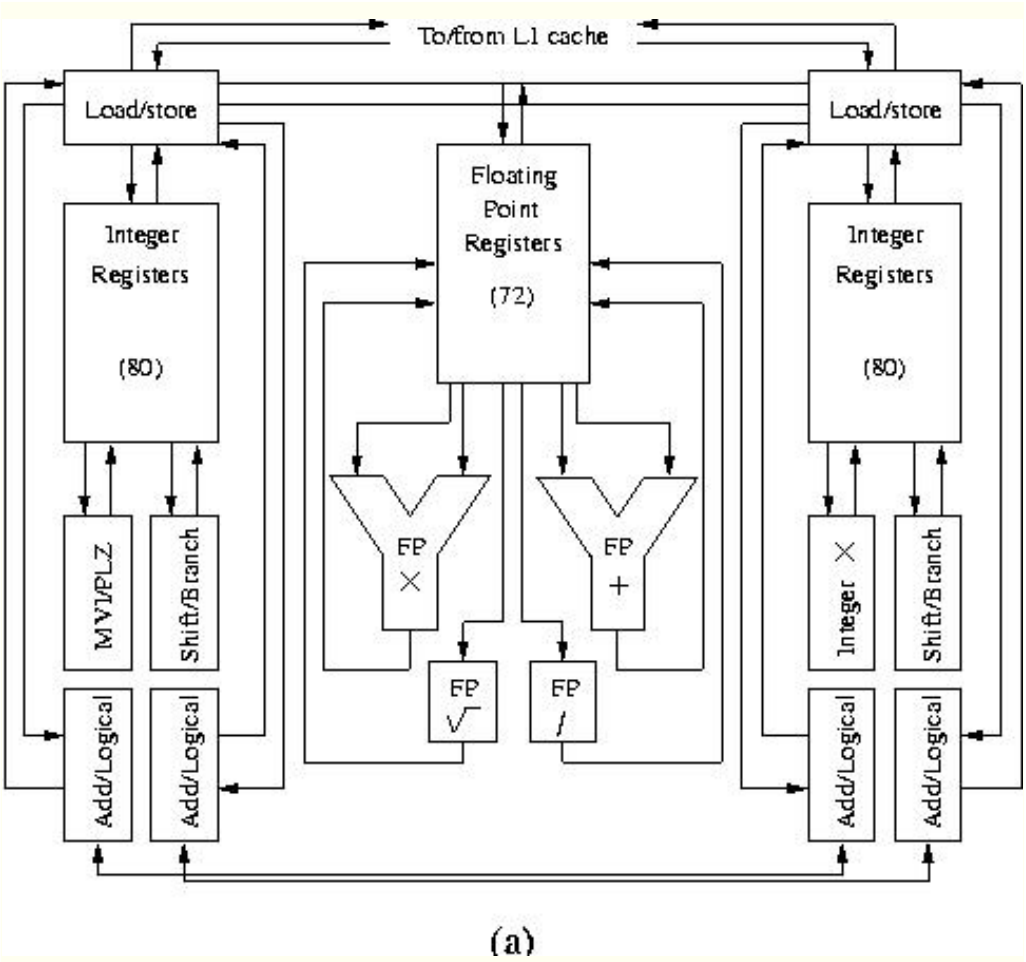

Figure 1 . Block diagram showing the functional units in an Alpha EV7 processor

-

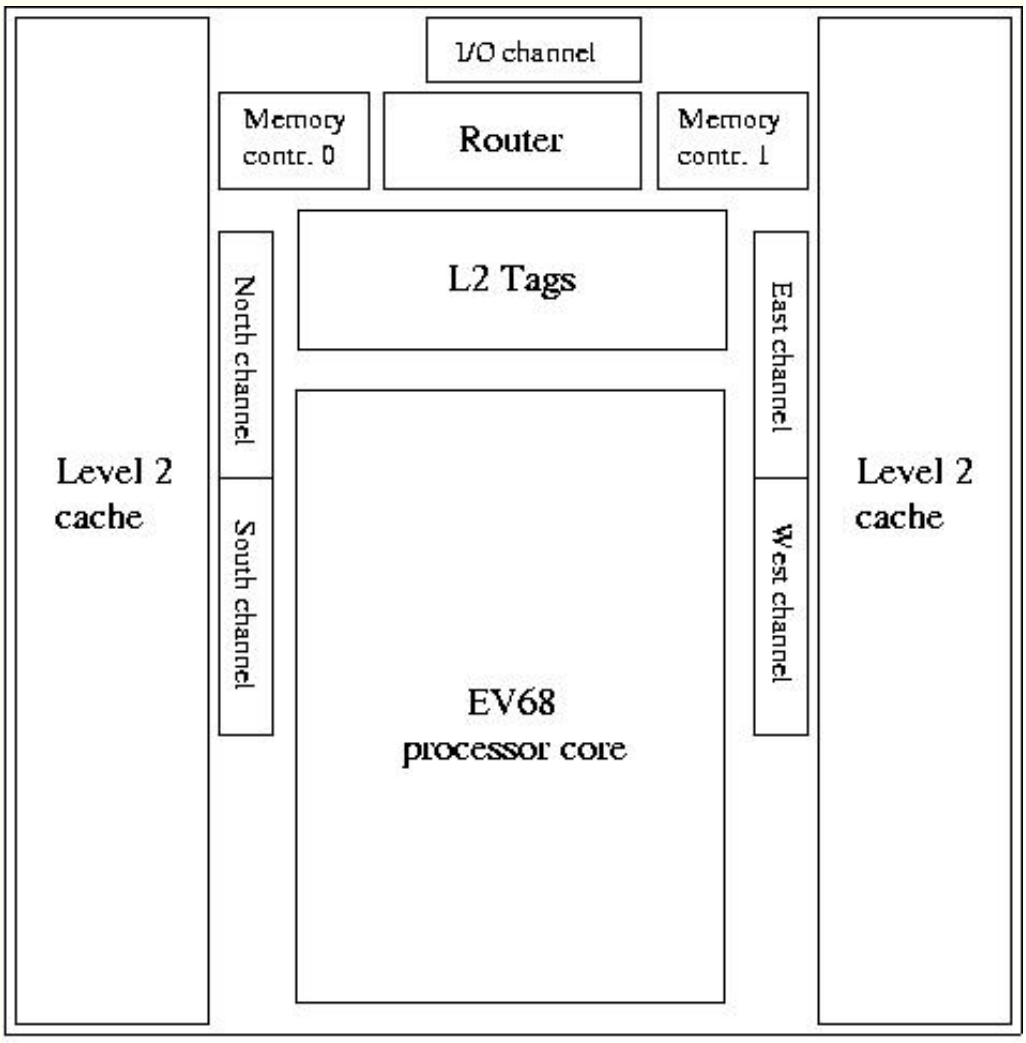

Figure 2 . Chip layout for the Alpha EV7 processor

The present CPU that is employed in Compaq machines like the AlhpaServerSC and the Wildfire and in various cluster systems is the Alpha EV68 processor. Shortly, (second half 2002) EV7 processors will become available. Because of the EV7 structure the macro-architecture of these systems may also significantly change (see below). The core of the EV7 processor is almost identical to that of the EV68 architecture and is depicted in Figure figure 1

A notable fact is that there are ***two*** duplicate integer register files both with 80 entries, that each service a set of integer functional units called cluster 0 and cluster 1, respectively, by Compaq. The four integer Add/Logical units can exchange values in one cycle if required. Although this is not shown in the diagram, the integer multiply is fully pipelined. The two integer clusters and the two floating-point units enable the issueing of up to 6 instructions simultaneously. The two load/store units draw on a 64 KB instruction and a 64 KB data cache that are both 2-way set-associative. Four instructions can be accepted for (speculative) processing. Of the 80 integer and 72 floating-point registers 41 in both register files can hold speculative results. The out-of-order issueing of instructions is supported via an integer queue of length 20 and a floating-point queue with 15 entries. However, as the integer processing clusters do not contain the same functional units, the issueing of integer instructions cannot all be scheduled dynamically. Those instructions that need to execute in a particular unit (e.g., an integer multiply that is only available in cluster 0) are scheduled statically. As

soon as an instruction is issued or is terminated due to mis-speculation it is removed from the queue and can be replaced by another instruction. Instruction fetching is governed by the branch predictor. This hardware contains global and local prediction tables and Branch History Tables (BHTs) to train the predictor in order to obtain an optimal instruction fetch to the instruction cache and registers.

The feature density used is 0.18 µm instead of 0.25 µm which enables the location of a 1.5 MB secondary cache and 2 memory controllers on chip. The largest difference will be that there will be 4 dual channels (North, East, South, West) from the chip to interconnect it with neighouring chips at a bandwidth of 1.6 GB/s per single channel for what Compaq calls "seamless SMP processing" and is, as the name suggests, well-suited to build SMP nodes with low memory latency. The layout of the complete chip is shown in Figure figure 2

Section 1.2

Intel Itanium 2

-

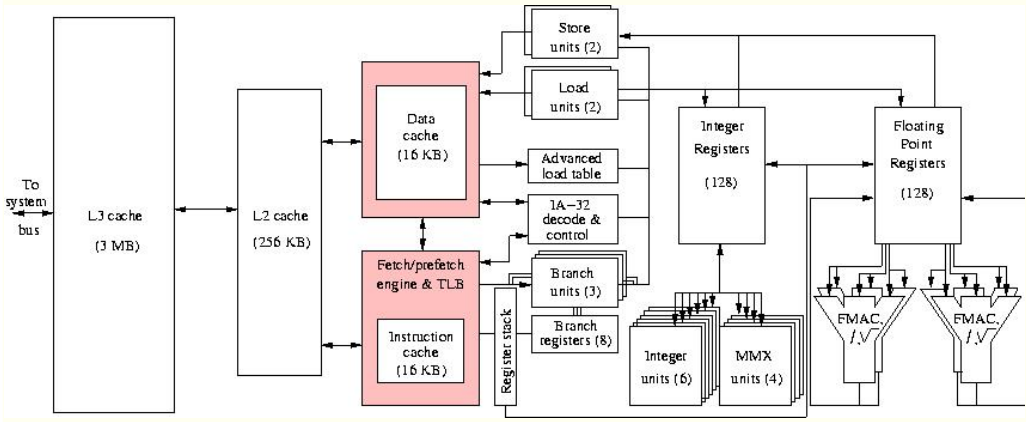

Figure 3 . Block diagram of the Intel Itanium 2

The Itanium 2 is a representative of Intel's IA-64 64-bit processor family and as such the second generation. Its predecessor, the Itanium, has been out for almost a year, but has not spread widely, primarily because the Itanium 2 would follow quickly with projected performance levels up to twice that of the first Itanium. The Itanium 2 will become available in 1--2 month at the time of writing and would improve on some aspects of the first generation, in particular integer processing and cache/memory bandwidth.

The Itanium family of processors has characteristics that are different from the RISC chips presented elsewhere in this section. A block diagram of the Itanium 2 is shown in .

The clock frequency for the Itanium 2 in the products to be shipped will be around 1 GHz. Figure shows a large amount of functional units that must be kept busy. This is done by large instruction words of 128 bits that contain 3 41-bit instructions and a 5-bit template that aids in steering and decoding the instructions. This is an idea that is inherited from the Very Large Instruction Word (VLIW) machines that have been on the market for some time about ten years ago. The two load/store units fetch two instruction words per cycle so six instructions per cycle are dispatched. The Itanium has also in common with these systems that the scheduling of instructions, unlike in RISC processors, is not done dynamically at run time but rather by the compiler. The VLIW-like operation is enhanced with predicated execution which makes it possible to

execute instructions in parallel that normally would have to wait for the result of a branch test. Intel calls this refreshed VLIW mode of operation EPIC, Explicit Parallel Instruction Computing. Furthermore, load instructions can be moved and the loaded variable used before a branch or a store by replacing this piece of code by a test on the place is originally came from to see whether the operations have been valid. To keep track of the advanced loads an Advanced Load Address Table records them. When a check is made about the validness of an operation depending on the advanced load, the ALAT is searched and when no entry is present the operation chain leading to the check is invalidated and the appropriate fix-up code is executed. Note that this is code that is generated at compile time so no control speculation hardware is needed for this kind of speculative execution. This would become exceedingly complex for the many functional units that may be simultaneously in operation at any time.

As can be seen from Figure there are four floating-point units capable of performing Fused Multiply Accumulate (FMAC) operations. However, two of these work at the full 82-bit precision which is the internal standard on Itanium processors, while the other two can only be used for 32-bit precision operations. When working in the customary 64-bit precision the Itanium has a theoretical peak performance of 4 Gflop/s at a clock frequency of 1 GHz. Using 32-bit floating arithmetic, the peak is doubled. In the first generation Itanium there were 4 integer units for integer arithmetic and other integer or character manipulations. Because the integer performance of this processor was modest, 2 integer units have been added to improve this. In addition four MMX units to accommodate instructions for multi-media operations, an inheritance from the Intel Pentium processor family. For compatibility with this Pentium family a special IA-32 decode and control unit is present.

The register files for integers and floating-point numbers is large: 128 each. However, only the first 32 entries of these registers are fixed while entries 33--128 are implemented as a register stack. The primary data and instruction caches are 4-way set associative and rather small: 16 KB each. This is the same as in the former Itanium processor. However, speed of the L1 cache is now doubled to full speed: data and instructions can now be delivered every clock cycle to the registers. Further more the secondary cache has been enlarged from 96 KB to 256 KB and it is 8-way set-associative. Moreover, the L3 cache is moved onto the chip and is no less than 3 MB. This cache structure greatly improves the bandwidth to the processor core, on average by a factor of 3. This does more for the performance improvement than the relatively modest increase in clock speed from 800 MHz to 1 GHz. Also the bandwidth from/to memory has increased by more than a factor of 3. The bus is now 128 bits wide and operates at a clock frequency of 400 MHz, totaling to 6.4 GB/s in comparison to 2.1 GB/s for its predecessor.

The introduction of the first Itanium has been deferred time and again which quenched the interest for use in high-performance systems. With the availability of the Itanium 2 in the second half of 2002 it is expected that the adoption will speed up. Apart from HP/Compaq also SGI, NEC and Fujitsu will include these processors in their systems in the not too distant future while phasing out the Alpha, PA-RISC, MIPS and SUN processors.

Section 1.3

AMD Opteron

-

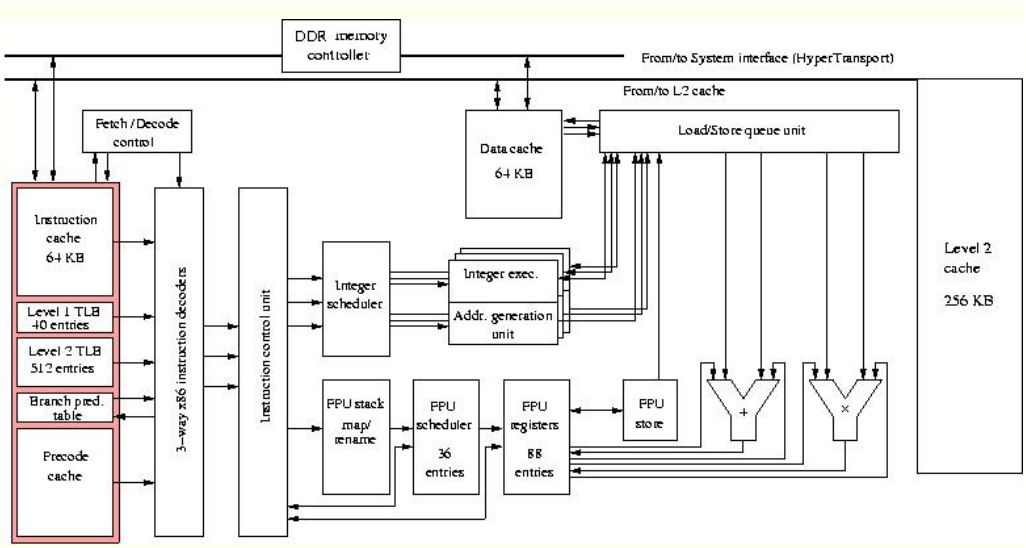

Figure 4 . Block diagram of Opteron processor

The Opteron (long known by its code name Hammer) is the newest processor from AMD and the successor of the Athlon processor. The first versions are expected to become available by the end of 2002. As it is, like the Athlon, a clone with respect to Intel's x86 Instruction Set Architecture, it will undoubtly frequently be used used in clusters. Therefore we discuss this processor here although it is not used presently in integrated parallel systems.

The Opteron processor has many features that are also present in modern RISC processors: it supports out-of-order execution, has multiple floating-point units, and can issue up to 9 instructions simultaneously. In fact, the processor core in very similar to that of the Athlon processor. A block diagram of the processor is shown in Figure

It shows that the processor has three pairs of Integer Execution Units and Address Generation Units that via an 24-entry Integer Scheduler takes care of the integer computations and address calculations. Both the Integer Scheduler and the Floating-Point Scheduler are fed by the 96-entry Instruction Control Unit that receives the decoded instructions from the instruction decoders. An interesting feature of the Opteron is the pre-decoding of x86 instructions in fixed-length macro-operations, called RISC Operations (ROPs), that can be stored in a Pre-decode Cache. This enables a faster and more constant instruction flow to the instruction decoders. Like in RISC processors, there is a Branch Prediction Table assisting in branch prediction.

The floating-point units allow out-of-order execution of instructions via the FPU Stack Map & Rename unit. It receives the floating-point instructions from the Instruction Control Unit and reorders them if necessary before handing them over to the FPU Scheduler. The Floating-Point Register File is 88 elements deep which approaches the number of registers as is available on RISC processors. (For the x86 instructions 16 registers in a flat register file are present instead of the register stack that is usual for Intel architectures.)

The floating-point part of the processor contains three units: a Floating Store unit that stores results to the Load/Store Queue Unit and Floating Add and Multiply units that can work in superscalar mode, resulting in two floating-point results per clock cycle. Because of the compatibility with Intel's Pentium III processors, the floating-point units also are able to execute Intel MMX instructions and AMD's own 3DNow! instructions. However, there is the general problem that such instructions are not accessible from higher level languages, like Fortran 90 or C(++). Both instruction sets are meant for massive processing of visualisation data and only allow for 32-bit precision to be used.

Due to the shrinkage of components the chip now can harbour the secondary cache of 256 KB and the memory controller. This, together with a significantly enhanced memory bus can deliver up to 5.3 GB/s of bandwidth, an enormous improvement over the former memory system. This memory bus, called HyperTransport by AMD, is derived from licensed Compaq technology and similar to that employed in Compaq's EV7 processors (see the). It allows for "glueless" connection of several processors to form multi-processor systems with very low memory latencies.

The clock frequency will be in the order of 2 GHz of the current processors the Opteron is an interesting alternative for many of the RISC processors that are available at this moment. Especially the HyperTransport interconnection possibilities could be highly interesting for building SMP-type clusters.

Section 1.4

Hewlett-Packard PA-RISC 8700

-

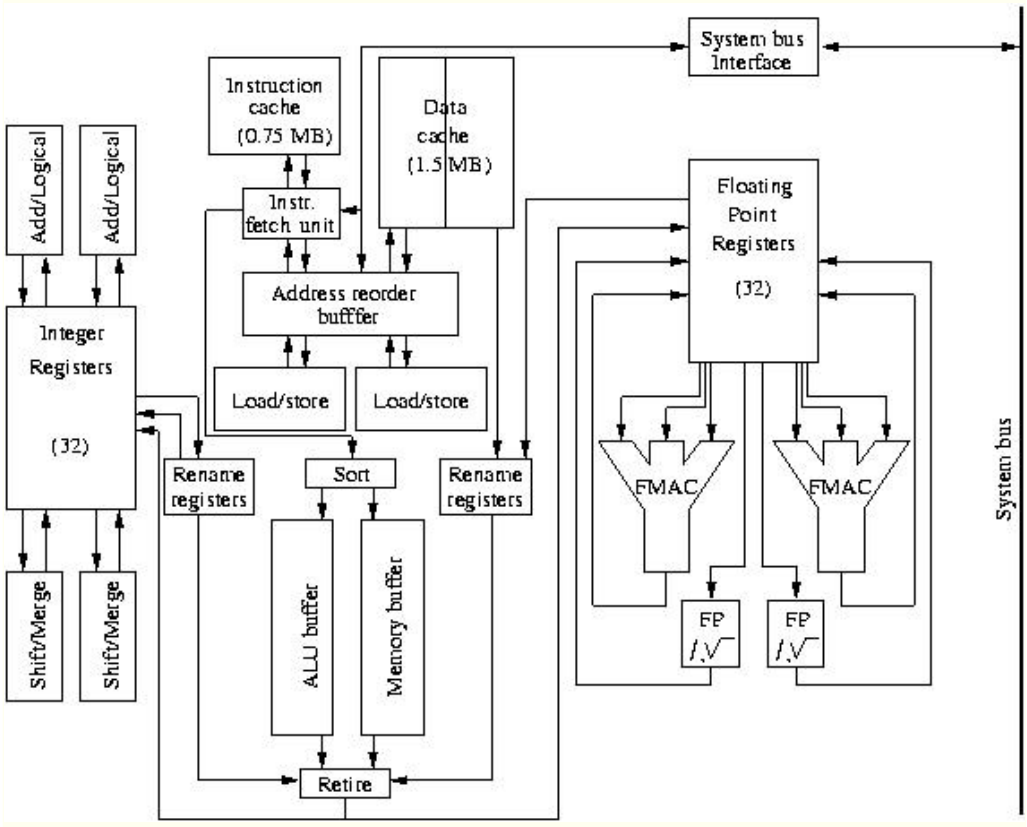

Figure 5 . Block diagram of a HP PA-RISC 8700 processor

The computational power for the Hewlett Packard systems, like the SuperDome, the V-class, and N-class servers is delivered by the PA-8600 and PA-8700 chips. The processor cores of these chips are essentially the same. However, the PA-8700 is made in 0.18 µm logic which made it possible to fit a very large 0.75 MB instruction and a 1.5 MB data cache on the chip and to raise the clock frequency to 750 MHz. A block diagram of the PA-8700 chip is shown in figure 5.

A peculiarity of the PA-8x00 chips is the abcense of a secondary cache. Instead, a very large primary cache is implemented: 0.75 MB instruction cache and 1.5 MB data cache. From the PA8600 on the shrinkingof the logic has allowed to put these caches on-chip. The latency of the caches is two cycles. To ensure data to be shipped to the registers every cycle, the load/store units work "out-of-phase". So, one unit loads from one half of the data cache while the other loads from the other half. The Address Reorder Buffer sets the priority for the loads and tries to load from the alternate halfs every cycle.

