misc

CM-5 Terminology

Getting Our Act Together

There has been a lot of confusion lately because we haven't been using a consistent set of names for the various parts of the CM-5. This confusion has caused a great deal of trouble recently in presentations to customers. The purpose of this memo is to dictate a standard set of terms to be used from now on

(Dictate?? At TMC? Yes, *dictate.* I have tried for several months by less formal means to get people to converge. But the problem is so pervasive, and the need so pressing, that there is no other way to solve the problem. This memo has been run past a large set of people already in all parts of the company. No one is completely happy with every decision in the list, but at this point it is much more important to agree than to argue over the choice of terms.)

As an example of the problem, during a recent day-long presentation the terms *PE*, *processing element*, *floating-point unit*, *vector execution unit*, and *Dash* were used by various speakers to refer to the same part of the architecture. Moreover, the terms *PE* and *processing element* were also used to refer to another concept, which was also called *node* and other terms. Over the course of several hours we lost perhaps 15 to 30 minutes, and a certain amount of good will, to terminological inconsistency (which is a highfalutin term meaning that we looked confused).

The problem isn't particularly anyone's fault. Also, if you find some of your favorite words on the "Forbidden List," please don't think you're "wrong" to have used them. The problem is systemic. The use of the words has evolved over time, and so has the way in which we wish to present the machine to the outside world. In the absence of careful coordination of word choice, each person and group has developed words and usages that serve their needs. The problem is that the various choices, and in some cases the purposes behind the choices, are in conflict.

The Official Choices laid down here are the result of compromise. I'm asking you to do a very difficult thing: to change a few of your customary habits when talking about things you may have called by different names for a year or more. This will not be easy. I'm asking you to do it for the good of Thinking Machines and our customers.

—Guy Steele
October 28, 1991

Version 1.2

This document has three parts: a short list of Forbidden Words; a longer Glossary that lists both Forbidden Words and Official Choices, as well as some in-between cases and problematical terms, with some discussion; and a list of CM-5 product codes.

Forbidden Words

"Forbidden" is a matter of context, of course. The simple rule: please don't use these terms when discussing CM-5 with persons outside TMC. If you absolutely have to use one of these terms internally, go ahead; but try to cultivate good habits. It really isn't that much harder to say "vector unit" than to say "DP." Many of these terms are not strictly forbidden, but require care. For extended discussion, see the Glossary.

Instead of this term:	Please use:
CM-X	**CM-5**
Dash	**vector unit**
vector execution unit	**vector unit**
datapath or *DP*	**vector unit**
Processing Element or *PE*	**node** or **processing node**
SPARC (the one in a node)	**[node] microprocessor**
PE memory bus	**node bus**
broadcast network	**Control Network**
combine network	**Control Network**
scan network	**Control Network**
router or *data router*	**Data Network**
DR	**Data Network** (but see Glossary)
DN	**Diagnostic Network** or **DIAG**
fat-tree	(none)
Hypertree	(none)
user partition	**partition**
I/O partition	(none)
system partition	(none)
Front End	**Control Processor** or **Partition Manager**
Scalar Processor or *Scalar*	**Control Processor** or **Partition Manager**
SP, FE	**CP** or **PM**
I/O Scalar Processor	**I/O Control Processor**
IOSP	**IOCP**
ScaleArray	**Scalable Disk Array** or **SDA**
word	**32-bit word** or **32 bits**
doubleword	**64-bit word** or **64 bits**
halfword	**16 bits**
single precision	**32-bit**
double precision	**64-bit**
SIMD programs	**data parallel programs**
MIMD programs	**message-passing programs**

Glossary

Just before the CM-1 and CM-2 announcements, Jim Bailey wrote up glossaries of preferred and forbidden words. Those of you who were around in 1987 may remember such forbidden words as "lock-step," "Payne House," and "Slimebolics." These glossaries were a tremendous help in getting our story straight. This glossary tries to do that again for CM-5. (It is particularly amusing to note that four years ago "supercomputer" was one of the forbidden words! Now that our role in the marketplace has changed and we have a new story to present, "supercomputer" is one of the **required** words!)

In the headings, Good Words are in **boldface**; Bad Words are in *italics*.

address

See **Network Address**.