Like all advanced RISC processors the PA-8700 has out-of-order execution, the sequence of instructions being determined by the instruction reorder buffer (IRB) which contains an ALU buffer that drives the computational functional units and a memory buffer that controls the load/store units. When speculative branches have been mis-predicted the dependent instructions are retired from the IRB and new candidate instructions replaced them. Branch prediction is controlled through the branch history table (BHT) but, in addition to this dynamic branch prediction, a static branch prediction can be performed at the compiler level or by execution traces of former executions of a program. The BHT was rather small in the predecessors of the PA-8600 and has been enlarged significantly to get better prediction results. Also the Translation Lookaside Buffer (a component of the load/store units not shown in Figure figure 5) has been enlarged for a more effective address translation. Also there is a pre-fetch capability in the new PA-8700 from the data cache.

As can be seen in Figure figure 5, there are 2 floating-point units which each can deliver 2 flops per cycle but only when the operation is in the `axpy` form $x = x + a...y$. This is called a Floating Multiply Accumulate instruction (FMAC) by HP. At a clock frequency of 550 MHz this leads to a theoretical peak performance of 3 Gflop/s. However, when the operations occur in another order or with another composition, 1 flop per cycle per floating-point unit can be executed with a correspondingly lower flop rate.

According to HP's roadmap at least another two generations of the PA-8*x*00 are projected: PA-8800 and PA-8900 that will be on the market concurrently with the IA-64 Itanium 2 (McKinley) and Itanium 3 (Deerfield), respectively. After that the PA-RISC family will be withdrawn to give way to the IA-64 architecture.

Section 1.5

Intel Pentium 4

-

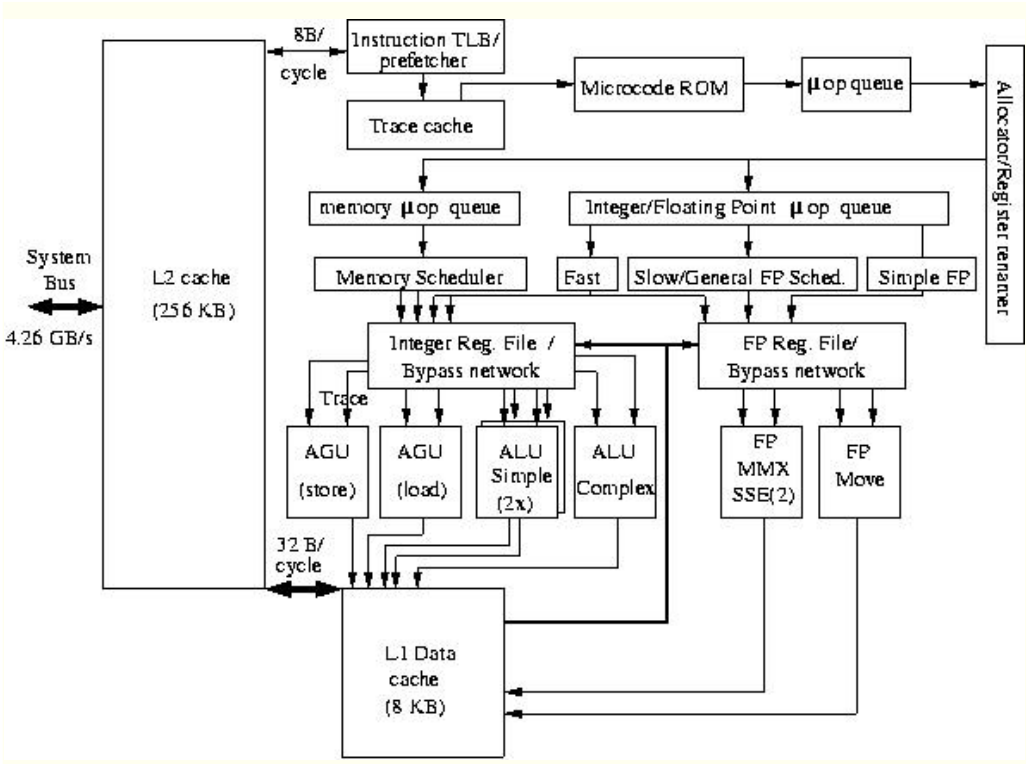

Figure 6 . Block diagram of the Intel Pentium 4

Although Pentium processors are not applied in integrated parallel systems these days, they play a major role in the cluster community as most compute nodes in Beowulf clusters are of this type. Therefore we briefly discuss also this type of processor.

Intel only provides scant information on its processor. Therefore, a rough block diagram of the P4 processor can only be synthesized from various sources. It is shown in Figure figure 6.

There is a number of distinctive features with respect to the earlier Pentium generations. There are two main ways to increase the performance of a processor: by raising the clock frequency and by increasing the number of instructions per cycle (IPC). These two approaches are generally in conflict: when one wants to increase the IPC the chip will become more complicated. This will have a negative impact on the clock frequency because more work has to be done and organised within the same clock cycle. Very seldomly chip designers succeed in raising both clock frequency and IPC simultaneously. Also in the Pentium 4 this could not be done. Intel has chosen for a high clock speed (initially about 40% more than that of the Pentium III with the same fabrication technology) while the IPC decreased by 10--20%. This still gives a net performance gain even if other changes would have been made to the processor. To sustain the very high clock rate that the present processors have, currently > 2 GHz, a very deep instruction pipeline is required. The instruction pipeline has no less than 20 stages, double the number of stages in that of the Pentium III. Although this favours a high clock rate, the penalty for a pipeline miss (e.g., a branch mis-predict) is much heavier and therefore Intel has improved the branch prediction by a increasing the size of the Branch Target Buffer from 0.5 to 4 KB. In addition, the Pentium 4 has an execution trace cache which holds partly decoded instructions of former execution traces that can be drawn upon, thus foregoing the instruction decode phase that might produce holes in the instruction pipeline. The allocator dispatches the decoded instructions, "micro operations", to the appropriate µop queue, one for memory operations, another for integer and floating-point operations.

Two integer Arithmetic/Logical Units are kept simple in order to be able to run them at

twice the clock speed. In addition there is an ALU for complex integer operations that cannot be executed within one cycle. There is only one Floating-point functional unit that delivers one result per cycle. However, besides the normal Floating-point Unit, there also are additional units that execute the Streaming SIMD Extensions 2 (SSE2) repertoire of instructions, a 144-member instruction set, that is especially meant for multimedia, and 3-D visualisation applications. The length of the operands for these units is 128 bits. The Intel compilers have the ability to address the SSE2 units. This makes it in principle possible to achieve a two times higher floating-point performance.

The primary cache is quite small by today's standards: 8 KB. This is again to accommodate the high clock speed. With this size of cache it is possible to have a latency of two cycles for the cache, where it was 3 cycles in the Pentium III. The secondary cache has a size of 256 KB and has a wide 256-bit bus, which amounts to a bandwidth of 54.4 Gb/s. Also the memory bandwidth has improved significantly over that of the Pentium III: although the bus cycle frequency is 133 MHz, four transactions per cycle can be done, making it effectively a 533 MHz bus. This should give quite an improvement for codes that cannot be kept in cache.
It will depend heavily on the availability of compilers that are able to take advantage of all the facilities present in the P4 processor. But if they can, the processor could form a good basis for any HPC platform.

Section 1.6

IBM POWER4

-

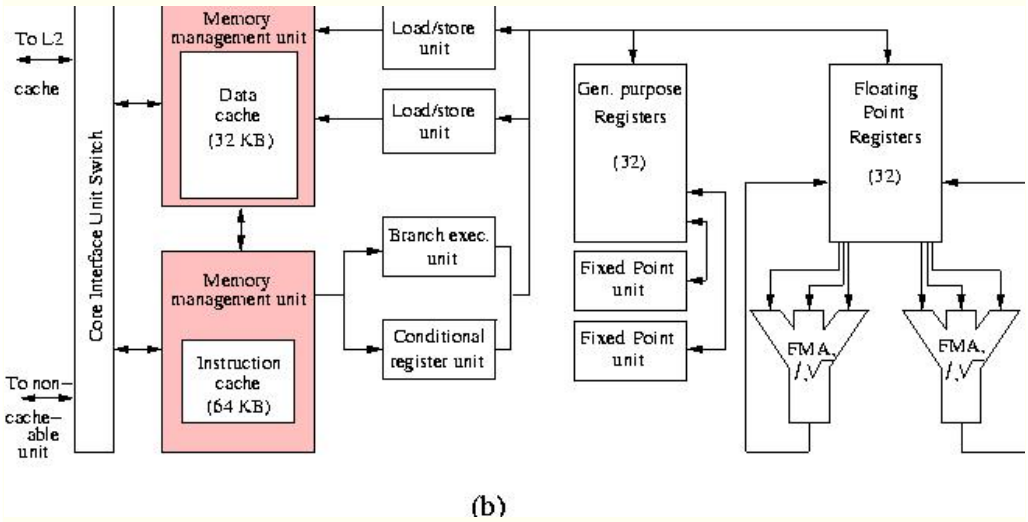

Figure 7 . Block diagram of the POWER4 processor core

-

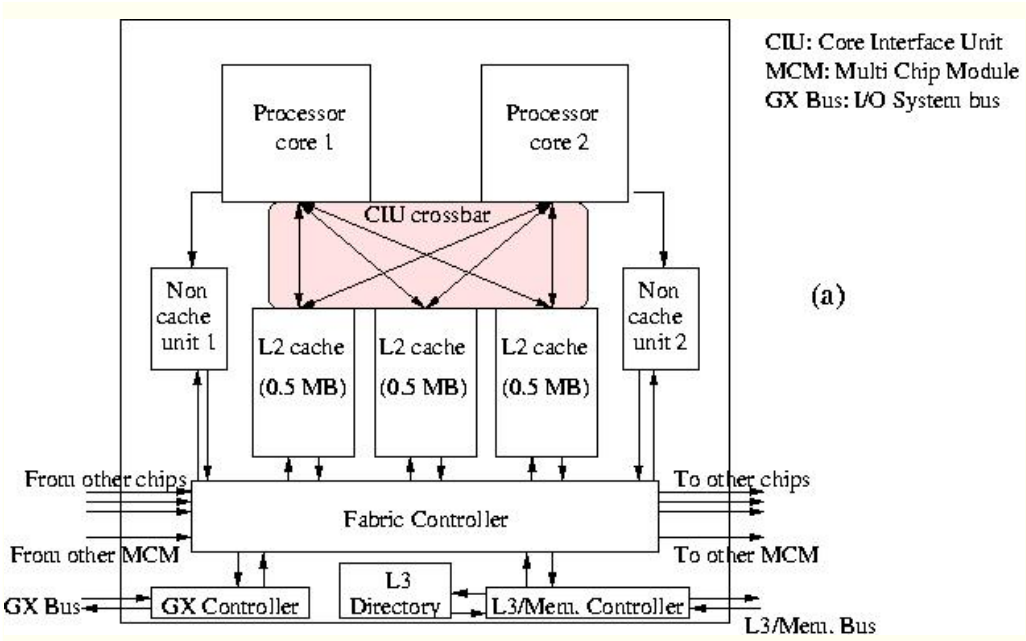

Figure 8 . Diagram of the IBM POWER4 chip layout

In the newest IBM SP systems the nodes contain the POWER4 chip, the latest variant of the RS/6000 family of processors. At the time of writing, the clock frequency of the POWER4 is 1.3 GHz. The chip size has become so large (or rather the feature size has become so small) that IBM places now two processor cores on one chip as shown in Figure figure 8. The chip also harbours 1.5 MB of secondary cache divided over three modules of 0.5 MB each.

The L2 cache module are connected to the processors by the Core Interface Unit (CIU) switch, a 2 x 3 crossbar with a bandwidth of 40 B/cycle per port. This enables to ship 32 B to either the L1 instruction cache or the data cache of each of the processors and to store 8 B values at the same time. Also, for each processor there is a Non-cacheable Unit that interfaces with the Fabric Controller and that takes care of non-cacheable operations. The Fabric Controller is responsible for the communication with three other chips that are embedded in the same Multi Chip Module (MCM), to L3 cache, and to other MCMs. The bandwidths at 1.3 GHz are 10.4, 6.9, and 5.2 GB/s, respectively. The chip further still contains a variety of devices: the L3 cache directory and the L3 and Memory Controller that should bring down the off-chip latency considerably, the GX Controller that responsible for the traffic on the GX bus. This bus transports data to/from the system and in practice is used for I/O. Some of the integrated devices, like the Performance Monitor, and logic for error detection and logging are not shown in Figure figure 8.

A block diagram of the processor core is shown in Figure figure 7.

In many ways the POWER4 processor core is similar to the former POWER3 processor: there are 2 integer functional units instead of 3 (called Fixed Point Units by IBM) and instead of a fused Branch/Dispatch Unit, the POWER4 core has a separate Branch and Conditional Register Unit, 8 execution units in all. Oddly, the instruction cache is two times larger than the data cache (64 KB direct-mapped vs. 32 KB two-way set associative, respectively) and all execution units have instruction queues associated with them that enables the out-of-order processing of up to 200 instructions in various stages. Having so may instructions simultaneously in flight calls for very sophisticated branch prediction facilities. Instructions are fetched from the Instruction Cache under control of the Instruction Fetch Address Register which in turn is influenced by the branch predict logic. This consists of a local and a global Branch History Table (BHT), each with 16 K entries and a so-called selector table which keeps track of which of the BHTs has functioned best in a particular case in order to

select the prediction priority of the BHTs for similar cases coming up.

Unlike in the POWER3, the fixed point units performs integer arithmetic operations that can complete in one cycle as well as multi-cycle operations like integer multiply and divide. There are no separate floating-point units for operations that require many cycles like divisions and square roots. All floating-point operations are taken care of in the FP units and, like in the HP PA-8700, there is an instruction to accommodate the `axpy` operation, called Fused Multiply Add (FMA) at IBM's which could deliver 2 floating-point results every cycle. This brings the theoretical peak performance at 1.3 Gflop/s at the current clock frequency. Like in the HP processor, the composition of the floating-point operations should be such that the units have indeed enough FMAs to perform otherwise the performance drops by a factor of 2.

Although here the dual core version of the chip is described that is positioned for general processing, also a single core version is marketed that is recommended for HPC use. The reason is that in this case the bandwidth from the L2 cache does not have to be shared between the CPUs and a contention-free transfer of up to 83.2 GB/s can be achieved while in the dual core version a peak bandwidth of 124.8 GB/s is to be shared between both CPUs.

It is interesting to see that presently three vendors (AMD, Compaq, and IBM) have facilities that enable glueless coupling of processors although the packaging and implementation is somewhat different. All implementations allow for low-latency SMP nodes with a considerable number of processors stimulating the trend to build parallel systems based on SMP nodes.

Section 1.7

MIPS R14000A

-

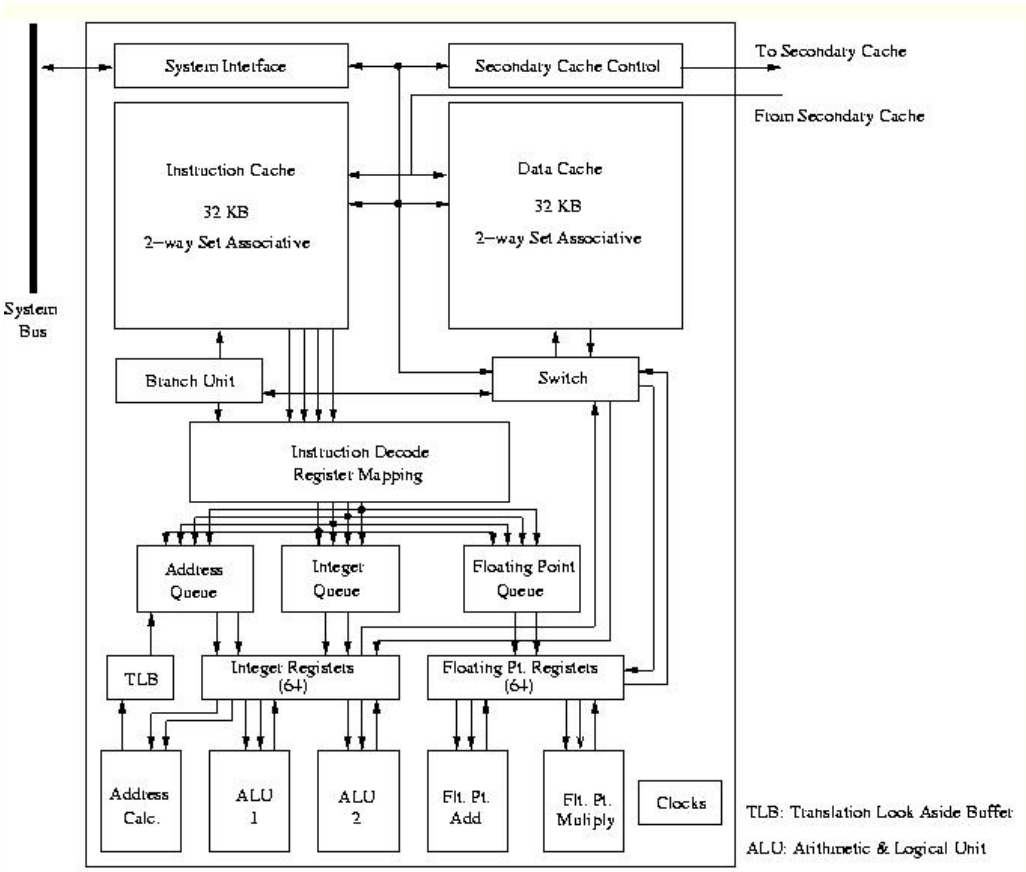

Figure 9 . Block diagram of the MIPS R14000 processor

The essentials of the MIPS R1x000 series of processors have not changed since the introduction of the first in this family, the R10000. The current processor that is at the heart of the SGI Origin3000 series is the R14000A. The R14000A is similar to the preceding R14000 except for the clock cycle: this is presently 600 MHz and as such the lowest of all RISC processors employed in High Performance systems. A block diagram of this processor is given in Figure .

The R14000 is a typical representative of the modern RISC processors that are capable of out-of-order and speculative instruction execution. Like in the Compaq Alpha processor there are two independent floating-point units for addition and multiplication and, additionally, two units that perform floating division and square root operations (not shown in Figure). The latter, however, are not pipelined and with latencies of about 20--30 cycles are relatively slow. In all there are 5 pipelined functional units to be fed: an address calculation unit which is responsible for address calculations and loading/storing of data and instructions, two ALU units for general integer computation and the floating-point add and multiply pipes already mentioned.

The level 1 instruction and data caches have a moderate size of 32 KB and are 2-way set-associative. In contrast, the secondary cache can be very large: up to 16 MB. Both the integer and the floating-point registers have a physical size of 64 entries, however, 32 of them are accessible by software while the other half is under direct CPU control for register re-mapping.

The clock frequency of the MIPS R1x000 processors have always been on the low side. The first R10000 appeared at a frequency of the 180 MHz while in the new

R14000A the clock cycle is 600 MHz and will slightly rise during its lifetime. With the initial 600 MHz frequency the theoretical peak performance is 1.2 Gflop/s. Because of the independent floating-point units without fused multiply-add capabilities often a fair fraction of that speed can be realised. There also have been made some improvements with respect to the earlier chips: the bus speed has been doubled from 100 MB/s to 200 MB/s and the L1 cache that ran at a 2/3 speed in the predecessor R12000 has been sped up to full speed in the R14000A.

The R14000A is built in advanced 0.13 µm technology and it has at the present 600 MHz clock frequency an extremely low power consumption: only 17 Watt, several factors lower than that of the other processors discussed here. SGI keeps the clock frequency intentionally as low as possible to enable to build "dense" systems that can accommodate a large amount of processors in a small volume.

A R16000 successor is planned for next year that will be a shrunken version of the R14000 made in 0.11 µm technology and with a clock frequency of 700 MHz. In the current plans it seems that SGI will stay with the MIPS processors (along with systems with Itanium processors like most vendors). A R18000 will in all probability become available in 2004 both as dual and single core chips while even a R20000 is envisioned around 2005 that would double the amount of floating-point units to four per processor core.

Section 1.8

Sun UltraSPARC III

-

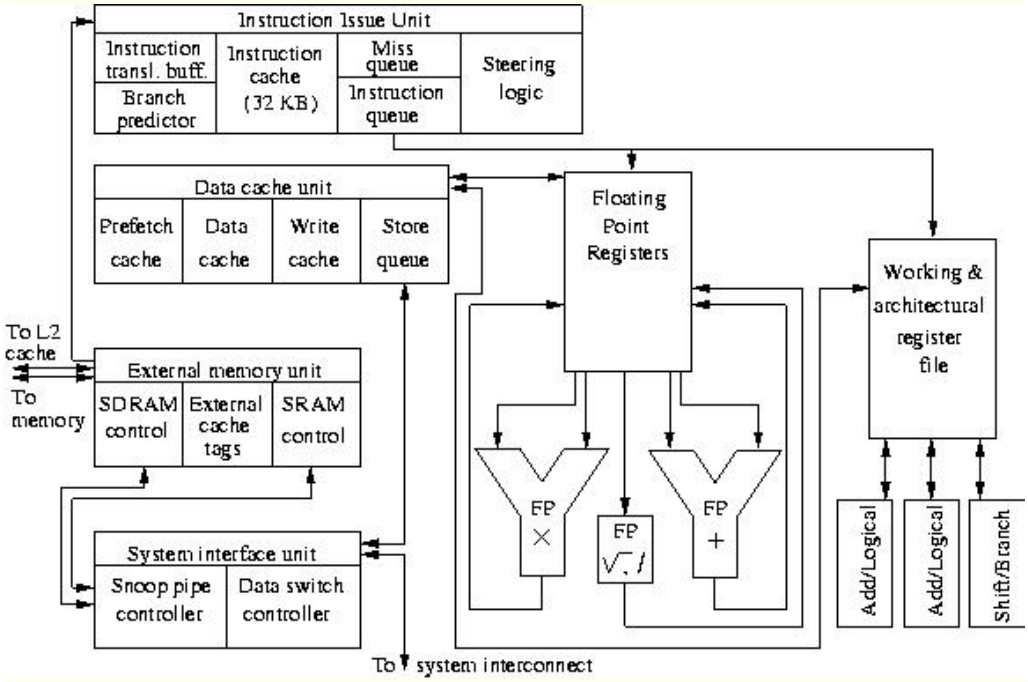

Figure 10 . Block diagram of the UltraSPARC III processor

The UltraSPARC-III is the third generation from the UltraSPARC family and, as one of the last RISC processor families, with full 64-bit precision and addressing range. It is built in 0.18 µm CMOS technology at a clock frequency that is currently 900 MHz. It is a complete revamp of earlier UltraSPARC designs but backward compatible with these older processors. UltraSPARCs are used in all SUN products from workstations

to the heavy E10000 servers and also in Fujitsu products like the AP-3000. We show a block diagram of the UltraSPARC-III in Figure .

The chip is characterised by large large amount of caches of various sorts as can be seen in the figure. The Data Cache Unit (DCU) contains apart from a 4-way set associative cache of 64 KB also a write and a pre-fetch cache, both of 2 KB. The pre-fetch cache is independent from the data cache and can load data when this is deemed appropriate. The write cache defers writes to the L2 cache and so may evade unnecessary writes of individual bytes until entire cache lines have to be updated. The Instruction Issue Unit (IIU) contains the 32 KB 4-way set associative instruction cache together with the instruction TLB which is called Instruction translation buffer in SUN's terminology. The IIU also contains a so-called miss queue that holds instructions that are immediately available for the execute units when a branch has been mis-predicted. Branch prediction is fully static in the UltraSPARC-III. It is implemented as a 16 KB table in the IIU that is pipelined because of its size.

The Integer Execute Unit (IEU) has two Add/Logical Units and a branch unit. Integer adds and multiplies are pipelined but the divide operation is not. It is performed by an Arithmetic Special Unit (not shown in the figure) that does not burden the pipelines for the ALUs. The integer register file is effectively divided in two and is called the Working and Architectural Register File by SUN. Operands are accessed and results stored in the working registers. When an exception occurs, the results to be undone in the working registers are overwritten by those from the architectural file.

The floating-point unit (FPU) has two independent pipelined units for addition and multiplication and a non-pipelined unit for floating division and square-root computation that require in the order of 20--25 cycles. The FPU also contains graphics hardware (not shown in Figure) that shares the pipelined adder and multiplier with general 64-bit calculations. For the chips delivered at 900 MHz, the theoretical peak performance is 1.8 Gflop/s. It is expected that the UltraSPARC-III technology can be shrunk to reach a clock frequency to 1 GHz by the end of its life cycle.

The memory controller and the L2 cache controller together with the L2 cache tags are all housed on the chip in the External Memory Unit. This shortens the latency of accesses from both memory levels. In addition, both controllers communicate with the System Interface Unit (SIU), also on-chip to keep in touch with the snoop pipe controller in the SIU. The processor has been built with multi-processing in mind and the snoop controller keeps track of data requests in the whole system to ensure coherency of the caches when required.

As the UltraSPARC-III is around for about a year at the time of writing and the clock frequency has gone up in that period from 750 to 900 MHz. The next generation will take some time (about a year) to appear and, after the radical redesign in the present generation, will have most of the same characteristics as the current one.

Chapter 3
Recount of the (almost) available systems

In this section we give a recount of all types of systems as discussed in the former section and that are marketed presently or will appear within 6 months from now. When vendors market more than one type of machine we will discuss them in distinct subsections. So, for instance, we will discuss Fujitsu systems under entries, VPP5000 and PRIMEPOWER because they have a very different structure.

At the time of writing this report the takeover of Compaq by HP became a fact. But at present it not clear what the impact on the product lines of the two branches would be. Therefore the systems of Compaq and HP will in this issue still be discussed separately and with their old company names.

Section 1

Test

Section 1.1

The Fujitsu AP3000

machine-type: RISC-based distributed-memory multi-processor
operating-system: Cell OS (transparent to the user) and Solaris (Sun's Unix variant)
connection-structure: 2-D torus
compilers: Parallel Fortran/AP, Fortran 90, HPF, C, C++.
vendor web-site:
year-of-introduction: 1996

Model	AP3000
Clock cycle	300 MHz
Processor performance	600 Mflop/s
Peak performance	614 Gflop/s
Maximum system memory	2 Tbyte
Number of processors	4 - 1024

Section 1.1.1

Remarks

The AP3000 is the sucessor of the earlier AP1000 system. Although the name could suggest otherwise, few characteristics of the AP1000 have been retained except that Sun Sparc processors are used in the nodes. No front-end processor is required anymore as in the former system.

Also the communication network has been simplified considerably with respect to that in the earlier model: where three different networks were present in the AP1000 (see Hori91), in the AP3000 the nodes are connected in a 2-D torus structure with a bi-directional bandwidth of 200 MB/s. The maximum amount of memory is large: a full 1024 node system can accomodate 2 TB.

Another difference with the AP1000 system is that the fastest nodes (the (U300 nodes described here) can have either 1 or 2 CPUs as opposed to only one CPU in the AP1000. The two CPUs share the on-board memory.

The available software for the AP3000 is extensive: Parallel Fortran/AP is a Fortran 77 with extensions that offers a shared memory-like programming model for the system. In addition, HPF is available and the machine can also be used with a message passing model as customised MPI/AP and PVM/AP are offered. As sequential languages to be used with the message passing libraries Fortran 90, C and C++ are available.

The current motto on Fujitsu's English home page reads "The possibilities are infinite". This certainly is true for looking for relevant information on this system: when following the links for these machines one ends up on unreadable Japanese pages from which it is difficult to find your way back.

Section 1.1.2

Measured Performances

: The system has been announced in March 1996 and installations have been done in Japan, the University of Singapore and at the Australian National University but as yet no performance figures are published. Although the theoretical bandwidth is 200 MB/s, the best measured bandwidth with MPI as given by Fujitsu is 88 MB/s with a latency of 12 µs.

Section 1.1.3

TOP500 Systems

Section 1.2

EnterTheGrid description of company

Section 1.3

The C-DAC Param 10000 OpenFrame

machine-type: RISC-based distributed memory multi-processor.
operating-system: SunOS, Sun's Unix flavour
connection-structure: Variable (see remarks)
compilers: Fortran 77/90, C, C++
vendor web-site: http://www.cdacindia.com/html/openframe.htm
year-of-introduction: 2000

Model	C-DAC Param 10000 OpenFrame
Clock cycle	400 MHz
Processor performance	800 Mflop/s
Peak performance	- Mflop/s
Maximum system memory	1 Gbyte
Number of processors	-

Section 1.3.1

Remarks

The PARAM systems are highly variable machines to such an extent that they almost can be regarded as clusters. However, CDAC has developed its own communication network and optimised MPI which gives it the flavour of an integrated parallel machine. The maximum number of processors is unclear although the information on the vendor's web page suggests that systems with a peak performance of up to a Tflop/s could be delivered. As the basic processor presently is the UltraSPARC II processor with a clock frequency of 400 MHz, this would amount to systems with more than 1200 processors. Such systems could communicate through CDAC's own PARAMNet at a peak bandwidth of 50 MB/s bi-directional, Myrinet at 160 MB/s, ATM at 155/622 Mb/s, or Fast Ethernet. Unfortunately, there is no firm information about the structure of PARAMNet. These different possibilities stress the cluster character of the PARAM systems or, as CDAC expresses it, the OpenFrame policy for its systems.
CDAC is not new in the High Performance Computing business which shows in the software that is available for the PARAM machines. Apart from CDAC's MPI, KSHIPRA a lightweight, low latency communication layer based on Berkeley's Active Messages-II, performance profilers and a parallel debugger are offered along with a complete compiler set with Fortran 77/90, C, and C++.

The PARAM machines have been mostly sold on the internal Indian market where more than 20 systems have been installed, mostly with 8 processors. However, since July 2000 a system is placed at the Russian National Academy of Sciences in Moscow in a collaboration project between India and Russia to develop parallel applications in the area of structural analysis and Computational Fluid Dynamics.

Section 1.3.2

Measured Performances

No performance measurements of PARAM 10000 systems are available at all, although one would presume that they will not be very different from other UltraSPARC II-based systems using MPI for parallelisation.

Section 1.3.3

TOP500 Systems

Section 1.4

EnterTheGrid description of company

Section 1.5

The NEC Cenju-4

machine-type: RISC-based distributed-memory multi-processor.
operating-system: Cenjuiox (Mach micro-kernel based Unix flavour).
connection-structure: Multi-stage crossbar.
compilers: Fortran 77, Fortran 90, HPF (subset), ANSI C.
vendor web-site: http://kiefer.gmd.de:8002/popcorn/services/Overview.html
year-of-introduction: 1998

Model	Cenju-4
Clock cycle	5 ns
Processor performance	400 Mflop/s
Peak performance	410 Gflop/s
Maximum system memory	512 Gbyte
Number of processors	8 - 1024

Section 1.5.1

Remarks

The name Cenju-4 suggests that there have been predecessors, Cenju-1, Cenju-2, and Cenju-3. This is indeed the case but the first two systems have only been used internally by NEC for research purposes and were never officially marketed. The Cenju-3 was also placed externally but, again, mostly for evaluation purposes. The same is the case for the present Cenju-4: it is not actively marketed, although NEC will have no objections to selling it. Officially, the Cenju-series is regarded by NEC as systems to gain experience in massively parallel computing and to develop the proper tools for it.

The Cenju-4 is based on the MIPS R10000 RISC processor. All processors have, apart from their on-chip 32 KB primary data and instruction cache, a secondary cache of 1 MB to mitigate the problems that arise in the high data usage of the CPU.

The interconnection type used in the Cenju is a multistage crossbar build from 4×4

modules that are pipelined. So, in a full configuration the maximal number of levels in the crossbar to be traversed is six. The peak transfer rate of the crossbar is quoted as 200 MB/s irrespective of the data placement. Preliminary measurements of the author of this report show that the practical transfer rate for point-to-point communication is at least 175 MB/s with MPI; a quite high efficiency.

The system needs a front-end processor of the EWS4800 type (functionally equivalent to Silicon Graphics workstations) of SUN. The I/O requirements have to be fulfilled by the front-end system as the Cenju does not have local (distributed) I/O capabilities.

There is some software support that should make the programmer's life somewhat easier. The library PARALIB/CJ contains proprietary functions for forking processes, barrier synchronisation, remote procedure calls, and block transfer of data. Like on the t3e.html#t3e, the Hitachi sr8000.html#sr8000, and on the former gone the programmer has the possibility to write/read directly to/from non-local memories which avoids much message passing overhead.

Section 1.5.2

Measured Performances

: No systematic performance measurement have been done yet on the Cenju-4. However, from comparative studies it seems that the speed on some applications is presently about 2/3 of an equivalent SGI R10000 node due to a different compiler technology (CaSt98). Nagel reports a speed of 90-100 Mflop/s for in-cache matrix-matrix multiplication in Fortran 90 per node (Nagel98).

Section 1.5.3

TOP500 Systems

Section 1.6

EnterTheGrid description of company

Aspen Systems

Aspen is a computer hardware designer and manufacturer. Aspen ia also active in the Alpha market with the Alpine line of Alpha-based workstations and servers. Aspen has branched its expertise to include a variety of high-performance technical computing disciplines with emphasis on serving the needs of advanced technical computing users and corporations.

http://www.aspsys.com

Section 1.7

The Compaq AlphaServer SC

machine-type: RISC-based SMP-clustered DM-MIMD system.
operating-system: Tru64 Unix (Compaq's flavour of Unix)
connection-structure: Fat Tree
compilers: Fortran 77, HPF, C, C++
vendor web-site: http://www.compaq.com/hpc/systems/sys_sc.html
year-of-introduction: 1999

Model	AlphaServer SC
Clock cycle	1 GHz
Processor performance	2 Gflop/s
Peak performance	8 Tflop/s
Maximum system memory	8 Tbyte
Number of processors	1 - 4096

Section 1.7.1

Remarks

The AlphaServer SC is the very high end of Compaq's AlphaServer line (SC stands for SuperComputer). The system is typical for the present development of SMP-based clustered systems. In the SC system the basic SMP node is the Compaq ES45, a 4-CPU SMP system with the Alpha 21264a (EV68) as its processor. The clock rate is 1 ns. The SMP node has a crossbar as its internal network with an aggregate bandwidth of 5.2 GB/s (1.33 GB/s/processor). This is sufficient to deliver 1.33 byte/clock cycle to each processor in the node simultaneously.

Within a node the system is a shared memory machine that allows for shared-memory parallel processing, for instance by using `OpenMP`. When more than four processors are required, one has to use a message passing programming model like MPI, PVM, or HPF (Compaq is one of the few companies that still provides its own HPF compiler).

For communication between the SMP nodes the SC uses QsNet, a network manufactured by QSW Limited. In fact QsNet is the follow-on of the network employed in the former Meiko CS-2 systems (see section gone). The network has the structure of a fat tree, is based on PCI technology, and has a point-to-point bandwidth of 210 MB/s. Because of its fat tree structure the bandwidth in the upper level of the network is 340 MB/s sustained. QSW claims a very low latency of 5 µs for MPI messages.

Section 1.7.2

Measured Performances

In Dong02 a performance of 4463 Gflop/s processors was reported solving a full linear system of order 280,000 on a configuration with 4463 processors, an efficiency of 73.8%. For a small system of order 1000 an efficiency of about 50% was measured in using the EuroBen Benchmark (see EurB99).

Section 1.7.3

TOP500 Systems

3	AlphaServer SC ES45/1 GHz	http://www.psc.edu/
4	AlphaServer SC ES45/1 GHz	http://www.cea.fr/
6	AlphaServer SC ES45/1 GHz	http://www.lanl.gov/
37	AlphaServer SC ES45/1 GHz	http://www.wes.hpc.mil/
38	AlphaServer SC ES45/1 GHz	http://www.gsfc.nasa.gov/
39	AlphaServer SC ES45/1 GHz	http://www.asc.hpc.mil
42	AlphaServer SC ES45/1 GHz	http://www.apac.edu.au/
83	AlphaServer SC ES45/1 GHz	http://www.asc.hpc.mil
84	AlphaServer SC ES40/EV67	http://www.compaq.com
85	AlphaServer SC ES40/EV67	http://www.llnl.gov/
131	AlphaServer SC ES40/833 MHz	http://www.wes.hpc.mil/
132	AlphaServer SC ES40/833 MHz	http://www.jamstec.go.jp/
133	AlphaServer SC ES40/833 MHz	http://www.lanl.gov/
140	AlphaServer SC ES40/833 MHz	http://www.cea.fr/
175	AlphaServer SC ES45/1 GHz	http://www.ntu.edu.sg/sce/centres/bioinformatics/
181	AlphaServer SC ES40/EV67	http://www.csm.ornl.gov/ccs/
182	AlphaServer SC ES40/EV67	http://www.psc.edu/
254	AlphaServer SC ES40/EV67	http://www.cea.fr/
262	AlphaServer SC ES45/1 GHz	http://www.cineca.it
374	AlphaServer SC ES40/833 MHz	http://www.cea.fr/
429	AlphaServer SC ES40/833 MHz	http://www.vpac.org/

Section 1.8

EnterTheGrid description of company

HP - Compaq HPTC

Compaq produces the whole range of technical and scientific computers, including clusters and high-end supercomputers. According to IDC, Compaq is the largest company in terms of high performance server revenue. The company is represented with several entries in the TOP500. It is building the next-generation ASCI machine: the "Q". After the recent merger, Compaq is now a part of Hewlett Packard (HP).

AlphaServer SC supercomputers scale to support hundreds of processors (16 to 512). These systems use standard AlphaServer products in configurations with highly scalable, very high performance switched interconnects based on Quadrics Elan technology. The integrated, multinode systems provide many single-system management (SSM) features that allow users, programmers, and administrators to view them as single, unified system. Costs are kept to a minimum by using standard, high-volume components allowing Compaq to deliver superior price/performance for distributed, scalable supercomputing applications.

1. The U.S. Department of Energy's National Nuclear Security Administration (NNSA) selected Compaq to participate in the Accelerated Strategic Computing Initiative (ASCI) and to build a 30+ TeraOPS, 12,000-processor AlphaServer system,

codenamed "Q", to provide an integrated programme of surveillance, experiments, non-nuclear tests, archived data, modelling and simulation to assess and certify the safety, security and reliability of nuclear weapons without underground testing.
2. The Pittsburgh Supercomputing Center (PSC) has installed a 2,768-processor, 6 TeraOPS Compaq AlphaServer SC system entered service in autumn 2001. The National Science Foundation (NSF) funded system will conduct scientific research in areas such as the protein structure and dynamics for drug discovery, storm-scale weather forecasting, climate change modelling and earthquake simulation.
3. The French Atomic Energy Commission has a Compaq 5 TeraOPS AlphaServer SC system to simulate nuclear weapons testing.
4. The Japan Atomic Energy Research Institute at the Kansai Research Establishment (JAERI-KANSAI) has ordered a 1.5 TeraOPS AlphaServer SC system with 908 processors. The institute's Advanced Photon Research Center will use the system for core research activities such as X-ray microscopy, ultra-precision machining and medical diagnosis and treatment.
5. The Australian Partnership for Advanced Computing (APAC) was delivered an AlphaServer SC system containing more than 450 processors for use by researchers to conduct innovative large-scale scientific and engineering research in chemistry, physics, environmental science and biotechnology. Australia's Victorian Partnership for Advanced Computing (VPAC) agency also selected Compaq to build a 128-processor AlphaServer SC system for research in areas such as molecular modelling for new drugs and pattern discovery for fraud protection.

In June 2001, Compaq announced a strategic partnership with Intel Corporation, designed to enhance Compaq's long-term HPTC product roadmap. Beginning in 2004, Compaq will implement its 64-bit enterprise servers, including the SC family, using the Intel Itanium microprocessor architecture.

http://www.compaq.com/hpc/

Section 1.9

The Compaq GS series

machine-type: RISC-based SMP system.
operating-system: Tru64 Unix (Compaq's flavour of Unix).
connection-structure: Variable (see remarks)
compilers: Fortran 77, Fortran 90, HPF, C, C++.
vendor **web-site:**
http://www.digital.com/products/quickspecs/10643_na/10643_na.html
year-of-introduction: 1999

Model	GS80	GS160	GS320
Clock cycle	1 GHz	1 GHz	1 GHz
Processor performance	2 Gflop/s	2 Gflop/s	2 Gflop/s
Peak performance	16 Gflop/s	32 Gflop/s	64 Gflop/s
Maximum system memory	64 Gbyte	128 Gbyte	156 Gbyte
Number of processors	- 8	- 16	- 32

Section 1.9.1

Remarks

The GS series is a family of SMP servers with currently the fastest Alpha 21264 processor available at 1 GHz. The systems are build from ``Quad Building Blocks'' (QBBs), blocks of 4 processors. The GS80 can house 2 of these blocks, while the largest configuration, the GS320 has up to 32 processsors in 8 QBBs. The processors in a QBB have access to the memory via a crossbar with an aggregate bandwidth of 7.0 GB/s. This means that for each individual processor the bandwidth is 1.75 GB/s or slightly more than a quarter of an 8-byte operand per cycle. The QBBs are again connected by a crossbar with the same bandwidth which amounts to an aggregate bandwidth of 57 GB/s for the largest GS configuration.
Because of their SMP character, users can employ OpenMP for shared-memory parallelisation on the GS systems to up to 32 processors in the GS320. Of course also MPI can be used along with the full range of Compaq compilers.

Section 1.9.2

Measured Performances

: In Dong02 a performance of 47.1 Gflop/s is given for a 32-processor GS320 system in solving linear system of order 40,000. An efficiency of 73.5%. Moreover ES40-based GS320's at a clock frequency of 731 MHZ have been 2-way and 4-way clustered which yielded speeds of 63.8 and 87.5 Gflop/s, respectively. As the internode bandwidth of the clusters markedly less, the efficiencies dropped accordingly to 68.2 and 46.7% respectively.

Section 1.9.3

TOP500 Systems

Section 1.10

EnterTheGrid description of company

Section 1.11

The Cray SX-6

machine-type:
operating-system:
connection-structure:
compilers:
vendor web-site:
year-of-introduction:

Section 1.11.1

Remarks

The Cray SX-6 is in fact the NEC SX-6 as marketed by Cray in the USA. See the section on the sx-6.html for the description.

Section 1.11.2

TOP500 Systems

Section 1.12

EnterTheGrid description of company

Section 1.13

The Cambridge Parallel Processing Gamma II Plus

machine-type: Processor array
operating-system: DEC, HP, or Sun workstation, stand-alone for dedicated applications
connection-structure: Internal OS transparent to the user, Unix on front-end
compilers: 2-D mesh, row- and column datapaths (see remarks)
vendor web-site:
year-of-introduction: http://www

Model	Gamma II Plus 1000	Gamma II Plus 4000
Clock cycle	30 MHz	30 MHz
Processor performance	0.6 Mflop/s	0.6 Mflop/s
Peak performance	0.6 Gflop/s	2.4 Gflop/s
Maximum system memory	128 MB	512 MB
Number of processors	- 1024	- 4096

Section 1.13.1

Remarks

In November 1995 the new Gamma II Plus models have been announced by CPP. In essence there is not much difference with its predecessor the DAP Gamma. However, the clock cycle has tripled to 33 ns with an equivalent rise in the peak performance of the systems.

The Gamma II is presented as the fourth generation of this type of machine. Indeed, the macro architecture of the systems has hardly changed since the first ICL DAP (the first generation of this system) was conceived. As in the ICL DAP in the Gamma 1000 models the 1024 processors are ordered in a 32 x 32 array, while the Gamma 4000 has 4096 processors arranged in a 64 x 64 square.

The systems are able to operate byte parallel on appropriate operands to speed up floating-point operations, however, for logical operations bit-wise operations are possible, which makes the machines quite fast in this respect. As the byte parallel code consists of separate sequences of microcode instructions, the bit processor plane and the byte processor plane are in fact independent and can work in parallel. This is also the case for I/O operations. Also character-handling can be done very efficiently. This is the reason that Gamma systems are often used for full text searches.

As in all processor-array machines, the control processor (called the Master Control Unit (MCU) in the Gamma II) has a separate memory to hold program instructions while the data are held in the data memory associated with each Processing Element (PE) in the processor array. So, for a Gamma 1000 with 128 MB of data memory each PE has 128 KB of data memory directly associated to it. To access data in other PE's memories these must be brought up to the data routing plane and shifted to the appropriate processor.

As already mentioned under the heading of the connection structure, there are two ways of connecting the PEs. One is the 2-D mesh that connects each element to its North-, East-, West-, and South neighbour. In addition there are row- and column data paths that enable the fast broadcast of a row or column to an entire matrix by replication. Conversely, they can be used for row or column-wise reduction of matrix objects into a column or row-vector of results from, e.g., a summing or maximum operation.

Separate I/O processors and disk systems can be attached to the Gamma directly thus not burdening the front-end machine (and the connection between front-end and Gamma-II) with I/O operations and unnecessary data transport. One of these I/O devices is the GIOC that can transport data to the data memory at a sustained rate of 80 MB/s transposing the data to the vertical storage mode of the data memory on the fly. Also, a direct video interface is available to operate a frame buffer.

A nice (non-standard) feature of the FORTRAN-PLUS compiler is the possibility to use logical matrices as indexing objects for computational matrix objects. This enables a very compact notation for conditional execution on the processor array. In addition, since 1997 C++ is available.