Batch Partition

A **Partition** that has been designated by the system administrator solely for batch processing of queued user tasks. Note that it is also possible to perform batch processing in a partition timeshared with non-batch (interactive) tasks. See **Dedicated Partition, Timeshared Partition**.

broadcast network

Use **Control Network**. Broadcasting is only one of the functions of the Control Network. One of our Simplified, Easy-to-Grasp Messages is that the CM-5 has exactly two internal user-visible networks, with the pleasantly symmetrical names "Control Network" and "Data Network". Using other terms such as "broadcast network" muddles this message (and confuses customers).

cabinets

See **Device Cabinet, Volume Expansion Device Cabinet, Network Cabinet,** *Height-3,4 Cabinet*, and *Height-5 Cabinet.*

chunk

A group of 32 **network addresses**; also, the chassis or physical space for holding equipment that occupies 32 network addresses. One **device cabinet** holds eight chunks. See **cluster**.

cluster

A group of 32 **Processing Nodes**, that is, a **chunk** fully populated with nodes.

CM-X

As you are no doubt already aware, the official name for the product is **CM-5**. We must of course get out of the habit (carefully cultivated by some of for the last N months!) of calling it *CM-X*.

CM-5

The world's most powerful computer. It is massively parallel. It is known for scalability, universality, and availability. It provides processing power that scales over a range of more than two orders of magnitude, as well as tremendous amounts of I/O capacity. It handles not only data parallel programs but also message-passing and other parallel programming structures. And it takes a licking but keeps on ticking.

CN

This is the approved abbreviation for Control Network for hardware labels and software function names. Avoid using it in block diagrams, running text, and oral presentations.

combine network

Use **Control Network**. Combining is only one of the functions of the Control Network. See *broadcast network*.

Control Network

This is the official name for the part of the CM-5 that does broadcasting, reduction, parallel prefix operations, interrupt broadcasting, and so on. The terms *broadcast network*, *combine network*, and *scan network* should be avoided because it sounds as though there are more networks in the machine. The message we wish to convey to customers is that there are two user-visible networks: the Control Network and the Data Network. This is easy to remember and has a pleasant symmetry.

Control Processor, CP

The official term for processors that manage partitions or I/O. A control processor runs a full copy of UNIX. Do not use *Scalar Processor* for this.

The abbreviation **CP** is suitable for use in block diagrams and oral presentations.

The term "Control Processor" is a hardware term and a collective term. Particular Control Processors will have different roles in the system, some perhaps playing more than one role at a time. The Control Processor associated with a partition may be referred to as a **Partition Manager**. A Control Processor that services I/O requests and directs I/O devices is an **I/O Control Processor** (in contrast to an **I/O Processor**, which is out there on the board with the **NI** chips that connect to an I/O device). A Control Processor that serves as the system console interface is the **System Console Processor**.

In principle, all Control Processors are more or less identical. In practice, when a system is configured, each Control Processor will be assigned a specific role. When a customer orders hardware, we might specifically configure it with a system console processor, two IOCP's, and five partition managers, not simpl eight Control Processors. This comes about because partition managers need to be connected into the networks at certain spread-out points, not all bunched up together, and because typically one IOCP is needed for each I/O device. In a pinch, however, a partition manager might be able to pinch-hit for a broken IOCP, and vice versa, but the software details remain to be worked out.

Control Processor Interface, CPI

An interface for connecting Control Processors to the rest of the CM-5 system. One CPI can support up to four control processors (**CP-1, CP-2, IOCP**, or **IOCP-VME**), with the restriction that no more than two of them can be partition managers.

CP

See **Control Processor.**

CP-1

A **Control Processor** model CP-1 has the necessary interface to the **Diagnostic Network** to serve as a **System Console Processor.** Normally it is internally mounted within a CM-5 cabinet. It also has an external monitor and keyboard for operator control. Every CM-5 system must have at least one CP-1. Under certain circumstances, a CP-1 acting as the System Console Processor can serve as a **Partition Manager** at the same time. (Whether this is advisable depends on the overall size of the system as well as on the details of its configuration.)

CP-2

A **Control Processor** model CP-2 normally is internally mounted within a CM-5 cabinet. It is used as a **Partition Manager.** The recommended maximum number of users per CP-2 is 10. As an example, a CM-5 system large enough to support four partitions would probably have one CP-1 and three CP-2's and could support a recommended maximum of 30 to 40 simultaneous users.

CPI

See **Control Processor Interface.**

Dash

Dash is an implementation name, the name of a chip. One Dash chip contains two vector units. One node contains four vector units but only two Dash chips. We should avoid using the word "Dash" outside the company except when discussing the chip per se.