Section 1.13.2

Measured Performances

: In Flan91 the speed of matrix multiplication on various Gamma-II models (precursors of the Gamma systems) is analyzed. The documentation states 32-bit floating-point add speed of 1.68 Gflop/s on 4096 PEs, while a 32-bit 1,024 complex FFT would attain 2.49 Gflop/s. No independent performance figures for the Gamma II systems are available.

Section 1.13.3

TOP500 Systems

Section 1.14

EnterTheGrid description of company

Section 1.15

The Cray MTA

machine-type: Distributed-memory multi-processor
operating-system: Unix BSD4.4 + proprietary micro kernel
connection-structure: Fortran 77/90, ANSI C, C++
compilers: http://www.cray.com/products/systems/craymta/
vendor web-site:
year-of-introduction:

Model	MTA-2x
Clock cycle	
Processor performance	750 Mflop/s
Peak performance	192 Gflops
Maximum system memory	1 Tbyte
Number of processors	16 - 256

Section 1.15.1

Remarks

The exact peak speed of the MTA-2 systems cannot be given as this new CMOS version is yet to be delivered (see below for performances of the first GaAs-based MTA-1 machine). The data sheets on the Cray MTA-2 are not overly informative in this respect but a lower bound of the peak performance is quoted.

Although the memory in the MTA is physically distributed, the system is emphatically presented as a shared memory machine (with non-uniform access time). The latency incurred in memory references is hidden by *multi-threading*, i.e., usually many concurrent program threads (instruction streams) may be active at any time. Therefore, when for instance a load instruction cannot be satisfied because of memory latency the thread requesting this operation is stalled and another thread of which an operation can be done is switched into execution. This switching between program threads only takes 1 cycle. As there may be up to 128 instruction streams and 8 memory references can be issued without waiting for preceding ones, a latency of 1024 cycles can be tolerated. References that are stalled are retried from a retry pool. A construction that works out similarly is to be found in the Stern Computing Systems gone machines.

The connection network connects a 3-D cube of p processors with sides of $p^{1/3}$ of

which alternately the *x*- or *y* axes are connected. Therefore, all nodes connect to four out of six neighbours. In a *p* processor system the worst case latency is $4.5p^{1/3}$ cycles; the average latency is $2.25p^{1/3}$ cycles. Furthermore, there is an I/O port at every node. Each network port is capable of sending and receiving a 64-bit word per cycle which amounts to a bandwidth of 5.33 GB/s per port. In case of detected failures, ports in the network can be bypassed without interrupting operations of the system.

Although the MTA should be able to run "dusty-deck" Fortran programs because parallelism is automatically exploited as soon as an opportunity is detected for multi-threading, it may be (and often is) worthwhile to explicitly control the parallelism in the program and to take advantage of known data locality occurrences. MTA provides handles for this in the form of library routines, including synchronisation, barrier, and reduction operations on defined groups of threads. Controlled and uncontrolled parallelism approaches may be freely mixed. Furthermore, each variable has a full/empty bit associated with it which can be used to control parallelism and synchronisation with almost zero overhead.
A first MTA-2 system with 28 processors (instead of the normal 32) will be installed at the Naval Research Lab, USA, in 2002.

Section 1.15.2

Measured Performances

: The company has presently delivered a 16-processor system to the San Diego Supercomputing Center. This system runs at a clock cycle of 4.4 ns instead of the planned 3 ns. Consequently, the peak performance of a processor is 450 Mflop/s. Using the http://www.euroben.nl a performance of 388 Mflop/s out of 450 Mflop/s was found for an order 800 matrix-vector multiplication, an efficiency of 86%. For 1-D FFTs up to 1 million elements a speed of 106 Mflop/s was found on 1 processor and the about the same speed on 4 processors due to an insufficient availability of parallel threads.

Section 1.15.3

TOP500 Systems

Section 1.16

EnterTheGrid description of company

Cray

Cray adresses the high-end supercomputer market with the MTA , SV1 and Cray T3E superocmputers. It distributes the NEC vector supercomputers.

http://www.cray.com/

Section 1.17

IBM eServer p690

machine-type: RISC-based distributed-memory multi-processor cluster
operating-system: AIX (IBMs Unix variant)
connection-structure: -switch
compilers: XL Fortran (Fortran 90), HPF, XL C, C++
vendor **web-site:** http://www-1.ibm.com/servers/eserver/pseries/hardware/datactr/p690_desc.html
year-of-introduction: 2001 (16/32-CPU POWER4 SMP)

Model	eServer p690
Clock cycle	1.3 GHz
Processor performance	5.2 Gflop/s
Peak performance	166.4 Gflop/s
Maximum system memory	128 Tbyte
Number of processors	8 - 16384

Section 1.17.1

Remarks

The eServer p690 is the successor of the RS/6000 SP. It retains much of the macro structure of this system: multi-CPU nodes are connected within a frame either by a dedicated switch or by other means, like switched Ethernet. The structure of the nodes, however, has changed considerably, see \ref{s:pwr4}. Up to four Multichip Modules (MCMs) are housed in a node totaling 16 or 32 CPUs in a node depending on whether the dual or single core version of the chip is used. For High Performance Computing IBM recommends to employ the 16 CPU, single core, nodes because a higher effective bandwidth from the L2 cache can be expected in this case. For less data intensive work that primarily uses the L1 cache the difference would be small while there is a large cost advantage using the 32-CPU so-called Turbo nodes.

The p690 is accessed through a front-end control workstation that also monitors system failures. Failing nodes can be taken off line and exchanged without interrupting service.

The so-called high-performance switch, the SP Switch2, is an Omega-switch as described in the section on sm-mimd.html and, although we mentioned only the highest speed option for the communication, the high-performance switch, there is a wide range of other options that could be chosen instead: Ethernet, Token Ring, FDDI, etc., are all possible. The high performance switch is the third generation of this interconnect. The single-direction bandwidth is quoted as 500 MB/s and tests with MPI-based point-to-point communication from the EuroBen Distributed memory benchmark have shown that one can come very close to this limit.

Applications can be run using PVM or MPI. Also High Performance Fortran is supported, both a proprietary version and a compiler from the Portland Group. IBM uses its own PVM version from which the data format converter XDR has been stripped. This results in a lower overhead at the cost of generality. Also the MPI implementation, MPI-F, is optimised for the eServer p690 systems. As the nodes are

in effect shared-memory SMP systems, within the nodes OpenMP can be employed for shared-memory parallelism and it can be freely mixed with MPI if needed.

The standard commercial models that are marketed contain up to 128 nodes. However, on special request systems with up to 512 nodes can be built. This largest configuration is used in the table above (although never a system of a size exceeding 128 nodes has been sold yet).

Section 1.17.2

Measured Performances

In Dong02 a performance of 2310 Gflop/s for an 864 processor (54 HPC-node) system is reported for solving a 275,000-order dense linear system yielding an efficiency of 51%. A system with 8 Turbo nodes was reported to obtain a speed of 737 Gflop/s out of 1331 Gflop/s on a linear system of size 285,000, an efficiency of 55%. As this type of application primarily operates from the L1 cache, the more or less similar efficiencies are as expected.

Section 1.17.3

TOP500 Systems

2	ASCI White, SP Power3 375 MHz	http://www.llnl.gov/
5	SP Power3 375 MHz 16 way	http://www.nersc.gov/
8	pSeries 690 Turbo 1.3GHz	http://www.csm.ornl.gov/ccs/
9	ASCI Blue-Pacific SST, IBM SP 604e	http://www.llnl.gov/
10	pSeries 690 Turbo 1.3GHz	
11	SP Power3 375 MHz 16 way	http://www.awe.co.uk
12	pSeries 690 Turbo 1.3GHz	
16	SP Power3 375 MHz	http://www.navo.hpc.mil/
17	SP Power3 375 MHz 16 way	http://www.dwd.de/
18	SP Power3 375 MHz 16 way	http://www.scd.ucar.edu/
20	SP Power3 375 MHz	http://www.ncep.noaa.gov/
21	SP Power3 375 MHz	http://www.ncep.noaa.gov/
23	SP Power3 375 MHz 16 way	http://www.llnl.gov/
25	SP Power3 375 MHz 8 way	http://www.sdsc.edu/
31	SP Power3 375 MHz 16 way	http://www.mhpcc.edu/
32	pSeries 690 Turbo 1.3GHz GigEth	http://www.cineca.it
33	pSeries 690 1.1GHz GigEth	http://www.csc.fi/english/
34	pSeries 690 1.1GHz GigEth	http://www.fsu.edu/
40	SP Power3 375 MHz	http://www.charlesschwab.com/
43	SP Power3 375 MHz	http://www.ncsc.org/
44	pSeries 690 Turbo 1.3GHz	http://www.ibm.com
46	SP Power3 375 MHz	http://www.csm.ornl.gov/ccs/
47	pSeries 690 Turbo 1.3GHz GigEth	http://www.rrzn.uni-hannover.de
48	pSeries 690 Turbo 1.3GHz GigEth	http://www.zib.de/
58	pSeries 690 Turbo 1.3GHz GigEth	http://www.idris.fr/
59	pSeries 690 Turbo 1.3GHz GigEth	http://www.nchc.gov.tw/
60	pSeries 690 Turbo 1.3GHz GigEth	http://www.cscs.ch

62	pSeries 690 Turbo 1.3GHz GigEth	http://www.colsa.com/
66	SP Power3 375 MHz 16 way	http://www.sprint.com/
67	pSeries 690 Turbo 1.3GHz GigEth	http://www.gm.com/
68	pSeries 690 Turbo 1.3GHz GigEth	http://www.ihpc.nus.edu.sg/
69	SP Power3 375 MHz 16 way	http://www.research.ibm.com
70	SP Power3 375 MHz	http://www.dtag.de/
71	SP Power3 375 MHz	http://www.asc.hpc.mil
73	SP Power3 375 MHz	http://www.statefarm.com/
74	SP Power3 375 MHz 16 way	http://www.saudiaramco.com/
75	SP Power3 375 MHz 16 way	http://www.arl.mil/
76	SP Power3 375 MHz	
77	SP Power3 375 MHz	http://www.indiana.edu/iub/
81	pSeries 690 Turbo 1.3GHz GigEth	http://www.rzg.mpg.de/
82	SP Power3 375 MHz	http://www.statefarm.com/
89	SP Power3 375 MHz	http://www.cines.fr
91	pSeries 690 Turbo 1.3GHz GigEth	http://www.eds.com
94	SP Power3 375 MHz 16 way	http://www.llnl.gov/
96	ASCI Blue-Pacific CTR, IBM SP 604e	http://www.llnl.gov/
107	SP Power3 375 MHz	
109	pSeries 690 Turbo 1.3GHz GigEth	
110	pSeries 690 Turbo 1.3GHz GigEth	
138	SP Power3 375 MHz	http://www.msi.umn.edu/
139	SP Power3 375 MHz	http://www.mhpcc.edu/
154	pSeries 690 1.1GHz GigEth	http://i2.com/
156	SP Power3 222 MHz	http://www.wes.hpc.mil/
157	pSeries 690 Turbo 1.3GHz GigEth	
158	pSeries 690 Turbo 1.3GHz GigEth	http://super.seri.re.kr/
159	pSeries 690 Turbo 1.3GHz GigEth	
164	SP Power3 375 MHz	http://www.edinfor.pt/
168	SP Power3 375 MHz	http://www.pgs.com/
173	SP Power3 375 MHz	http://www.smdc.army.mil/
183	SP Power3 375 MHz	http://www.detecsm.de/
186	SP Power3 375 MHz	http://www.purdue.edu/PUCC/
231	pSeries 690 Turbo 1.3GHz GigEth	http://www.bouygtel.com/
232	pSeries 690 Turbo 1.3GHz GigEth	http://scv.bu.edu/SCV/
233	pSeries 690 Turbo 1.3GHz GigEth	
234	pSeries 690 Turbo 1.3GHz GigEth	http://www.tu-darmstadt.de/
235	pSeries 690 Turbo 1.3GHz GigEth	http://www.parallab.uib.no/
244	SP Power3 375 MHz	http://www.gwdg.de/
245	SP Power3 375 MHz	http://www.deere.com
246	SP Power3 375 MHz 16 way	http://www.saudiaramco.com/
250	SP Power3 375 MHz	http://www.psa.fr/
252	SP Power3 375 MHz	http://www.bayer.com/
265	SP Power3 375 MHz	http://www.arsc.edu/
267	pSeries 690 1.1GHz GigEth	http://www.gwdg.de/
268	pSeries 690 1.1GHz GigEth	
269	pSeries 690 1.1GHz GigEth	http://www.verizon.com/
270	pSeries 690 Turbo 1.3GHz GigEth	

322	SP Power3 375 MHz	http://www.painewebber.com
354	pSeries 690 Turbo 1.3GHz	http://www.arl.mil/
355	SP Power3 375 MHz 4/16 way	http://www.pik-potsdam.de
365	pSeries 690 1.1GHz GigEth	http://www.commerzbank.de/
373	SP Power3 375 MHz	http://www.idris.fr/
376	SP Power3 375 MHz	http://www.bankofamerica.com/
377	SP Power3 375 MHz	http://www.hpcf.cam.ac.uk/
378	SP Power3 375 MHz	http://www.detecsm.de/
379	SP Power3 375 MHz	http://www.fsu.edu/
380	SP Power3 375 MHz	http://www.nchc.gov.tw/
386	pSeries 690 1.1GHz GigEth	http://www.deutsche-bank.de/
387	pSeries 690 1.1GHz GigEth	http://www.mckesson.com/
388	SP Power3 375 MHz	http://www.abnamro.com/
389	SP Power3 375 MHz	http://www.daimlerchrysler.com/
391	SP Power3 375 MHz	
392	SP Power3 375 MHz	http://afw.offutt.af.mil/
393	SP Power3 375 MHz	
394	SP Power3 375 MHz	http://www.kroger.com/
395	SP Power3 375 MHz	http://philipmorris.com/
402	SP Power3 375 MHz	http://www.ponl.com/
403	pSeries 690 Turbo 1.3GHz GigEth	
404	pSeries 690 Turbo 1.3GHz GigEth	http://www.deutsche-bank.de/
405	pSeries 690 Turbo 1.3GHz GigEth	http://www.llnl.gov/
406	pSeries 690 Turbo 1.3GHz GigEth	http://www.lufthansa.com/
407	pSeries 690 Turbo 1.3GHz GigEth	http://www.nas.nasa.gov/
408	pSeries 690 Turbo 1.3GHz GigEth	http://www.sara.nl/
409	pSeries 690 Turbo 1.3GHz GigEth	http://www.tacc.utexas.edu/
413	SP Power3 375 MHz	http://www.supnet.com/
414	SP Power3 375 MHz	http://www.unilever.com/
420	SP Power3 375 MHz	http://www.dassault-aviation.fr/
421	SP Power3 375 MHz 16 way	http://www.snu.ac.kr/
422	pSeries 690 1.1GHz GigEth	http://www.adp.com/
427	SP Power3 200 MHz	http://www.aist.go.jp/TACC/index_e.html
442	SP Power3 375 MHz	http://www.prudential.com/
443	SP Power3 375 MHz	http://www.mcc.ac.uk/
463	SP Power3 375 MHz	
465	SP Power3 375 MHz	http://www.pulsen.se/
471	SP Power3 375 MHz 16 way	http://www.caspur.it
472	SP Power3 375 MHz 16 way	http://www.cineca.it
473	SP Power3 375 MHz 16 way	http://www.sara.nl/
474	SP Power3 375 MHz 16 way	http://www.toshiba.co.jp/
475	SP Power3 375 MHz 16 way	http://www.uvic.ca/minerva
476	pSeries 690 1.1GHz GigEth	http://www.duke-energy.com/
477	pSeries 690 1.1GHz GigEth	http://www.wiliamsenergy.com/
499	SP Power3 375 MHz	http://www.kff.org/

Section 1.18

EnterTheGrid description of company

IBM

IBM produces the RS/6000 SP parallel supercomputer series. Today these are part of the eServer pSeries product line. The company holds the first position in the November 2001 TOP500 and has the largest number of machines in that list.

The IBM eServer pSeries is a broad product line that ranges from powerful workstations ideal for mechanical design to mission-critical SMP servers for solutions such as ERP, SCM, CRM, transaction processing and Web serving, all the way up to parallel systems of clusters of pSeries SMP's that are suitable for handling the grand challenges in Scientific and Technical Computing.

IBM eServer pSeries 690 code named "Regatta" is an 8- to 32-way SMP server utilizing the first ever POWER4 dual processor on a chip using IBM advanced silicon-on-insulator and copper technologies.

The SP is a distributed memory, multinode server designed for demanding technical and commercial workloads. The system can run serial, symmetric multiprocessor and parallel workloads all managed from a central point-of-control. It can have up to 512 SMP nodes per system, this means such a parallel system can reach up to 85 TFlop/s with today's technology.

Some customer highlights

1. ASCI White is one of the largest supercomputer installations in the world. It has a peak computational performance of 12.28 Tflop/s. Possessing more than 160 Tbytes of disk storage capacity, it holds 16,000 times more data than the average desktop PC. The U.S. Department of Energy, as part of its Accelerated Strategic Computing Initiative (ASCI), aims at eliminating live nuclear testing without compromising the nation's safety and security.
2. IBM recently built Switzerland's largest Supercomputer as part ot the Centro Svizzero di Calcolo Scientifico (CSCS) of the ETH Zurich. Solving gran challenges in areas like material sciences, the pSeries 690 based system supplying 1.3 Tflop/s is one of Europe's fastest supercomputers for public research.
3. The Max Planck Society has contracted IBM to build one of Europe's largest supercomputers, the first TFlop/s now up and running.
4. The German Met Office (DWD) is doing its weather forecast production on an IBM supercomputer (number 10 on the top500 list) since April 9 2002
5. The European Centre for Medium-Range Weather Forecasts (ECMWF) has asked IBM to build both a powerful supercomputer and a data management system for weather prediction, enabling meteorologists to offer new and much improved forecasts.
6. The North German Supercomputer Project (HLRN) has choosen IBM to build a POWER4 based parallel supercomputer system at two sites with a fast network in between Hannover and Berlin.

http://www-1.ibm.com/grid/index.shtml

Section 1.19

The Quadrics Apemille

machine-type: Processor array
operating-system: Almost any Unix workstation
connection-structure: Internal OS transparent to the user, Unix on front-end
compilers: 3-D mesh, (see remarks)
vendor web-site:
year-of-introduction: http://www

Model	Apemille
Clock cycle	267 MHz
Processor performance	533 Mflop/s
Peak performance	1 Tflop/s
Maximum system memory	64 Gbyte
Number of processors	8 - 2048

Section 1.19.1

Remarks

The Apemille is a commercial spin-off of the APE-1000 project of the Italian National Institute for Nuclear Physics and a successor to the APE-100 systems. The systems are available in multiples of 8 processor nodes where up to 16 boards can be fitted into one crate or in multiples of 128 nodes by adding up to 15 crates to the minimal 1-crate system. The interconnection topology of the Quadrics is a 3-D grid with interconnections to the opposite sides (so, in effect a 3-D torus). The 8-node floating-point boards (FPBs) are plugged into the crate backplane which provides point-to-point communication and global control distribution. The FPBs are configured a 2^3 cubes that are connected to the other boards appropriately to arrive at the 3-D grid structure.

The basic floating-point processor, the so-called MAD chip, contains a register file of 128 registers. Of these registers the first two hold permanently the values 0 and 1 to be able to express any addition or multiplication as a ``normal operation'', i.e., a combined multiply-add operation, where an addition is of the form, $a \times b + 0$ and a multiplication is $a \times 1 + b$. In favourable circumstances the processor can therefore deliver two floating-point operations per cycle. Instructions are centrally issued by the controller at a rate of one instruction every two clock cycles.

Communication is controlled by the Memory Controller and the Communication Controller which are both housed on the backplane of a crate. When the Memory Controller generates an address it is decoded by the Communication Controller. In case non-local access is desired, the Communication Controller will provide the necessary data transmission. The memory bandwidth per processor is not disclosed in the documentation, nor the bandwidth for non-local communication. Regrettably, Quadrics provides no details on local or global communication speeds whatsoever.

The Apemille communicates with the front-end system via a PCI adapter card and should therefore have a bandwidth of about 100 MB/s. The actual speed is not specified, however. The interface can write and read the memories of the nodes and the Controller. I/O and should have a bandwidth up to 8.5 GB/s according to the documentation.

The TAO language has several extensions to employ the SIMD features of the Quadrics. Firstly, floating-point variables are assumed to be local to the processor that owns them, while integer variables are assumed to be global. Local variables can be promoted to global variables. Other extensions are the `ANY`, `ALL`, and `WHERE/END WHERE` keywords that can be used for global testing and control. Processors that not meet a global condition effectively skip the operation(s) that are associated with it. For easy referencing nearest-neighbour locations special constants `LEFT`, `RIGHT`, `UP`, `DOWN`, `FRONT`, and `BACK` are provided. In addition, new data types and operators on these data types are supported together with overloading of operators. This enables very concise code for certain types of calculations.

Section 1.19.2

Measured Performances

No measured performances have been reported for this machine.

Section 1.19.3

TOP500 Systems

Section 1.20

EnterTheGrid description of company

QSW

Quadrics is a provider of high performance clustering technology. Since being established in 1996, Quadrics has been able to call on a long heritage of technological expertise and the strength of one of Europe's leading corporations. Quadrics' software and hardware expertise is behind some of the world's fastest computers.

http://www.quadrics.com/

Section 1.21

The Hitachi SR8000

machine-type: RISC-based distributed memory multi-processor
operating-system: HI-UX/MPP (Micro kernel Mach 3.0)
connection-structure: Mult-dimensional crossbar (see remarks)
compilers: Fortran 77, Fortran 90, Parallel Fortran, HPF, C, C++
vendor web-site: http://www.hitachi.co.jp/Prod/comp/hpc/eng/sr81e.html
year-of-introduction: 1998, E1 and F1: 1999, G1: 2000

Model	SR8000	SR8000 E1	SR8000 F1
Clock cycle	250 MHz	300 MHz	375 MHz

Processor performance	8 Gflop/s	9.6 Gflop/s	12 Gflop/s
Peak performance	1 Tflop/s	4.9 Tflop/s	6.1 Tflop/s
Maximum system memory	1 Tbyte	8 Tbyte	8 Tbyte
Number of processors	4 - 128	4 - 512	4 - 512

Section 1.21.1

Remarks

The SR8000 is the third generation of distributed-memory parallel systems of Hitachi. It is to replace both its direct predecessor, the SR2201 and the late top-vectorprocessor, the S-3800 (see gone).

The basic node processor is a 2.22--4 ns clock PowerPC node with major enhancements made by Hitachi. E.g., a hardware barrier synchronisation is added and the additions required for "Pseudo Vector Processing" (PVP). The latter means that for operations on long vectors one does not incur the detrimental effects of cache misses that often ruin the performance of RISC processors unless code is carefully blocked and unrolled. This facility was already available on the SR2201 and experiments have shown that this idea seems to work well (see Hisr2201).

The peak performance per basic processor, or IP, can be attained with 2 simultaneous multiply/add instructions resulting in a speed of 1 Gflop/s on the SR8000. However, eight basic processors are coupled to form one processing node all addressing a common part of the memory. For the user this node is the basic computing entity with a peak speed of 8 Gflop/s. Hitachi refers to this node configuration as COMPAS, **C**o-**o**perative **M**icro-**P**rocessors in single **A**ddress **S**pace. In fact this is a kind of SMP clustering as discussed in the sections on architecture and ccNUMA.html. A difference with most of these systems is that for the user the individual processors in a cluster node are not accessible. Every node also contains an SP, a system processor that performs system tasks, manages communication with other nodes and a range of I/O devices.

The SR8000 has a multi-dimensional crossbar with a bi-directional link speed of 1 GB/s. From 4--8 nodes the cross-section of the network is 1 hop. For configurations 16--64 it is 2 hops and from 128-node systems on it is 3 hops.

The E1 and F1 models are in almost every respect equal to the basic SR8000 model, however, the clock cycles for these models are 3.3 and 2.66 ns, respectively. Furthermore, the E1, F1, and G1 models can house twice the amount of memory per node and the maximum configurations can be extended to 512 processors making them at the time of writing this report the most powerful commercially available systems --- at least in theory. The Hitachi documentation quotes a bandwidth of 1.2 GB/s for the network in the E1 model while it is 1 GB/s for the basic SR8000 and the F1. By contrast, the G1 model has a bandwidth of 1.6 GB/s.

Like in some other systems as the t3e.html#t3e, and the compaqsc.html#compaqsc, and the late NEC Cenju-4, one is able to directly access the memories of remote processors. Together with the very fast hardware-based barrier synchronisation this should allow for writing distributed programs with very low parallelisation overhead.

The following software products will be supported in addition to those already mentioned above: PVM, MPI, PARMACS, Linda, and FORGE90. In addition a

numerical libraries like NAG and IMSL are offered.

Section 1.21.2

Measured Performances

Results for the all of the SR8000 types are available from Dong02, of which we quote the most significant ones. On a 144-node G1 (450 MHz) configuration a speed of 1709 Gflop/s out of 2074 was observed, an efficiency of 63% for the solution of a 141,000 full linear system. On a 112-node 375 MHz F1 model 1035 out of 1344 Gflop/s could be achieved, an efficiency of 77%. On a single node of this processor speeds of over 6.2 and 4.1 Gflop/s were measured in solving a full linear system and a full symmetric eigenvalue problem of order 5000, respectively (see EurB99 for the last two results). Furthermore 2 SR8000 G1 frames have been coupled and a speed of 1709 Gflop/s out of 2074 has been attained on 1152 processors for solving a 141,000-order linear system. The efficiency in this case is 82%, quite high for externally coupled systems.

Section 1.21.3

TOP500 Systems

13	SR8000/MPP	http://www.u-tokyo.ac.jp/
14	SR8000-F1/168	http://www.lrz-muenchen.de/
26	SR8000-F1/100	http://www.kek.jp/
29	SR8000/128	http://www.u-tokyo.ac.jp/
41	SR8000-G1/64	http://www.imr.tohoku.ac.jp/Eng/index.html
50	SR8000-E1/80	http://www.kishou.go.jp/english/index.html
61	SR8000-F1/60	http://www.issp.u-tokyo.ac.jp/index_e.html
98	SR8000/64	http://www.aist.go.jp/TACC/index_e.html
187	SR8000/36	http://www.mri-jma.go.jp/
242	SR8000/32	http://www.hucc.hokudai.ac.jp
263	SR8000-F1/20	http://www.jaeri.go.jp/english/index.cgi
266	SR8000-G1/16	http://www.hitachi.co.jp/Div/merl/index-e.html
426	SR8000-G1/12	http://www.rccp.tsukuba.ac.jp/
466	SR8000/20	http://www.ism.ac.jp/index-e.html

Section 1.22

EnterTheGrid description of company

Hitachi Europe supercomputer

Hitachi markets the vector and massively parallel processing (MPP) architectures, based SR8000 supercomputer. The largest vector supercomputer in Europe is an Hitachi machine at Leibniz Rechencentrum in Munich.

Hitachi has a long history of supercomputing, supplying the first Japanese manufactured supercomputer to the University of Tokyo in 1983. Among the company's major acheivements was the introduction of the S-3000 series of vector

supercomputers. This series culminated in the S-3800 in 1993, a vector processor of 8 Gflops peak performance.

In 1996, Hitachi launched the massively parallel SR2201 supercomputer. The 1024-node system with a peak performance of 307 Gflops was cited as the world's most powerful computer in the June 1996 Top500 list. The most recent product from Hitachi is the SR8000 Super Technical Server, which has integrated concepts from vector and MPP architectures into a single system.

Some of the largeste machines in the world are Hitachi SR8000 supercomputers. The customer list includes:

- University of Tokyo has two of these supercomputers
- Leibniz Rechenzentrum houses the top-Europe machine
- High Energy Accelerator Research Organization/KEK in Japan is a typical research organisation
- Institute for Materials Research at Tohoku University
- In Stuttgart, HWW/Universitaet Stuttgart and DLR jointly operate an Hitachi supercomputer

http://www.hitachi-eu.com/hel/hpcc

Section 1.23

The Sun Fire 3800-15K

machine-type: RISC-based distributed-memory multi-processor
operating-system: Solaris (Sun's Unix flavour)
connection-structure: Crossbar (see remarks)
compilers: Fortran 77, Fortran 90, HPF, C, C++
vendor web-site: http://www.sun.com/servers/highend/sunfire15k/details.html
year-of-introduction: 2001

Model	Fire 3800-15K
Clock cycle	900 MHz
Processor performance	1.8 Gflop/s
Peak performance	190.8 Gflop/s
Maximum system memory	576 Gbyte
Number of processors	- 106

Section 1.23.1

Remarks

In the Fire 15K the processor/memory boards are plugged into a backplane that is an 18×18 flat crossbar. Each board contains four 900 MHz UltraSPARC III processors and a maximum of 32 GB of memory. So, normally the maximum number of

processors would 72. However, the 15K in fact contains *three* of these 18×18 crossbars, for data, addresses, and signals. It is possible to sacrifice I/O capacity and use 17 of the 18 slots of the second crossbar to put in 2-CPU boards without local memory, adding another 34 processors to obtain the maximum of 106. Obviously, such a system is less balanced and such a configuration will normally only be chosen for very specific compute-intensive tasks with small I/O requirements. Because of the flat crossbar memory access is uniform and the aggregate bandwidth of the crossbar is 172.8 GB/s. This is equivalent to 2.4 GB/s/processor or 2.66B/cycle. So, an 8-byte operand needs 3 cycles to be shipped to the processor. Of course, for processors in excess to 72 that are not on the data backplane the situation is more complicated and it is hard to estimate what the effective bandwidth would be.

The Fire 15K is a typical SMP machine with provisions for shared-memory parallelism in the Fortran and C(++) compilers by directives in the source code. Sun has joined the OpenMP consortium for standardising the shared-memory programming model.

Section 1.23.2

Measured Performances

In Top500 a speed of 357 Gflop/s is reported for a 4-way cluster of 72 processor machines in solving a dense linear system of unspecified size. The efficiency for this problem is 69%.

Section 1.23.3

TOP500 Systems

111	HPC 4500 400 MHz Cluster	
112	HPC 4500 400 MHz Cluster	
113	HPC 4500 400 MHz Cluster	
114	HPC 4500 400 MHz Cluster	http://www.sun.com/
129	Fire 15K	http://www.rwth-aachen.de/
271	Fire 15K	
272	Fire 15K	
273	Fire 15K	http://www.bmw.com/
274	Fire 15K	http://www.daimlerchrysler.com/
275	Fire 15K	http://www.daimlerchrysler.com/
276	Fire 15K	http://www.daimlerchrysler.com/
277	Fire 15K	
278	Fire 15K	http://www.kyoto-u.ac.jp/
309	Fire 6800/Sun Fire Link	
358	Fire 6800	http://www.rwth-aachen.de/
359	Fire 6800	http://www.rwth-aachen.de/
478	HPC 10000 400 MHz Cluster	http://www.mot.com/
479	HPC 10000 400 MHz Cluster	http://www.ci.nyc.ny.uc/
480	HPC 10000 400 MHz Cluster	http://www.sun.com/
481	HPC 10000 400 MHz Cluster	http://www.arl.mil/
482	HPC 10000 400 MHz Cluster	
483	HPC 10000 400 MHz Cluster	http://www.clearstream.com/
484	HPC 10000 400 MHz Cluster	http://www.clearstream.com/

485	HPC 10000 400 MHz Cluster	
486	HPC 10000 400 MHz Cluster	
487	HPC 10000 400 MHz Cluster	http://www.ford.com/
488	HPC 10000 400 MHz Cluster	http://www.gte.com/
489	HPC 10000 400 MHz Cluster	http://www.gte.com/
490	HPC 10000 400 MHz Cluster	http://www.gateway.com/
491	HPC 10000 400 MHz Cluster	
492	HPC 10000 400 MHz Cluster	http://www.mobilcom.de/
493	HPC 10000 400 MHz Cluster	http://www.mobilcom.de/
494	HPC 10000 400 MHz Cluster	
495	HPC 10000 400 MHz Cluster	
496	HPC 10000 400 MHz Cluster	
497	HPC 10000 400 MHz Cluster	
498	HPC 10000 400 MHz Cluster	

Section 1.24

EnterTheGrid description of company

Sun Microsystems

Sun Microsystems is active in HPC computing systems and software. Sun has traditonally been very strong in science and engineering. With the Starfire, which was very successful in commercial applications, it had one of the first machine massively present in the TOP500 with a large percentage of indsutrial applications. Sun was also pioneering in Grid computing with the GridEngine, which originated in the German company Genias.

Sun has a whole range of computer systems, ranging from workstations to high-end servers. The former high-end family, the Starfire Enterprise 10000 Server was well represented in the TOP500 for a number of years. Sun never tried to bring out a machine that could challenge the first place, however. The current top-offering is the Sun Fire 15K Server.

Sun Grid Engine finds a pool of idle resources and harnesses it productively, so an organisation gets as much as five to ten times the usable power out of systems on the network. That can increase utilisation to as much as 98%. Sun Grid Engine software aggregates available compute resources and delivers compute power as a network service.

SUN high end systems are also used in industry as the first examples show.

1. ***MobilCom*** in Germany has two PC 10000 400 MHz systems.
2. Sun Fire 15K servers, introduced September 2001, have been shipped to organizations such as Clearstream Services, Mid-Sweden University, Ocwen Technology Xchange, Tai Fook Securities Group, and Uppsala University.
3. Nortel Networks NITEC Monkstown is using the Sun Grid Engine software to control the access of simulation jobs to the hardware accelerator.
4. The CFD Group at SAAB has relied on the cluster computing capabilities of 100 networked Sun workstations to perform compute-intensive external aerodynamics and other computational fluid dynamics simulations with GridEngine.
5. debis Systemhaus in Germany is using a Sun Enterprise 10000 for application service providing.

http://www.sun.com/

Section 1.25

The HP 9000 SuperDome

machine-type: RISC-based ccNUMA system.
operating-system: HP-UX (HP's usual Unix flavour)
connection-structure: Crossbar
compilers: Fortran 77, Fortran 90, Parallel Fortran, HPF, C, C++
vendor web-site: http://www.hp.com/products1/servers/scalableservers/index.html
year-of-introduction: 2000

Model	HP 9000 SuperDome
Clock cycle	750 MHz
Processor performance	3 Gflop/s
Peak performance	192 Gflop/s
Maximum system memory	128 Gbyte
Number of processors	16 - 64

Section 1.25.1

Remarks

The Superdome is to replace the Exemplar V2600 system which is also still marketed by HP but not as a multi-node system anymore (see section gone). The aggregate peak speed of the Superdome is in fact 2 times lower than that of the 4-node V2600 because the same CPUs are used, but the maximal configuration only can harbour 64 processors against 128 in the 4-node V2600.

The connection structure of the Superdome has significantly improved over that of the V2600: where the latter had a crossbar within its 32-way SMP nodes with an aggregate bandwidth of 15.4 GB/s and a 3.8 GB/s aggregate bandwidth between the SMP nodes, in the Superdome the aggregate bandwidth in 64 GB/s in a 2-level crossbar. This greatly improves the communication within the system. The PA-RISC 8600 CPUs run at a clock frequency of 750 MHz. As a CPU contains 2 floating-point units that are able to execute a combined floating multiply-add instruction, in favourable circumstances four flops/cycle can be achieved and a Theoretical Peak Performance of 3 Gflop/s per CPU can be attained. This amounts to a peak speed of 192 Gflop/s for a full configuration.

As in the former systems a shared memory parallel model is supported. HP is a partner in the OpenMP organisation and will therefore provide this style of shared-memory parallel programming in addition to (and later on instead of) its proprietary parallel model.

Section 1.25.2

Measured Performances

In Dong02 a speed of 86.45 Gflop/s is reported for solving a full linear system of size 41,000. This amounts to an efficiency of 61%. Also results for a 4-way coupled system with a total of 256 processors are reported: solving a full linear system of order 340,092 showed a speed of 471 Gflop/s, 61% of the 768 Gflop/s peak. For coupling a Hyperfabric network was used showing no degradation with respect to the internal network.

Section 1.25.3

TOP500 Systems

95	SuperDome 750 MHz/HyperFabric	http://www.convex.com/
97	SuperDome 750 MHz/HyperPlex	http://www.hp.com/
108	SuperDome 750 MHz/HyperPlex	http://www.uky.edu/
127	SuperDome 750 MHz/HyperPlex	http://www.centrica.co.uk/
128	SuperDome 750 MHz/HyperPlex	http://www.convex.com/
166	SuperDome 750 MHz/HyperPlex	http://www.agilent.com/
188	SuperDome 750 MHz/HyperFabric	http://www.hp.com/
191	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
192	SuperDome 750 MHz/HyperPlex	http://www.capinfo.com.cn/
193	SuperDome 750 MHz/HyperPlex	http://www.hoovers.com/
194	SuperDome 750 MHz/HyperPlex	http://www.francetelecom.fr/
195	SuperDome 750 MHz/HyperPlex	http://www.netsiel.it/
196	SuperDome 750 MHz/HyperPlex	http://www.omnitel.it/
197	SuperDome 750 MHz/HyperPlex	http://www.debis.de/
198	SuperDome 750 MHz/HyperPlex	http://www.abnamro.com/
199	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
200	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
201	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
202	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
203	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
204	SuperDome 750 MHz/HyperPlex	http://www.bmw.com/
205	SuperDome 750 MHz/HyperPlex	http://www.sella.it/
206	SuperDome 750 MHz/HyperPlex	http://www.centrica.co.uk/
207	SuperDome 750 MHz/HyperPlex	http://www.centrica.co.uk/
208	SuperDome 750 MHz/HyperPlex	
209	SuperDome 750 MHz/HyperPlex	http://www.wachovia.com/
210	SuperDome 750 MHz/HyperPlex	http://www.wachovia.com/
211	SuperDome 750 MHz/HyperPlex	http://www.hutchison3g.com/
212	SuperDome 750 MHz/HyperPlex	http://www.is-informationssysteme.de/
213	SuperDome 750 MHz/HyperPlex	http://www.itellium.de/
214	SuperDome 750 MHz/HyperPlex	http://www.magirus.de/
215	SuperDome 750 MHz/HyperPlex	http://www.migros.ch/
216	SuperDome 750 MHz/HyperPlex	http://www.sprint.com/
217	SuperDome 750 MHz/HyperPlex	http://www.verizon.com/

218	SuperDome 750 MHz/HyperPlex	http://www.wind.it/
221	SuperDome 750 MHz/HyperPlex	http://www.alcoa.com/
222	SuperDome 750 MHz/HyperPlex	http://www.groupecegetel.fr/
223	SuperDome 750 MHz/HyperPlex	http://www.telecomitalia.it/
240	SuperDome 750 MHz/HyperPlex	http://www.gm.com/
279	SuperDome/HyperPlex	http://www.abchina.com/
280	SuperDome/HyperPlex	http://www.amdocs.com/
281	SuperDome/HyperPlex	http://www.amdocs.com/
282	SuperDome/HyperPlex	http://www.bell.ca/
283	SuperDome/HyperPlex	http://www.bell.ca/
284	SuperDome/HyperPlex	http://www.brasiltelecom.net.br/
285	SuperDome/HyperPlex	
286	SuperDome/HyperPlex	http://www.cayenta.com/
287	SuperDome/HyperPlex	
288	SuperDome/HyperPlex	
289	SuperDome/HyperPlex	
290	SuperDome/HyperPlex	
291	SuperDome/HyperPlex	http://www.newsky.com/
292	SuperDome/HyperPlex	http://www.nokia.com/
293	SuperDome/HyperPlex	http://www.one2one.co.uk/
294	SuperDome/HyperPlex	http://www.overwrite.com/
295	SuperDome/HyperPlex	http://www.saimaalines.fi/
296	SuperDome/HyperPlex	http://www.posdata.co.kr/
297	SuperDome/HyperPlex	http://www.posdata.co.kr/
298	SuperDome/HyperPlex	
299	SuperDome/HyperPlex	http://www.qwest.com/
300	SuperDome/HyperPlex	http://www.telkom.co.za/
301	SuperDome/HyperPlex	http://www.telkom.co.za/
302	SuperDome/HyperPlex	http://www.navy.mil/
303	SuperDome/HyperPlex	http://www.verizon.com/
304	SuperDome/HyperPlex	http://www.vodafone.com/
305	SuperDome/HyperPlex	http://www.microtunneling.de/
306	SuperDome/HyperPlex	http://www.amazon.com/
307	SuperDome/HyperPlex	
308	SuperDome/HyperPlex	http://www.kone.com/
310	SuperDome/HyperPlex	http://www.brasiltelecom.net.br/
311	SuperDome/HyperPlex	http://www.cisco.com/
312	SuperDome/HyperPlex	http://www.cisco.com/
313	SuperDome/HyperPlex	http://www.francetelecom.fr/
314	SuperDome/HyperPlex	
315	SuperDome/HyperPlex	
316	SuperDome/HyperPlex	
317	SuperDome/HyperPlex	
318	SuperDome/HyperPlex	http://www.lgeds.com/
319	SuperDome/HyperPlex	http://www.omnitel.it/
320	SuperDome/HyperPlex	http://www.raytheon.com/
321	SuperDome/HyperPlex	http://www.scourt.go.kr/english/
323	SuperDome 750 MHz/HyperPlex	http://www.att.com/

324	SuperDome 750 MHz/HyperPlex	http://www.amdocs.com/
325	SuperDome 750 MHz/HyperPlex	http://www.americanexpress.com/
326	SuperDome 750 MHz/HyperPlex	http://www.pharma.aventis.de/
327	SuperDome 750 MHz/HyperPlex	http://www.bellsouth.com/
328	SuperDome 750 MHz/HyperPlex	http://www.digitel.com.ve/
329	SuperDome 750 MHz/HyperPlex	http://www.daimlerchrysler.com/
330	SuperDome 750 MHz/HyperPlex	http://www.geappliances.de/
331	SuperDome 750 MHz/HyperPlex	http://www.hi3gaccess.se/
332	SuperDome 750 MHz/HyperPlex	
333	SuperDome 750 MHz/HyperPlex	http://www.netsiel.it/
334	SuperDome 750 MHz/HyperPlex	http://www.starbucks.com/
335	SuperDome 750 MHz/HyperPlex	http://www.telecomitalia.it/
336	SuperDome 750 MHz/HyperPlex	http://www.aol.com/
337	SuperDome 750 MHz/HyperPlex	http://www.aol.com/
338	SuperDome 750 MHz/HyperPlex	http://www.belgacom.be/
339	SuperDome 750 MHz/HyperPlex	http://www.belgacom.be/
340	SuperDome 750 MHz/HyperPlex	http://www.bell.ca/
341	SuperDome 750 MHz/HyperPlex	http://www.citibank.com/
342	SuperDome 750 MHz/HyperPlex	http://www.francetelecom.fr/
343	SuperDome 750 MHz/HyperPlex	http://www.francetelecom.fr/
344	SuperDome 750 MHz/HyperPlex	http://www.omnitelvodafone.it/
345	SuperDome 750 MHz/HyperPlex	http://www.sprint.com/
346	SuperDome 750 MHz/HyperPlex	http://www.sprint.com/
347	SuperDome 750 MHz/HyperPlex	http://www.sprint.com/
348	SuperDome 750 MHz/HyperPlex	http://www.transco.uk.com/
349	SuperDome 750 MHz/HyperPlex	http://www.transco.uk.com/
350	SuperDome 750 MHz/HyperPlex	http://www.transco.uk.com/
351	SuperDome 750 MHz/HyperPlex	http://www.transco.uk.com/
352	SuperDome 750 MHz/HyperPlex	http://www.transco.uk.com/
353	SuperDome 750 MHz/HyperPlex	http://www.verizon.com/
370	SuperDome 750/500 MHz/HyperPlex	http://www.dkfz-heidelberg.de/index.html
415	SuperDome 750 MHz/HyperPlex	http://www.cilea.it/
431	SuperDome/HyperPlex	http://www.arnold.af.mil/
432	SuperDome/HyperPlex	http://www.deltadt.com/
433	SuperDome/HyperPlex	http://www.kma.go.kr/
434	SuperDome/HyperPlex	http://www.nvon.nl/
435	SuperDome/HyperPlex	http://www.netsiel.it/
436	SuperDome/HyperPlex	http://www.nuon.nl/
437	SuperDome/HyperPlex	http://www.posco.co.kr/
438	SuperDome/HyperPlex	http://www.yodobashi.co.jp/
439	SuperDome/HyperPlex	http://www.yodobashi.co.jp/
440	SuperDome/HyperPlex	http://www.talk.com/
441	SuperDome/HyperPlex	http://www.ups.com/
444	SuperDome/HyperPlex	http://www.alliantenergy.com/
445	SuperDome/HyperPlex	http://www.americanair.com/
446	SuperDome/HyperPlex	http://www.americanair.com/
447	SuperDome/HyperPlex	http://www.americanair.com/
448	SuperDome/HyperPlex	http://www.americanair.com/

449	SuperDome/HyperPlex	http://www.bell.ca/
450	SuperDome/HyperPlex	http://cgi.resourceindex.com/
451	SuperDome/HyperPlex	http://www.equiva.com/
452	SuperDome/HyperPlex	http://www.equiva.com/
453	SuperDome/HyperPlex	http://www.fosythesolutions.com/
454	SuperDome/HyperPlex	http://www.hutchison.de/
455	SuperDome/HyperPlex	http://www.hutchison.co.th/
456	SuperDome/HyperPlex	http://www.online.arcor.net/
457	SuperDome/HyperPlex	http://www.nike.com/
458	SuperDome/HyperPlex	http://www.nike.com/
459	SuperDome/HyperPlex	http://www.sprint.com/
460	SuperDome/HyperPlex	http://www.sprint.com/
464	SuperDome 750 MHz/HyperPlex	http://www.gm.com/

Section 1.26

EnterTheGrid description of company

Section 1.27

The Cray SV1

machine-type: Shared-memory multi-vector processor.
operating-system: UNICOS (Cray Unix variant).
connection-structure: Crossbar.
compilers: Fortran 90, C, C++, Pascal, ADA.
vendor web-site: http://www.cray.com/products/systems/craysv1/
year-of-introduction: 2000

Model	Cray SV1ex-1A	Cray SV1ex-1	Cray SV1ex-4
Clock cycle	500 MHz	500 MHz	500 MHz
Processor performance	8 Gflop/s	8 Gflop/s	8 Gflop/s
Peak performance	32 Gflop/s	64 Gflop/s	256 Gflop/s
Maximum system memory	32 Gbyte	96 Gbyte	384 Gbyte
Number of processors	8 - 16	8 - 32	32 - 128

Section 1.27.1

Remarks

The Cray SV1ex series is a "midlife kicker" that bridges the gap between the Cray SV1 that appeared in 1998 and the SV2 which is expected to appear in 2002. Essentially the SV1ex machines are identical to the SV1s, however, the clock frequency has been raised by 50%. This speeds up the single-processor peak

performance from 1.2 to 1.8 Gflop/s. Furthermore, the speed of memory has increased by a factor of two which respect to the SV1.

The Cray SV1(ex) is the successor both to the CMOS-based Cray J90 and the Cray T90 which was based on ECL technology. The SV1ex systems are CMOS-based and therefore much cheaper to manufacture than the ECL-based systems. In this respect it has followed the trend set in by Fujitsu and NEC a few years ago with their vector systems (see the vpp5000.html and the sx-6.html). The Cray vector processor tradition has also been followed in that the SV1ex series uses its own Cray-specific floating-point format instead of the IEEE 754 standard.

The single-cabinet configurations come in two sizes, the SV1ex-1A and the SV1ex-1 that can house 4 and 8 processor boards, respectively. Each processor board contains 4 CPUs that can deliver a peak rate of 4 floating-point operations per cycle, amounting to a theoretical peak performance of 2 Gflop/s per CPU. However, 4 CPUs can be coupled *across* CPU boards in a configuration to form a so-called Multi Streaming Processor (MSP) resulting in a processing unit that has effectively a Theoretical Peak Performance of 8 Gflop/s. The reconfiguration into MSPs and/or single CPU combinations can be done dynamically as the workload dictates. The vector start-up time for the single CPUs is smaller than for MSPs, so for small vectors single CPUs might be preferable while for programs containing long vectors the MSPs should be of advantage. The number of combinations that can be made is large but at least 8 CPUs must be configured as single 2 Gflop/s CPUs. So a full SV1ex-1 cabinet may be configured as 32 single 2 Gflop/s CPUs or as 1--6 MSPs with the remaining processors as single CPUs.

Another feature in the SV1ex is a combined scalar and vector cache of 256 KB per CPU. This cache is important because the bandwidth of 6.4 GB/s per CPU board amounts to only 1.5 eight-byte operands per cycle. The cache can ship 4 operands per cycle to a CPU. This relative bandwidth is much smaller than what was offered in the former Cray systems which makes the cache all the more important. As the available bandwidth from a memory interface is divided over the 4 processors on a board on an as-needed basis and it is assumed that not all processors require the maximum amount of data all the time the average data requirement of the processor boards is hoped to be met.

Like in the NEC SX-6 single cabinets can be combined to form a cluster (Supercluster in Cray's terminology) by a so-called GigaRing. The GigaRing, which is also used to couple I/O sub-systems, is comprised of two counter-rotating rings with a bandwidth of 1 GB/s each. Where the systems in a cabinet are SM-MIMD systems, a multi-cabinet Supercluster is an DM-MIMD system and can be operated in parallel only by some parallel programming model like MPI or HPF. The SV1ex-4 is a standard configuration that is offered by Cray Inc. but larger clusters with up to 32 SV1ex-1 nodes are also possible.

Section 1.27.2

Measured Performances

In Dong02 a performance of 48.17 Gflop/s is reported for solving a dense linear system of size 40,320 on a 32-processor machine. This amounts to an efficiency of 75.3%.

Section 1.27.3

TOP500 Systems

Section 1.28

EnterTheGrid description of company

Section 1.29

The NEC SX-6

machine-type: Distributed-memory multi-vector processor
operating-system: Super-UX (Unix variant based on BSD V.4.3 Unix).
connection-structure: Multi-stage crossbar (see Remarks)
compilers: Fortran 90, HPF, ANSI C, C++
vendor web-site: http://www.sw.nec.co.jp/hpc/sx-e/sx6/index.htm
year-of-introduction: 2002

Model	SX-6i	SX-6A	SX-6xMy
Clock cycle	500 MHz	500 MHz	500 MHz
Processor performance	8 Gflop/s	8 Gflop/s	8 Gflop/s
Peak performance	8 Gflop/s	64 Gflop/s	
Maximum system memory	8 Gbyte	64 Gbyte	8 TB
Number of processors	-	4 - 8	8 - 1024

Section 1.29.1

Remarks

The SX-6 series is offered in numerous models but most of these are just smaller frames that house a smaller amount of the same processors. We only discuss the essentially different models here. All models are based on the same processor, an 8-way replicated vector processor where each set of vector pipes contains a logical, mask, add/shift, multiply, and division pipe (see section sm-simd.html for an explanation of these components). As multiplication and addition can be chained (but not division) the peak performance of a pipe set at 500 MHz is 1 Gflop/s. Because of the 8-way replication a single CPU can deliver a peak performance of 8 Gflop/s. The vector units are complemented by a scalar processor that is 4-way super scalar and at 500 MHz has a theoretical peak of 1 Gflop/s. The peak bandwidth per CPU is 32 GB/s or 64 B/cycle. This is sufficient to ship 8 8-byte operands back or forth and just enough to feed one operand to each of the replicated pipe sets.

The SX-6i is the single CPU system that because of the single chip implementation is offered as a desk side model. Also a rack model is available that enables housing two systems in a rack but there is no connection between the systems.

In a single frame of the SX-6A models fit up to 8 CPUs at the same clock frequency as the SX-6i. Internally the CPUs in the frame are connected by a 1-stage crossbar with the same bandwidth as that of a single CPU system: 32 GB/s/port. The fully configurated frame can therefore attain a peak speed of 64 Gflop/s.

In addition, there are multi-frame models (SX-6*x*M*y*) where *x = 8,...,1024* is the total number of CPUs and *y = 2,...,128* is the number of frames coupling the single-frame systems into a larger system. There are two ways to couple the SX-6 frames in a multi-frame configuration: NEC provides a full crossbar, the so-called IXS crossbar to connect the various frames together at a speed of 8 GB/s for point-to-point unidirectional out-of-frame communication (1024 GB/s bi-sectional bandwidth for a maximum configuration). Also a HiPPI interface is available for inter-frame communication at lower cost and speed. When choosing for the IXS crossbar solution, the total multi-frame system is globally addressable, turning the system into a NUMA system. However, for performance reasons it is advised to use the system in distributed memory mode with MPI.

The technology used is CMOS. This lowers the fabrication costs and the power consumption appreciably (the same approach is used in the vpp5000.html#vpp5000 and the sv1.html#sv1) and all models are air cooled.

For distributed computing there is an HPF compiler and for message passing an optimised MPI (MPI/SX) is available. In addition for shared memory parallelism, OpenMP is available.

Section 1.29.2

Measured Performances

Results for a 8-frame SX-6/128M16 processors are available from Dong02. The system attained 982 Gflop/s, an efficiency of 96%. The size of the linear system this result was 204,800.

Section 1.29.3

TOP500 Systems

1	Earth-Simulator	http://www.es.jamstec.go.jp/
19	SX-5/128M8 3.2ns	http://www.osaka-u.ac.jp/
24	SX-6/128M16	http://www.nec.co.jp/index e.html
86	SX-6/64M8	http://www.dkrz.de
87	SX-6/64M8	http://www.nal.go.jp/
88	SX-6/64M8	http://www.nies.go.jp/
155	SX-6/40M5	http://www.crl.go.jp/
162	SX-5/40M3	http://www.idris.fr/
220	SX-4/128H4	http://www.tohoku.ac.jp/
226	SX-5/32M2	http://www.bom.gov.au/
227	SX-5/32M2	http://www.tor.ec.gc.ca/
228	SX-5/32H2	http://www.nrim.go.jp/
253	SX-5/28M2	http://www.kma.go.kr/
366	SX-5/24M2	http://www.nec.co.jp/index e.html

Section 1.30

EnterTheGrid description of company

Section 1.31

The Cray T3E

machine-type: RISC-based distributed-memory multi-processor
operating-system: UNICOS/mk (micro kernel-based Unix)
connection-structure: 3-D Torus
compilers: Fortran 77, Fortran 90, HPF, ANSI C, C++.
vendor web-site: http://www.cray.com/products/systems/crayt3e/
year-of-introduction: T3E-1200E: 1998 T3E-1350: 2000

Model	T3E-1200E	T3E-1350
Clock cycle	600 MHz	675 MHz
Processor performance	1.2 Gflop/s	1.35 Gflop/s
Peak performance	2458 Gflop/s	2938 Gflop/s
Maximum system memory	4 Tbyte	1 Tbyte
Number of processors	6 - 2048	40 - 2176

Section 1.31.1

Remarks

The T3E is the second generation of DM-MIMD systems from Cray. Lexically, it follows in name after its predecessor T3D which name referred to its connection structure: a 3-D torus. In this respect it has still the same interconnection structure as the T3D. In many other respects, however, there are quite some differences. A first and important difference is that no front-end system is required anymore (although it is still possible to connect to a Cray vector systems). The systems up to 128 processors are air-cooled. The larger ones, from 256-2176 processors, are liquid cooled.

The T3E uses the DEC Alpha 21164 for its computational tasks. In 2000, a T3E-1350 was introduced that uses the latest 21164A processors at a clock rate of only 675 MHz but that is identical in almost all other aspects to the T3E-1200E. Cray stresses, that the processors are encapsulated in such a way that they can be exchanged easily for any other (faster) processor as soon as this would be available without affecting the macro-architecture of the system. However, in practice this is not likely to happen.

Each node in the system contains one processing element (PE) which in turn contains a CPU, memory, and a communication engine that takes care of communication between PEs. The bandwidth between nodes is quite high: 300 MB/s. Like the T3D, its predecessor, the T3E has hardware support for fast synchronisation. E.g., barrier synchronisation takes only one cycle per check.

Each node in the system contains one processing element (PE) which in turn contains a CPU, memory, and a communication engine that takes care of communication between PEs. The bandwidth between nodes is quite high: 325 MB/s, bi-directional. The T3E has hardware support for fast synchronisation. E.g., barrier synchronisation takes only one cycle per check. The node also contains a set of E-registers and streaming registers that allows for aggressive prefetching to ameliorate the restrictions

of the processor/memory bottleneck. An interesting additional feature is the availability of 32 contexts per processor which opens the door for multiprocessing.

In the T3E distributed I/O is present. For every 8 PEs an I/O channel can be configured in the air-cooled systems and 1 I/O channel per 16 nodes in the liquid-cooled systems. The maximum bandwidth for a channel is about 1 GB/s, the actual speed will be in the order of 500 MB/s.

The T3E supports various programming models. Apart from PVM and MPI for message passing and HPF for data distribution, a Cray proprietary one-sided communication library, the so-called `shmem` library can be employed for message passing. In addition, the BSP library (see Hill97), also a one-sided message passing library is available. The `shmem` library is implemented close to the hardware and shows very low latency of only 1.6 µs.

There are some differences in the available configurations between the T3E-1200 and the T3E-1350: In the T3E-1200 the amount of memory per node ranges from 64 MB to 2 GB while in the 1350 model there is only a choice between 256 and 512 MB per node. Furthermore, there is an air-cooled model (up to 128 PEs) of the T3E-1200 while the larger configurations are liquid-cooled. The T3E-1350 knows only liquid-cooled configurations that can be incremented from 40 processors on with modules of 8 processors. The 1200 systems start at 6 processors and modules of 4 or 8 processors can be added.

Section 1.31.2

Measured Performances

In Dong02 a speed of 1.127 Tflop/s is reported for the solution of a dense linear system of order 148800 on a T3E-1200 with 1488 processors. The efficiency for such an exercise is 63%. The same source quotes a speed of 113.9 out of 172.8 Gflop/s on a 128-processor T3E-1350, giving an efficiency of 66% for solving a size 89,088 linear system.

Section 1.31.3

TOP500 Systems

22	T3E1200	
27	T3E1200	http://www.arc.umn.edu/
30	T3E1200/900	http://www.wes.hpc.mil/
36	T3E900	
53	T3E1200	http://www.csar.cfs.ac.uk/
54	T3E1200	http://www.dwd.de/
72	T3E900	http://www.met-office.gov.uk/
79	T3E1200	http://www.met-office.gov.uk/
80	T3E	http://www.gsfc.nasa.gov/
99	T3E1200	http://www.cray.com/
100	T3E1200	http://www.fz-juelich.de
101	T3E1200	
102	T3E1200	
103	T3E1200	
104	T3E900	http://www.nersc.gov/

130	T3E	http://www.rzg.mpg.de/
134	T3E900	http://www.hlrs.de/
135	T3E900	http://www.psc.edu/
136	T3E900	http://www.zib.de/
177	T3E750	http://www.csc.fi/english/
230	T3E1200	
236	T3E	http://www.cray.com/
237	T3E	http://www.fz-juelich.de
249	T3E900	http://www.epcc.ed.ac.uk/
383	T3E900	http://www.msc.edu/
384	T3E900	http://www.arsc.edu/

Section 1.32

EnterTheGrid description of company

Section 1.33

The Fujitsu VPP5000 series

machine-type: Distributed-memory vector multi-processor
operating-system: UXP/V (a V5.4 based variant of Unix)
connection-structure: Full distributed crossbar
compilers: Fortran 90/VP (Fortran 90 Vector compiler), Fortran 90/VPP (Fortran 90 Vector Parallel compiler), C/VP (C Vector compiler), HPF, C, C++
vendor web-site:
year-of-introduction:

Model	VPP5000U	VPP5000
Clock cycle	300 MHz	300 MHz
Processor performance	9.6 Gflop/s	9.6 Gflop/s
Peak performance	9.6 Gflop/s	1.22 Tflop/s
Maximum system memory	16 Gbyte	2 Tbyte
Number of processors	1 - 1	4 - 128

Section 1.33.1

Remarks

The VPP5000 is the sucessor of the former VPP700/VPP700E systems (with E for extended, i.e., the clock cycle 6.6 instead of 7 ns). The overall architectural changes with respect to the VPP700 series are slight. The clock cycle has been halved and the floating-point vectorpipes are able to deliver floating multiply-add results. With a replication factor of 16 for these vectorpipes, 32 floating-point results per clock cycle can be generated, at least in theory. In this way a four-fold increase in speed per

processor can be attained with respect to the VPP700E.

The architecture of the VPP5000 nodes is almost identical to that of the VPP700: Each node, called a Processing Element (PE) in the system is a powerful (9.6 Gflop/s peak speed with a 3.3 ns clock) vector processor in its own right. The vector processor is complemented by a RISC scalar processor with a peak speed of 1.2 Gflop/s. The scalar instruction format is 64 bits wide and may cause the execution of up to 4 operations in parallel. Each PE has a memory of up to 16 GB while a PE communicates with its fellow PEs at a point-to-point speed of 1.6 GB/s. This communication is taken care of by separate Data Transfer Units (DTUs). To enhance the communication efficiency, the DTU has various transfer modes like contiguous, stride, sub array, and indirect access. Also translation of logical to physical PE-ids and from Logical in-PE address to real address are handled by the DTUs. When synchronisation is required each PE can set its corresponding bit in the Synchronisation Register (SR). The value of the SR is broadcast to all PEs and synchronisation has occurred if the SR has all its bits set for the relevant PEs. This method is comparable to the use of synchronisation registers in shared-memory vector processors and much faster than synchronising via memory. The network is a direct crossbar which should lead to an excellent throughput of the network. This is in contrast to the VPP700 where a level-2 crossbar was employed for configurations larger than 16 processors. On special order 512 PE systems can be built by Fujitsu, quadrupling the maximum amount of memory and the theoretical peak performance.

The VPP5000U is one of the few single-processor vector processors that is offered. It is simply a single-processor version of the VPP5000, of course without the network and data transfer extentions that are required in the VPP5000.

The Fortran compiler that comes with the VPP5000 has extensions that enable data decomposition by compiler directives. This evades in many cases restructuring of the code. The directives are different from those as defined in the High Performance Fortran Proposal but it should be easy to adapt them. Furthermore, it is possible do define parallel regions, barriers, etc., via directives, while there are several intrinsic functions to enquire about the number of processors and to execute `POST/WAIT` commands. Furthermore, also a message passing programming style is possible by using the PVM or MPI communication libraries that are available.

Just like for the Fujitsu AP3000, no information via a web page is available anymore (unless perhaps in Japanese) since the restructuring of the Fujitsu web site.

Section 1.33.2

Measured Performances

The system was announced in November 1999 and some results are available by now. In Dong02 for a 100 processor system a speed of 886 Gflop/s was measured solving an order 195,600 full linear system which amounts to an efficiency of 90%. On a single processor a speed of 6.04 Gflop/s was measured in solving a system of order 2000. In evaluating a 10-th order polynomial a speed of 8.68 Gflop/s was observed, also an efficiency of over 90% (see EurB99 for both last results).

Section 1.33.3

TOP500 Systems

28	VPP5000/100	http://www.ecmwf.int/
45	VPP5000/80	http://www.tsukuba.ac.jp/
64	VPP5000/64	http://www.jaeri.go.jp/

65	VPP5000/64	http://www.kyushu-u.ac.jp/
90	VPP5000/56	http://www.nagoya-u.ac.jp/
92	VPP800/63	http://www.kyoto-u.ac.jp/
146	VPP700/160E	http://www.riken.go.jp/
167	VPP5000/32	http://criepi.denken.or.jp/
174	VPP5000/31	http://www.meteo.fr
178	VPP5000/30	http://www.ims.ac.jp/
239	VPP5000/25	http://www.cwb.gov.tw/
251	VPP700/116	http://www.ecmwf.int/
430	VPP5000/16	http://www.cea.fr/

Section 1.34

EnterTheGrid description of company

Section 1.35

The Fujitsu/Siemens PRIMEPOWER

machine-type: RISC-based shared-memory multi-processor.
operating-system: Solaris (Sun's Unix variant).
connection-structure: Crossbar.
compilers: Parallel Fortran 90, C, C++.
vendor web-site: http://primepower.fujitsu.com/en/index.html
year-of-introduction: 2000

Model	PRIMEPOWER 2000
Clock cycle	675 MHz
Processor performance	1.35 Gflop/s
Peak performance	173 Gflop/s
Maximum system memory	4 Tbyte
Number of processors	8 - 128

Section 1.35.1

Remarks

We only discuss here the PRIMEPOWER 2000 as the smaller models have the same structure but less processors (maximally 16 in the 800 model and 32 in the 1000 model). In many respects this machine is akin to the sun.html. The processors are 64-bit Fujitsu implementations of SUN's SPARC processors, called SPARC 64 GP processors and they are completely compatible with the SUN products. The processors are available in a 563 MHz and a 675 MHz variant. Also the interconnection of the processors in the PRIMEPOWER systems is like the one in the Fire 3800-15K: a crossbar that connects all processors at the same footing, i.e., it is

not a NUMA machine.

Unfortunately, there is no sound technical information available beyond the data sheets that are provided via Fujitsu's web site. These data sheets omit any information about the bandwidth of the interconnect be it point-to-point, bi-sectional, or aggregate. Judging from the available information the system is more positioned as a communication server than as a high performance computer while the structure is well suited for this kind of tasks.

Section 1.35.2

Measured Performances

Dongarra reports in Dong02 a performance of 118 Gflop/s out of a maximum of 172.8 Gflop/s for solving a system of order 116,480. This amounts to an efficiency of 68.3%.

Section 1.35.3

TOP500 Systems

Section 1.36

EnterTheGrid description of company

Section 1.37

The SGI Origin 3000 series

machine-type: RISC-based distributed-memory multi-processor
operating-system: IRIX (SGI's Unix variant)
connection-structure: Crossbar, hypercube (see remarks)
compilers: Fortran 77, Fortran 90, C, C++ , ADA, Pascal
vendor web-site: http://www.sgi.com/servers/
year-of-introduction: 2000

Model	Origin 3400	Origin 3800
Clock cycle	600 MHz	600 MHz
Processor performance	1.2 Gflop/s	1.2 Gflop/s
Peak performance	38.3 Gflop/s	614 Gflop/s
Maximum system memory	64 Gbyte	1 Tbyte
Number of processors	4 - 32	6 - 512

Section 1.37.1

Remarks

By July 2000 has passed from its Origin2000 series to its new Origin3000 series comprised of the Origin3200, Origin3400, and Origin3800 models. In the system parameter list above we only included the 3400 and 3800 models because of their peak performance. Many of the characteristics of the Origin2000 have been retained of which the most important is its ccNUMA character. The processor used is presently the MIPS R14000, a direct successor of the R12000s in the Origin2000 systems. The R14000 is very similar to the R12000 processor, be it that the primary cache is at full speed where the R12000 operated at 2/3 speed. In the newest systems the 600 MHz R14000A is offered, although also R14000 500 MHz-based systems are still available.

SGI has further modularised the Origin3000 in comparison with its predecessor. A system contains so-called C-bricks, CPU boards with 2-4 processors and a router chip connecting the on-board memory with the processors, to router boards called R-bricks for communication with the rest of the system, and to I-bricks that contain disks, PCI expansion slots, etc. and that together make up the I/O sub-system of the machine. The basic hardware bandwidth within a C-brick is 1.6 GB/s from the router chip to one pair of CPUs, 3.2 GB/s from memory to the router chip (2 x 1.6 GB/s full duplex). The same bandwidth is available for inter-node communication. The off-board I/O bandwidth is 2.4 GB/s (2 x 1.2 GB/s full duplex). The R-brick can be connected to 16 C-bricks and it has 8 ports to connect it to other R-bricks. So, 128 C-bricks or 512 processors can maximally be interconnected in this way.

The machine is a typical representative of the ccNUMA class of systems. The memory is physically distributed over the node boards but there is one system image. Because of the structure of the system, the bi-sectional bandwidth of the system remains constant from 8 processors on: 210 GB/s. This is a large improvement over the earlier Origin2000 systems where the bi-sectional bandwidth was 82 GB/s.

Parallelisation is done either automatically by the (Fortran or C) compiler or explicitly by the user, mainly through the use of directives. All synchronisation, etc., has to be done via memory. This may cause potentially a fairly large parallelisation overhead. Also a message passing model is allowed on the Origin using the optimised SGI versions of PVM and MPI, and the SGI/Cray-specific `shmem` library. Programs implemented in this way will possibly run very efficiently on the system.

A nice feature of the Origins is that it may migrate processes to nodes that should satisfy the data requests of these processes. So, the overhead involved in transferring data across the machine are minimised in this way. The technique is reminiscent of the late Kendall Square Systems although in these systems the data were moved to the active process. SGI claims that the time for non-local memory references is on average about 2 times longer than for local memory references, an improvement of 50% over the Origin2000 series.

Section 1.37.2

Measured Performances

As yet no performance figures for the 600 MHz-based systems are available but in Dong02 the performance of the solution of a linear system of order 230,000 is quoted for a 512 processor system with 500 MHz processors. In this case a speed of 405.6 Gflop/s was found, an efficiency of 79%.

Section 1.37.3

TOP500 Systems

15	ASCI Blue Mountain	http://www.lanl.gov/
51	ORIGIN 2000 250 MHz	http://www.lanl.gov/
115	ORIGIN 3000 600 MHz	http://www.nas.nasa.gov/
116	ORIGIN 3000 600 MHz	http://www.gfdl.gov
117	ORIGIN 3000 600 MHz	http://www.ntnu.no/
118	ORIGIN 3000 500 MHz	http://www.fnmoc.navy.mil
119	ORIGIN 3000 500 MHz	
120	ORIGIN 3000 500 MHz	http://www.jaeri.go.jp/
121	ORIGIN 3000 500 MHz	http://www.scl.kyoto-u.ac.jp/
122	ORIGIN 3000 500 MHz	http://www.sara.nl/
123	ORIGIN 3000 500 MHz	http://www.sara.nl/
147	ORIGIN 3000 400 MHz	http://www.csar.cfs.ac.uk/
148	ORIGIN 3000 400 MHz	http://www.wes.hpc.mil/
149	ORIGIN 3000 400 MHz	
150	ORIGIN 3000 400 MHz	http://www.nas.nasa.gov/
151	ORIGIN 3000 400 MHz	http://www.gsfc.nasa.gov/
152	ORIGIN 3000 400 MHz	http://www.sgi.com/
153	ORIGIN 3000 400 MHz	http://www.arl.mil/
160	ORIGIN 3000 500 MHz	http://www.bosai.go.jp/
161	ORIGIN 3000 500 MHz	http://www.ntnu.no/
165	ORIGIN 2000 400 MHz	http://www.nas.nasa.gov/
180	ORIGIN 2000 195/250 MHz	http://www.ncsa.uiuc.edu/
185	ORIGIN 3000 500 MHz	http://www.cines.fr
224	ORIGIN 2000 300 MHz	http://www.ifs.tohoku.ac.jp/
225	ORIGIN 2000 300 MHz	http://www.nas.nasa.gov/
256	ORIGIN 3000 600 MHz	http://www.gfdl.gov
257	ORIGIN 3000 600 MHz	http://www.gfdl.gov
258	ORIGIN 3000 500 MHz	
259	ORIGIN 3000 500 MHz	http://www.jaeri.go.jp/
260	ORIGIN 3000 500 MHz	http://www.scl.kyoto-u.ac.jp/
261	ORIGIN 3000 500 MHz	
396	ORIGIN 3000 400 MHz	
397	ORIGIN 3000 400 MHz	http://www.sgi.com/
398	ORIGIN 3000 400 MHz	http://www.scripps.edu/
417	ORIGIN 3000 500 MHz	http://www.ford.com/
418	ORIGIN 3000 500 MHz	
419	ORIGIN 3000 500 MHz	http://www.msi.umn.edu/
423	ORIGIN 2000 400 MHz	http://www.cc.titech.ac.jp/
424	ORIGIN 2000 400 MHz	http://www.ims.ac.jp

Section 1.38

EnterTheGrid description of company

SGI

SGI is active in high-performance computing technology. The company's systems, range from desktop workstations and servers to supercomputers in the world. The company is also very strong in 3D visualisation. The products are marketed into scientific, engineering, and creative professionals and large enterprises.

SGI Origin 3000 servers are scalable from 2-512 processors and can have up to 1 Tbyteof memory.

The SGI Onyx 3000 is designed from the ground up to support immersive visualisation, high performance computing and complex data management. The family is available in a range of configurations for every level of visual supercomputing. It offers two different graphics subsystems, InfinitePerformance graphics or InfiniteReality3 graphics.

1. ***U.S. Army Corps of Engineers*** uses 512-processor SGI Origin 3800 supercomputer to simulate the September 11 attack on the Pentagon.
2. ***Ford Motor Company*** increases quality and safety with SGI Origin 3000 Technology.
3. ***Lockheed Martin and Pratt and Whitney***: SGI advanced visualisation and high-performance computing (HPC) technologies are instrumental in helping both Lockheed Martin and Pratt and Whitney land the Joint Strike Fighter contract.
4. ***Fleet Numerical Meteorology and Oceanography Center***: the U.S. Navy uses SGI Origin 3000 series servers for advanced weather simulation and storage capacities.
5. ***Medtronic*** uses SGI technology for innovations in life-saving medical devices. SGI Origin 3000 series systems are chosen by Medtronic for computer modelling and simulation to help speed the design and develpment of new therapies for heart conditions, heart failure and various cardiovascular diseases.

http://www.sgi.com

Section 1.39

Summary Table

System	Clock-cycle	Peak performance	Max memory	Max processors	HPF available?
The Fujitsu AP3000	300 MHz	614 Gflop/s	2 Tbyte	1024	x
The C-DAC Param 10000 OpenFrame	400 MHz	- Mflop/s	1 Gbyte		
The NEC Cenju-4	5 ns	410 Gflop/s	512 Gbyte	1024	x
The Compaq AlphaServer SC	1 GHz	8 Tflop/s	8 Tbyte	4096	x
The Compaq GS series	1 GHz	64 Gflop/s	156 Gbyte	32	x
The Cray SX-6					
The Cambridge Parallel Processing Gamma II Plus	30 MHz	2.4 Gflop/s	512 MB	4096	
The Cray MTA		192 Gflops	1 Tbyte	256	

IBM eServer p690	1.3 GHz	166.4 Gflop/s	128 Tbyte	16384	x
The Quadrics Apemille	267 MHz	1 Tflop/s	64 Gbyte	2048	
The Hitachi SR8000	450 MHz	7.3 Tflop/s	8 Tbyte	512	x
The Sun Fire 3800-15K	900 MHz	190.8 Gflop/s	576 Gbyte	106	x
The HP 9000 SuperDome	750 MHz	192 Gflop/s	128 Gbyte	64	x
The Cray SV1	500 MHz	256 Gflop/s	384 Gbyte	128	
The NEC SX-6	500 MHz		8 TB	1024	x
The Cray T3E	675 MHz	2938 Gflop/s	1 Tbyte	2176	x
The Fujitsu VPP5000 series	300 MHz	1.22 Tflop/s	2 Tbyte	128	x
The Fujitsu/Siemens PRIMEPOWER	675 MHz	173 Gflop/s	4 Tbyte	128	
The SGI Origin 3000 series	600 MHz	614 Gflop/s	1 Tbyte	512	

Chapter 4
Systems disappeared from the list

As already stated in the introduction the list of systems is not complete. On one hand this is caused by the sheer number of systems that are presented to the market and are often very similar to systems described above (for instance, the Volvox system not listed is very similar but not equivalent to the listed C-DAC system and there are numerous other examples). On the other hand there many systems that are still in operation around the world, often in considerable quantities that for other reasons are excluded. The most important reasons are:

- The system is not marketed anymore. This is generally for one of two reasons:
  - The manufacturer is out of business.
  - The manufacturer has replaced the system by a newer model of the same type or even of a different type.
- The system has become technologically obsolete in comparison to others of the same type. Therefore, listing them is not sensible anymore.

Below we present a table of systems that fall into one of the categories mentioned above. We think this may have some sense to those who come across machines that are still around but are not the latest in their fields. It may be interesting at least to have an indication how such systems compare to the newest ones and to place them in context.

It is good to realise that although systems have disappeared from the section above they still may exist and are actually sold. However, their removal stems in such cases mainly from the fact that they are not serious candidates for high-performance computing anymore.

The table is, again, not complete and admittedly somewhat arbitrary. The data are in a highly condensed form: the system name, system type, theoretical maximum performance of a fully configured system, and the reason for their disappearance is given. The arbitrariness lies partly in the decision which systems are still sufficiently of interest to include and which are not.

We include the year of introduction and the year of exit of the systems when they were readily accessible. These timespans could give a hint of the dynamics that governs this very dynamical branch of the the computer industry.

The average 'age' of a supercomputer system architecture is: 3.4 year.

name: The Alex AVX 2
year-of-introduction: 1992
year-of-exit: 1997
machine-type: RISC-based distributed-memory multi-processor.
Theoretical peak performance: 3.84 gflops
Reason for dissapearance: System is obsolete, there is no new system planned.

name: Alliant FX/2800
year-of-introduction: 1989
year-of-exit: 1992
machine-type: Shared memory vector-parallel, max. 28 processors.
Theoretical peak performance: 1120 mflops
Reason for dissapearance: Manufacturer out of business.

name: Avalon A12
year-of-introduction: 1996
year-of-exit: 2000
machine-type: RISC-based distributed memory multi-processor, max. 1680 processors.
Theoretical peak performance: 1.3 tflops
Reason for dissapearance: Avalon is not in business anymore.

name: The AxilSCC
year-of-introduction: 1996
year-of-exit: 1997
machine-type: RISC-based distributed-memory system, max. 512 processors.
Theoretical peak performance: 76.8 gflops
Reason for dissapearance: System is not marketed anymore by Axil.

name: BBN TC2000
year-of-introduction: -
year-of-exit: 1990
machine-type: Virtual shared memory parallel, max. 512 processors.
Theoretical peak performance: 1 gflops
Reason for dissapearance: Manufacturer has discontinued marketing parallel computer systems.

name: Cambridge Parallel Processing DAP Gamma
year-of-introduction: 1986
year-of-exit: 1995
machine-type: Distributed memory processor array system.
Theoretical peak performance: 1.6 gflops
Reason for dissapearance: Peak performance for 32-bit. Replaced by newer gamma-II.html#gamma-II series.

name: C-DAC PARAM 9000/SS
year-of-introduction: 1995
year-of-exit: 1997
machine-type: RISC-based distributed-memory system, max. 200 processors.
Theoretical peak performance: 12.0 gflops
Reason for dissapearance: replaced by newer OpenFrame series (see below).

name: C-DAC PARAM Openframe serie
year-of-introduction: 1996
year-of-exit: 1999
machine-type: RISC-based distributed-memory system, max. 1024 processors.
Theoretical peak performance: Unspecified. gflops
Reason for dissapearance: The system is not actively marketed anymore by C-DAC.

name: Convex SPP-1000/1200/1600
year-of-introduction: 1995
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 128 processors.
Theoretical peak performance: 25.6 gflops
Reason for dissapearance: replaced by newer superdome.html#superdome series.

name: Cray Computer Corporation Cray-2
year-of-introduction: 1982
year-of-exit: 1992
machine-type: Shared memory vector-parallel, max. 4 processors.
Theoretical peak performance: 1.95 gflops
Reason for dissapearance: Manufacturer out of business.

name: Cray Computer Corporation Cray-3
year-of-introduction: 1993
year-of-exit: 1995
machine-type: Shared memory vector-parallel, max. 16 processors.
Theoretical peak performance: 16 gflops
Reason for dissapearance: Manufacturer out of business.

name: Cray Research Inc. APP
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Shared memory RISC based system, max. 84 processors.
Theoretical peak performance: 6.7 gflops
Reason for dissapearance: Product line discontinued, gap was expected to be filled by the Cray J90 (see below).

name: Cray T3D
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 2048 processors.
Theoretical peak performance: 307 gflops
Reason for dissapearance: replaced by newer t3e.html#t3e.

name: Cray T3E Classic
year-of-introduction: 1996
year-of-exit: 1997
machine-type: Distributed memory RISC based system, max. 2048 processors.
Theoretical peak performance: 1228 gflops
Reason for dissapearance: replaced by Cray T3Es with faster clock. t3e.html#t3e

name: Cray J90
year-of-introduction: 1994
year-of-exit: 1998
machine-type: Shared memory vector-parallel, max. 32 processors.
Theoretical peak performance: 6.4 gflops
Reason for dissapearance: replaced by newer sv1.html#sv1.

name: Cray Research Inc. Cray Y-MP, Cray Y-MP M90
year-of-introduction: 1989
year-of-exit: 1994
machine-type: Shared memory vector-parallel, max. 8 processors.
Theoretical peak performance: 2.6 gflops
Reason for dissapearance: replaced by newer C90 (see below).

name: Cray Y-MP C90
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Shared memory vector-parallel, max. 16 processors.
Theoretical peak performance: 16 gflops
Reason for dissapearance: replaced by newer T90 (see below).

name: Cray T90
year-of-introduction: 1995
year-of-exit: 1998
machine-type: Shared memory vector-parallel, max. 32 processors.
Theoretical peak performance: 58 gflops
Reason for dissapearance: replaced by newer sv1.html#sv1.

name: Digital Equipment Corp. Alpha farm
year-of-introduction: -
year-of-exit: 1994
machine-type: Distributed memory RISC based system, max. 4 processors.

Theoretical peak performance: 0.8 gflops
Reason for dissapearance: replaced by newer compaqgs.html#compaqgs.

name: Digital Equipment Corp. Alpha AlphaServer 8200 & 8400
year-of-introduction: -
year-of-exit: 1998
machine-type: Distributed memory RISC based system, max. 6 processors (AlphaServer 8200) or 14 (AlphaServer 8400).
Theoretical peak performance: 17.2 gflops
Reason for dissapearance: Peak 8200: 7.3 Gflop/s. replaced by newer compaqsc.html#compaqsc.

name: Fujitsu AP1000
year-of-introduction: 1991
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 1024 processors.
Theoretical peak performance: 5 gflops
Reason for dissapearance: replaced by the ap3000.html#ap3000.

name: Fujitsu VPP500 series
year-of-introduction: 1993
year-of-exit: 1995
machine-type: Distributed memory multi-processor vectorprocessors, max. 222 processors.
Theoretical peak performance: 355 gflops
Reason for dissapearance: replaced by the VPP300/700 series (see below).

name: Fujitsu VPP300/700 series
year-of-introduction: 1995
year-of-exit: 1999
machine-type: Distributed memory multi-processor vectorprocessors, max. 256 processors.
Theoretical peak performance: 614 gflops
Reason for dissapearance: replaced by the vpp5000.html#vpp5000

name: Fujitsu VPX200 series
year-of-introduction: -
year-of-exit: 1995
machine-type: Single-processor vectorprocessors.
Theoretical peak performance: 5 gflops
Reason for dissapearance: replaced by the VPP300/700 series (see above).

name: Hitachi S-3800 series
year-of-introduction: 1993
year-of-exit: 1998
machine-type: Shared-memory multi-processor vectorprocessors, max. 4 processors.
Theoretical peak performance: 32 gflops
Reason for dissapearance: Replaced by the newer sr8000.html#sr8000 system.

name: Hitachi S-3600 series
year-of-introduction: 1994
year-of-exit: 1999
machine-type: Single-processor vectorprocessor.
Theoretical peak performance: 2 gflops
Reason for dissapearance: Replaced by the newer sr8000.html#sr8000 system.

name: Hitachi SR2001 series
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 128 processors.
Theoretical peak performance: 23 mflops

Reason for dissapearance: Replaced by the successor SR2201 (see below).

name: Hitachi SR2201 series
year-of-introduction: 1996
year-of-exit: 1998
machine-type: Distributed memory RISC based system, max. 1024 processors.
Theoretical peak performance: 307 mflops
Reason for dissapearance: Replaced by the newer sr8000.html#sr8000.

name: HP/Convex C4600
year-of-introduction: 1994
year-of-exit: 1997
machine-type: Shared memory vector-parallel, max. 4 processors (C4640).
Theoretical peak performance: 3.2 mflops
Reason for dissapearance: The C4600 is not marketed by HP/Convex anymore.

name: The HP Exemplar V2600
year-of-introduction: 1999
year-of-exit: 2000
machine-type: Distributed-memory RISC based system, max. 128 processors.
Theoretical peak performance: 291 mflops
Reason for dissapearance: Replaced by the superdome.html#superdome.

name: IBM ES/9000 series
year-of-introduction: 1991
year-of-exit: 1994
machine-type: Shared memory vector-parallel system, max. 6 processors.
Theoretical peak performance: 2.67 gflops
Reason for dissapearance: IBM does not pursue high-performance computing by this product line anymore.

name: IBM SP1 series
year-of-introduction: 1992
year-of-exit: 1994
machine-type: Distributed memory RISC based system, max. 64 processors.
Theoretical peak performance: 8 gflops
Reason for dissapearance: Replaced by the newer sp.html#sp.

name: Intel Paragon XP
year-of-introduction: 1992
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 4000 processors.
Theoretical peak performance: 300 gflops
Reason for dissapearance: Except for a non-commercial research system (the ASCI Option Red system at Sandia National Labs.) Intel is not in the business of high-performance computing anymore.

name: Kendall Square Research KSR2
year-of-introduction: 1992
year-of-exit: 1994
machine-type: Virtually shared memory parallel, max. 1088 processors.
Theoretical peak performance: 400 gflops
Reason for dissapearance: Kendall Square has terminated its business.

name: Kongsberg Informasjonskontroll SCALI
year-of-introduction: 1996
year-of-exit: 1997
machine-type: Distributed memory RISC based system, max. 512 processors.
Theoretical peak performance: 76.8 gflops
Reason for dissapearance: Kongsberg does not market the system anymore.

name: MasPar MP-1, MP-2
year-of-introduction: 1991
year-of-exit: 1996
machine-type: Distributed memory processor array system, max. 16384 processors.
Theoretical peak performance: 2.4 Gflop/s (64-bit, MP-2) gflops
Reason for dissapearance: Systems are not marketed anymore.

name: Matsushita ADENART
year-of-introduction: 1991
year-of-exit: 1997
machine-type: Distributed memory RISC based system, 256 processors.
Theoretical peak performance: 2.56 gflops
Reason for dissapearance: Machine is obsolete and no new systems are developed in this line.

name: Meiko CS-1 series
year-of-introduction: 1989
year-of-exit: 1995
machine-type: Distributed memory RISC based system.
Theoretical peak performance: 80 Mflop/s per processor mflops
Reason for dissapearance: Meiko does not build complete systems anymore (but see below).

name: Machine: Meiko CS-2 series
year-of-introduction: 1994
year-of-exit: 1999
machine-type: Distributed memory RISC based system, max. 1024 processors. 200 Mflop/s per processor
Theoretical peak performance: 204.8 gflops
Reason for dissapearance: Quadrics Supercomputers World Ltd. does not market the system anymore. The updated network technology is now offered for other systems like compaqsc.html#compaqsc.

name: nCUBE 2S
year-of-introduction: 1993
year-of-exit: 1998
machine-type: Distributed memory system, max. 8192 processors.
Theoretical peak performance: 19.7 gflops
Reason for dissapearance: NCUBE has withdrawn from the scientific and technical market. The nCUBE2S is now offered as a parallel multimedia server.

name: nCUBE 3
year-of-introduction: -
year-of-exit: -
machine-type: Distributed memory system, max. 10244 processors.
Theoretical peak performance: 1 tflops
Reason for dissapearance: Was announced several times but was never finished. See remarks at nCUBE 2S

name: NEC Cenju-3
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Distributed-memory system, max. 256 processors.
Theoretical peak performance: 12.8 gflops
Reason for dissapearance: replaced by newer Cenju-4 series (see below).

name: NEC Cenju-4
year-of-introduction: 1998
year-of-exit: 2002
machine-type: Distributed-memory system, max. 1024 processors.

Theoretical peak performance: 410 gflops
Reason for dissapearance: NEC has withdrawn this machine in favour of a possible successor. Specifics are not known, however.

name: NEC SX-3R
year-of-introduction: 1993
year-of-exit: 1996
machine-type: Shared memory multi-processor vector processors, max. 4 processors.
Theoretical peak performance: 25.6 gflops
Reason for dissapearance: replaced by newer SX-4 series (see below).

name: NEC SX-4
year-of-introduction: 1995
year-of-exit: 1998
machine-type: Shared memory multi-processor vector processors, max. 256 processors.
Theoretical peak performance: 1 tflops
Reason for dissapearance: replaced by newer SX-5 series (see below).

name: NEC SX-5
year-of-introduction: 1998
year-of-exit: 2002
machine-type: Shared memory multi-processor vector processors, max. 512 processors.
Theoretical peak performance: 5.12 tflops
Reason for dissapearance: replaced by newer sx-6.html#sx-6 series.

name: Parsys SN9000 series
year-of-introduction: 1993
year-of-exit: 1995
machine-type: Distributed memory RISC based system, max. 2048 processors.
Theoretical peak performance: 51.2 Gflop/s gflops
Reason for dissapearance: Replaced by the newer TA9000 (but see below).

name: Parsys TA9000 series
year-of-introduction: 1995
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 512 processors.
Theoretical peak performance: 119.3 gflops
Reason for dissapearance: Parsys does not offer complete system anymore. Instead it sells node cards based on the TA9000 for embedded systems.

name: Parsytec GC/Power Plus
year-of-introduction: 1993
year-of-exit: 1996
machine-type: Distributed memory RISC based system.
Theoretical peak performance: 266.6 mflops
Reason for dissapearance: Peak performance per processor. System has been replaced by the Parsytec CC systems (see below).

name: Parsytec CC series
year-of-introduction: 1996
year-of-exit: 1998
machine-type: Distributed memory RISC based system.
Theoretical peak performance: unspecified. mflops
Reason for dissapearance: Vendor has withdrawn from the High-Performance computing market.

name: Siemens-Nixdorf VP2600 series
year-of-introduction: -

year-of-exit: 1995
machine-type: Single-processor vectorprocessors.
Theoretical peak performance: 5 gflops
Reason for dissapearance: eventually replaced by the vpp5000.html#vp5000 series.

name: Silicon Graphics PowerChallenge
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Shared memory multi-processor, max. 36 processors.
Theoretical peak performance: 14.4 gflops
Reason for dissapearance: replaced by the SGI Origin 2000 (see below).

name: SGI Origin 2000
year-of-introduction: 1996
year-of-exit: 2000
machine-type: Shared memory multi-processor, max. 128 processors.
Theoretical peak performance: 102.4 gflops
Reason for dissapearance: replaced by the SGI origin.html#origin.

name: Machine: Stern Computing Systems SSP
year-of-introduction: 1994
year-of-exit: 1996
machine-type: Shared memory multi-processor, max. 6 processors.
Theoretical peak performance: 2 gflops
Reason for dissapearance: Vendor terminated its business just before delivering first systems.

name: SUN E10000 Starfire
year-of-introduction: 1997
year-of-exit: 2001
machine-type: Shared memory multi-processor, max. 64 processors.
Theoretical peak performance: 51.2 gflops
Reason for dissapearance: replaced by the SUN Fire 3800-15K sun.html#sun.

name: Thinking Machine Corporation CM-2(00)
year-of-introduction: 1987
year-of-exit: 1991
machine-type: SIMD parallel machine with hypercube structure, max. 64K processors.
Theoretical peak performance: 31 gflops
Reason for dissapearance: was replaced by the newer CM-5 (but see below).

name: Thinking Machine Corporation CM-5
year-of-introduction: 1991
year-of-exit: 1996
machine-type: Distributed memory RISC based system, max. 16K processors.
Theoretical peak performance: 2 tflops
Reason for dissapearance: Thinking Machine Corporation has stopped manufacturing hardware and hopes to keep alive as a software vendor.

Chapter 5
Systems under development

Although we mainly want to discuss real, marketable systems and no experimental, special purpose, or even speculative machines, we want to include a section on systems that are in a far stage of development and have a fair chance of reaching the market. For inclusion in section 3 we set the rule that the system described there should be on the market within a period of 6 months from announcement. The systems described in this section will in all probability appear within one year from the publication of this report. However, there are vendors who do not want to disclose any specific data on their new machines until they are actually beginning to ship them. We recognise the wishes of such vendors (it is generally wise not to stretch the expectation of potential customers too long) and they will not disclose such information.

Below we discuss systems that may lead to commercial systems to be introduced on the market between somewhat more than half a year to a year from now. The commercial systems that result from it will sometimes deviate significantly from the original research models depending on the way the development is done (the approaches in Japan and the USA differ considerably in this respect) and the user group which is targeted.

A development that may be of significance in the near future is the introduction of Intel's IA-64 Itanium processor family. The first chip in the family has recently been succeeded by the second generation, the Itanium 2initially with a clock frequencies of 1 GHz. It is highly probable that a majority of vendors will incorporate IA-64 chips in favour of their proprietary RISC processors in a time span of 1--2 years. Understandable as this may be from an economical point of view, it is also slightly disturbing as the processor landscape may become rather barren in this way.

Section 1

Compaq

Compaq has now become a part of Hewlett-Packard and it is hard to say whether the now successful AlphaServer SC line will be continued in some way. It seems fairly sure that an EV7-based system (see the section on the ev7.html) will be marketed in the near future. At the same time itis almost as sure that no system with the projected dual core Alpha EV8 will be built. Already before Compaq joined with HP it committed itself to using IA-64 processors in future systems and this will undoubtly be pursued as a part of HP as this company was one of the main developers of the latter processor line. As of the macro structure of the future systems nothing can be sure, although the present interconnect technology is very successful and for that reason may be maintained at least for another generation.

Section 2

Cray Inc.

In the beginning of 2003 the next generation vector processor from Cray Inc., the

SV-2, should be ready to ship. It builds on the technology found in the present Cray SV-1s, but the speed per processor should be appreciably higher: 12.8 Gflop/s. Up to 4 processors will be fitted in a node that also harbours a maximum of 32 GB of memory per CPU. As many as 16 nodes can be put into one frame. Up to 64 frames can connected having a Single System Image. The inter-node communication speed is projected to be 100 GB/s. If these design targets can be achieved, the SV-2 would be a formidable system and also a testimony that vector processing is not a dead end in computer technology as except Cray and NEC all other vendors seem to have abandoned the concept (in the sr8000.html series pseudo-vector processing is implemented. However, the bandwidth to/from memory is markedly lower than what normally is expected in vectorprocessors).

Section 3

Hewlett-Packard

Because of the merger with Compaq it is not clear what the future course of HP at the high end will be. The present SuperDome does not belong to the very high-end systems and HP had no clear plans for making such systems in the near future. A logical decision would be to build upon the AlphaServer SC for this line of systems but as no strategy on this part is known presently, very little can be assumed except that the generation after the next will be based on the Intel/HP IA-64 processor.

Section 4

IBM

Since end 2001/begin 2002 IBM has systems available based on the POWER4 processor (see the p690.html description). Presently, the most extensive system built with 32 processor Turbo nodes would be able to attain a Theoretical Peak Performance of slightly in excess of 8 Tflop/s. There is still a way to go to the goal of building a 100 Tflop/s system as IBM ventures to make in a follow-on contract in the ASCI program.

One can expect that the integration level of the nodes will further increase and also the number of nodes may be increased beyond the 512 that now can be offered on special delivery. Assuming a doubling both of the clock frequency and the number of CPUs/node and extending the number of nodes that can be coupled by a factor of 4, the 100 Tflop/s boundary could be passed in about 2 years. With the increase of processor speed a matching increase of interconnect speed is in order. The High-Performance switch that presently is used for node interconnection has a bandwidth of 500 MB/s. This will be upgraded to speed of about 1 GB/s within the next two years and should be increased even more to be useful in a 100 Tflop/s system for general applications.

Section 5

SGI

In the SGI Origin3000 systems already a provision has been made in the C-bricks to put in Intel's IA-64 processors instead of the MIPS R14000 (see \ref{origin}. Though SGI projects that at least still one MIPS chip generation, the R16000, will (can) be used in Origin systems. At the moment SGI seems directed to making systems with a high "compute density", i.e., to integrate as many processors as possible in a smallest

possible volume. In this respect the MIPS processors have a very good track record. The R14000A, the processor employed at this moment dissipate a factor 4 to 5 less energy than the Alpha EV68 or the IBM POWER4. This should allow for building quite dense systems without running into cooling problems. Whether this is a sufficient argument in view of the relatively low clock frequency of the MIPS processor, remains to be seen. The very modular structure of the Origin systems is an architectural asset that makes it probable that the system structure will not change significantly in the near future.

Still, SGI has publicly committed itself to making systems using IA-64 processors. It will depend on the availability/price of the Itanium 2 or its successor whether such systems will be of interest, also because of the much larger energy requirements. If IA-64 based machines will be marketed, they will use Linux for an operating system as SGI already a few years ago discontinued porting its native IRIX OS to the IA-64 platform because of cost considerations.

Section 6

SRC

The SRC company represents a trend that is taking up remarkable speed at the moment. It consists of complementing general purpose processors with (a collection of) FPGAs, ***Field Programmable Gate Arrays***, see also the glossary.html. This makes it possible to configure such a machine for special user-defined tasks that would them make, at least in principle, significantly faster than general purpose processors for the same tasks. SRC proposes a system, the SRC-6, with 256 dual processor boards containing standard IA-32 processor each of which is connected to a unit, MAP, for Multi-Adaptive Processor, that consists of an FPGA, private memory, a MAP controller, and logic to reconfigure the unit when needed. MAPs are interconnected by a ring network and the standard processor boards that also have their local memory, are through the MAPs connected to a global memory by a read and a write crossbar.

Part 2
Back Matter

Glossary

Architectural class

Classification of computer systems according to its architecture: e.g., distributed memory MIMD computer, symmetric multi processor (SMP), etc. See this glossary and section architecture for the description of the various classes.

Architecture

The internal structure of a computer system or a chip that determines its operational functionality and performance.

ASCI

Accelerated Strategic Computer Initiative. A massive funding project in the USA concerning research and production of high-performance systems. The main motivation is said to be the management of the USA nuclear stockpile by computational modeling instead of actual testing. ASCI has greatly influenced the development of high-performance systems in a single direction: clusters of SMP systems.

Bank cycle time

The time needed by a (cache-)memory bank to recover from a data access request to that bank. Within the bank cycle time no other requests can be accepted.

Beowulf cluster

Cluster of PCs or workstations with a private network to connect them. Initially the name was used for do-it-yourself collections of PCs mostly connected by Ethernet and running Linux to have a cheap alternative for "integrated" parallel machines. Presently, the definition is wider including high-speed switched networks, fast RISC-based processors and complete vendor-preconfigured rack-mounted systems with either Linux or Windows as an operating system.

Bit-serial

The operation on data on a bit-by-bit basis rather than on byte or 4/8-byte data entities in parallel. Bit-serial operation is done in processor array machines where for signal and image processing this mode is advantageous.

Cache --- data, instruction

Small, fast memory close to the CPU that can hold a part of the data or instructions to be processed. The primary or level 1 caches are virtually always located on the same chip as the CPU and are divided in a cache for instructions and one for data. A secondary or level 2 cache is mostly located off-chip and holds both data and instructions. Caches are put into the system to hide the large latency that occurs when data have to be fetched from memory. By loading data and or instructions into the caches that are likely to be needed, this latency can be significantly reduced.

Capability computing

A type of large-scale computing in which one wants to accommodate very large and time consuming computing tasks. This requires that parallel machines or clusters are managed with the highest priority for this type of computing possibly with the consequence that the computing resources in the system are not always used with the greatest efficiency.

Capacity computing

A type of large-scale computing in which one wants to use the system (cluster) with the highest possible throughput capacity using the machine resources as efficient as possible. This may have adverse effects on the performance of individual computing tasks while optimising the overall usage of the system.

ccNUMA

Cache Coherent Non-Uniform Memory Access. Machines that support this type of memory access have a physically distributed memory but logically it is shared. Because of the physical difference of the location of the data items, a data request may take a varying amount of time depending on the location of the data. As both the memory parts and the caches in such systems are distributed a mechanism is necessary to keep the data consistent system-wide. There are various techniques to enforce this (directory memory, snoopy bus protocol). When one of these techniques is implemented the system is said to be cache coherent.

Clock cycle

Fundamental time unit of a computer. Every operation executed by the computer takes at least one and possibly multiple cycles. Typically, the clock cycle is now in the order of one to a few nanoseconds.

Clock frequency

Reciproke of the clock cycle: the number of cycles per second expressed in Hertz (Hz). Typical clock frequencies nowadays are 400 MHz--1 GHz.

Control processor

The processor in a processor array machine that issues the instructions to be executed by all the processors in the processor array. Alternatively, the control processor may perform tasks in which the processors in the array are not involved, e.g., I/O operations or serial operations.

Crossbar (multistage)

A network in which all input ports are directly connected to all output ports without interference from messages from other ports. In a one-stage crossbar this has the effect that for instance all memory modules in a computer system are directly coupled to all CPUs. This is often the case in multi-CPU vector systems. In multistage crossbar networks the output ports of one crossbar module are coupled with the input ports of other crossbar modules. In this way one is able to build networks that grow with logarithmic complexity, thus reducing the cost of a large network.

Distributed Memory (DM)

Architectural class of machines in which the memory of the system is distributed over the nodes in the system. Access to the data in the system has to be done via an interconnection network that connects the nodes and may be either explicit via message passing or implicit (either using HPF or automatically in a ccNUMA system).

Dual core chip

A chip that contains two CPUs and (possibly common) caches. Due to the progression of the integration level more devices can be fitted on a chip. In fact, IBM makes a dual core chip: the POWER4 and other vendors may follow in the near future.

EPIC

Explicitly Parallel Instruction Computing. This term is coined by Intel for its IA-64 chips and the Instruction Set that is defined for them. EPIC can be seen as Very Large Instruction Word computing with a few enhancements. The gist of it is that no dynamic instruction scheduling is performed as is done in RISC processors but rather that instruction scheduling and speculative execution of code is determined beforehand in the compilation stage of a program. This simplifies the chip design while potentially many instructions can be executed in parallel.

Fat tree

A network that has the structure of a binary (quad) tree but that is modified such that near the root the available bandwidth is higher than near the leafs. This stems from the fact that often a root processor has to gather or broadcast data to all other processors and without this modification contention would occur near the root.

FPGA

FPGA stands for Field Programmable Gate Array. This is an array of logic gates that can be hardware-programmed to fulfill user-specified tasks. In this way one can devise special purpose functional units that may be very efficient for this limited task. As FPGAs can be reconfigured dynamically, be it only 100--1,000 times per second, it is theoretically possible to optimise them for more complex special tasks at speeds that are higher than what can be achieved with general purpose processors.

Functional unit

Unit in a CPU that is responsible for the execution of a predefined function, e.g., the loading of data in the primary cache or executing a floating-point addition.

Grid --- 2-D, 3-D

A network structure where the nodes are connected in a 2-D or 3-D grid layout. In virtually all cases the end points of the grid are again connected to the starting points thus forming a 2-D or 3-D torus.

HPF

High Performance Fortran. A compiler and run time system that enables to run Fortran programs on a distributed memory system as on a shared memory system. Data partition, processors layout, etc. are specified as comment directives that makes it possible to run the processor also serially. Present HPF available commercially allow only for simple partitioning schemes and all processors executing exactly the same code at the same time (on different data, so-called Single Program Multiple Data (SPMD) mode).

Hypercube

A network with logarithmic complexity which has the structure of a generalised cube: to obtain a hypercube of the next dimension one doubles the perimeter of the structure and connect their vertices with the original structure.

Instruction Set Architecture

The set of instructions that a CPU is designed to execute. The Instruction Set Architecture (ISA) represents the repertoire of instructions that the designers determined to be adequate for a certain CPU. Note that CPUs of different making may have the same ISA. For instance the AMD processors (purposely) implement the Intel IA-32 ISA on a processor with a different structure.

Memory bank

Part of (cache) memory that is addressed consecutively in the total set of memory banks, i.e., when data item *a(n)* is stored in bank *b*, data item *a(n+1)* is stored in bank *b+1*. (Cache) memory is divided in banks to evade the effects of the bank cycle time (see above). When data is stored or retrieved consecutively each bank has enough time to recover before the next request for that bank arrives.

Message passing

Style of parallel programming for distributed memory systems in which non-local data that is required explicitly must be transported to the processor(s) that need(s) it by appropriate send and receive messages.

MPI

A message passing library, Message Passing Interface, that implements the message passing style of programming. Presently MPI is the *de facto* standard for this kind of programming.

OpenMP

A shared memory parallel programming model in which shared memory systems and SMPs can be operated in parallel. The parallelisation is controlled by comment directives (in Fortran) or pragmas (in C and C++), so that the same programs also can be run unmodified on serial machines.

Pipelining

Segmenting a functional unit such that it can accept new operands every cycle while the total execution of the instruction may take many cycles. The pipeline construction works like a conveyor belt accepting units until the pipeline is filled and than producing results every cycle.

Processor array

System in which an array (mostly a 2-D grid) of simple processors execute its program instructions in lock-step under the control of a Control Processor.

PVM

Another message passing library that has been widely used. It was originally developed to run on collections of workstations and it can dynamically spawn or delete processes running a task. PVM now largely has been replaced by MPI.

Register file

The set of registers in a CPU that are independent targets for the code to be executed possibly complemented with registers that hold constants like 0/1, registers for renaming intermediary results, and in some cases a separate register stack to hold function arguments and routine return addresses.

RISC

Reduced Instruction Set Computer. A CPU with its instruction set that is sijmpler in comparison with the earlier Complex Instruction Set Computers (CISCs) The instruction set was reduced to simple instructions that ideally should execute in one cycle.

Shared Memory (SM)

Memory configuration of a computer in which all processors have direct access to all the memory in the system. Because of technological limitations on shared bandwidth generally not more than about 16 processors share a common memory.

SMP

Symmetric Multi-Processing. This term is often used for compute nodes with shared memory that are part of a larger system and where this collection of nodes forms the total system. The nodes may be organised as a ccNUMA system or as a distributed memory system of which the nodes can be programmed using OpenMP while inter-node communication should be done by message passing.

TLB

Translation Look-aside Buffer. A specialised cache that holds a table of physical addresses as generated from the virtual addresses used in the program code.

Torus

Structure that results when the end points of a grid are wrapped around to connect to the starting points of that grid. This configuration is often used in the interconnection networks of parallel machines either with a 2-D grid or with 3-D grid.

Vector unit (pipe)

A pipelined functional unit that is fed with operands from a vector register and will produce a result every cycle (after filling the pipeline) for the complete contents of the vector register.

VLIW processing

Very Large Instruction Word processing. The use of large instruction words to keep many functional units busy in parallel. The scheduling of instructions is done statically by the compiler and, as such, requires high quality code generation by that compiler. VLIW processing has been revived in the IA-64 chip architecture, there called EPIC (see above).

Acknowledgements

It is not possible to thank all people that have been contributing to this overview. Many vendors and people interested in this project have been so kind to provide me with the vital information or to correct us when necessary. Therefore, we will have to thank them here collectively but not less heartily for their support.

References

[Amza95] *C. Amza, A.L. Cox, S. Dwarkadas, P. Keleher, H. Lu, R. Rajamony, W. Yu, W. Zwaenepoel* , TreadMarks: Shared Memory Computing on Networks of Workstations, to appear in IEEE Computer (also: , http://www.cs.rice.edu/~willy/TreadMarks/papers.htm.

[ASCI] , The ASCI program:, http://http://www.llnl.gov/asci/.

[CaSt98] *K. Cassirer, B. Steckel* , Block-Structured Multigrid on the Cenju, 2nd Cenju Workshop, October 1998, Sankt Augustin, Germany. .

[Cull98] *D.E. Culler, J.P. Singh, A. Gupta* , Parallel Computer Architecture: A Hardware/Software Approach, Morgan Kaufmann Publishers Inc., August 1998. .

[Dong02] *J.J. Dongarra* , Performance of various computers using standard linear equations software, Computer Science Technical Report CS-89-85, Univ. of Tennessee, July 17th, 2002. .

[EurB99] , Directory with EuroBen results: , http://www.euroben.nl/results.

[Flan91] *P. Flanders* , Matrix Multiplication on 'C' series DAPs , AMT Document TR40, Jan. 1991. .

[Flynn72] *M.J. Flynn* , Some computer organisations and their effectiveness, IEEE Trans. on Computers, Vol. C-21, 9, (1972) 948--960. .

[Geist94] *A. Geist, A. Beguelin, J. Dongarra, R. Manchek, W. Jaing, and V. Sunderam* , PVM: A Users' Guide and Tutorial for Networked Parallel Computing, MIT Press, Boston, 1994. .

[Gigan01] , http://www.giganet.com">http://www.giganet.com.

[Hill97] *J.M.D. Hill, W. McColl, D.C. Stefanescu, M.W. Goudreau, K. Lang, S.B. Rao, T. Suel, T. Tsantilas, R. Bisseling*, BSPlib: The BSP Programming Library, Technical report PRG-TR-29-9, Oxford University Computing Laboratory, May 1997. (Compressed Postscript with ANSI C examples, 142K; Compressed Postscript with Fortran 77 examples, 141K) .

[Hisr2201] , http://www.hitachi.co.jp/Prod/comp/hpc/eng/sr1.html.

[Hock88] *R. W. Hockney, C. R. Jesshope* , Parallel Computers II, Bristol: Adam Hilger, 1987. .

[Hori91] *T. Horie, H. Ishihata, T. Shimizu, S. Kato, S. Inano, M. Ikesaka* , AP1000 architecture and performance of LU decomposition, Proc. Internat. Symp. on Supercomputing, Fukuoka, Nov. 1991, 46--55. .

[HPF93] *High Performance Fortran Forum* , High Performance Fortran Language Specification, Scientific Programming, 2, 13, (1993) 1--170. .

[JaLa90] *D.V. James, A.T. Laundrie, S. Gjessing, G.S. Sohi*, Scalable Coherent Interface, IEEE Computer, 23, 6, (1990),74--77. See also: Scalable Coherent Interface: , http://http://sunrise.scu.edu/.

[MPI1] *M. Snir, S. Otto, S. Huss-Lederman, D. Walker, J. Dongarra*, MPI: The Complete Reference Vol. 1, The MPI Core, MIT Press, Boston, 1998. .

[MPI2] *W. Gropp, S. Huss-Ledermann, A. Lumsdaine, E. Lusk, B. Nitzberg, W. Saphir, M. Snir*, MPI: The Complete Reference, Vol. 2, The MPI Extensions, MIT Press,

Boston, 1998. .

[Myr00] , http://www.myrinet.com".

[Nagel98] *W.E. Nagel* , Applications on the Cenju: First Experience with Effective Performance, 2nd Cenju Workshop, October 1998, Sankt Augustin, Germany. .

[npb2] *Web page for the NAS Parallel benchmarks NPB2:* , http://science.nas.nasa.gov/Software/NPB/.

[opmp97] *OpenMP Forum* , Fortran Language Specification, version 1.0, Web page: , http://www.openmp.org/.

[SPECT00] *D.H.M. Spector*, Building Unix Clusters, O'Reilly, Sebastopol, CA, USA, July 2000 .

[Steen90] *A.J. van der Steen*, Exploring VLIW: Benchmark tests on a Multiflow TRACE 14/300, Academic Computing Centre Utrecht, Technical Report TR-31, April 1990. .

[Steen91] *A.J. van der Steen* , The benchmark of the EuroBen Group, Parallel Computing 17 (1991) 1211--1221. .

[Steen93] *A.J. van der Steen* , Benchmark results for the Hitachi S-3800, Supercomputer, 10, 4/5, (1993) 32--45. .

[Steen95] *A.J. van der Steen, ed.* , Aspects of computational science, NCF, The Hague, 1995. .

[Steen98] *A.J. van der Steen* , Benchmarking the Silicon Graphics Origin2000 System, Technical Report WFI-98-2, Dept. of Computational Physics, Utrecht University, The Netherlands, May 1998. The report can be downloaded from: , http://www.euroben.nl/reports/.

[Steen00] *A.J. van der Steen*, An evaluation of some Beowulf clusters, Technical Report WFI-00-07, Utrecht University, Dept. of Computational Physics, December 2000. (Also available through , http://www.euroben.nl.

[Ster99] >*T.L. Sterling, J. Salmon, D.J. Becker, D.F. Savaresse*, How to Build a Beowulf, The MIT Press, Boston, 1999 .

[Top500] *H.W. Meuer, E. Strohmaier, J.J. Dongarra, H.D. Simon*, Top500 Supercomputer Sites, 18th Edition, June 20, 2002, The report can be downloaded from: , http://www.netlib.org/benchmark/top500.html.

[TFCC] *Mark Baker (ed.)*, Cluster Computing White Paper, December 2000, to be downloaded from: , http://www.dcs.port.ac.uk/~mab/tfcc/WhitePaper.

List of Figures