It's okay to use the name internally, for example when speaking of a compiler that is "targeted to Dash" because in this instance one really means a specific implementation of a specific instruction set. But to outsiders one should speak of a compiler that "generates vector code" or "targets the vector units".

Data Network

This is the official name for the part of the CM-5 that does point-to-point message passing from one Network Interface to another. The terms *router* and *data router* should be avoided. The message we wish to convey to customers is that there are two user-visible networks: the Control Network and the Data Network. This is easy to remember and has a pleasant symmetry.

We also want to emphasize that data is routed by a rather different mechanism in the CM-5 as compared to the CM-2 Router. It's okay to make an explicit analogy between the CM-2 Router and the CM-5 Data Network, but the CM-5 Data Network has ever so many more capabilities.

Never abbreviate Data Network as DN, which has always stood for Diagnostic Network and will continue to do so. In the future the alternative abbreviation DIAG may be used for the diagnostic network for extra clarity. The approved abbreviations for the Data Network for such purposes as board and chip labels and subroutine names are **DR** (which has been used until now, and may continue to be) and **DATA**. In technical documentation, diagrams, and oral presentations the full phrase "Data Network" should be used. We may continue to use the term "router" internally, especially in hardware design and manufacturing, and also in such well-established phrases as "height-5 router," but outside the company the preferred term is "Data Network". Note that the CM-2 still has a router, not a data network.

As a verb, "routing" is perfectly acceptable for all purposes. For example, we might tell a customer that the CM-5 Data Network is responsible for routing messages.

Datapath, DP

After experimenting with "datapath" we got feedback internally that it was too bland and (more importantly) from customers who found it bland *and* confusing. We shall call it a **Vector Unit**.

Dedicated Partition

A **Partition** that has been designated by the system administrator for processing of a single user task. See **Batch Partition, Timeshared Partition**.

Device Cabinet

A primary CM-5 cabinet that can contain major components such as processing nodes, control processors, and peripheral devices and interfaces. Device cabinets also have power supplied, network logic, and connections to support and integrate these components as a complete system, possibly across multiple cabinets. A synonym used only within TMC is *Height-3,4 Cabinet*. See **Volume Expansion Device Cabinet** and **Network Cabinet**.

One device cabinet has a capacity of eight **chunks**, which is 256 **Network Addresses**.

Diagnostic Network

This is the official name for the part of the CM-5 that can test and isolate various chips and connections within the machine. The Diagnostic Network is sturdy and robust, and has caused very little trouble except for snarfing the abbreviation **DN** so that we cannot use DN for "Data Network."

DN

This is an approved abbreviation for Diagnostic Network for hardware labels and software function names. Avoid using it in block diagrams, running text, and oral presentations. In the future the alternative abbreviation **DIAG** may be used in order to avoid confusion with the Data Network; however, so many chips, cables, and drawings already use "DN" that we will continue to use and recognize it.

double precision

Avoid assuming that "double precision" means 64 bits. To many of our customers, single precision is 64 bits and double precision is 128 bits, IEEE Standard 754 for Floating-Point Arithmetic notwithstanding (they refer to 32-bit floating-point numbers, somewhat condescendingly, as "half-precision"). It saves a lot of confusion to speak of 32-bit and 64-bit floating-point arithmetic. See *word*.

DP

Do not use. **Vector Unit** is the preferred term. See *Datapath*.

DR

This is an approved abbreviation for Data Network for hardware labels and software function names. (Of course, DR really stands for *Data Router*, which itself a forbidden word. Someday we'll look back on this and chuckle.) Avoid using DR in block diagrams, running text, and oral presentations. In the future the alternative abbreviation **DATA** may be used in order to avoid confusion with the Diagnostic Network; however, so many chips, cables, and drawings already use "DR" that we will continue to use and recognize it.

Expansion Cabinet

See **Volume Expansion Device Cabinet**.

External I/O Processor

An **I/O Control Processor** that specifically has functional responsibility for driving one or more external I/O devices that typically do not have file systems—for example, tape drives or a graphics display. Contrast this with a **File System Control Processor**. This is a distinction that is of particular importance to systems programmers and users writing I/O code.

An external I/O Processor is implemented using the technology of a control processor; functionally, it performs tasks associated with **I/OProcessors** as well as certain non-file-system tasks usually handled by File System Control Processors.

Typically, though not always, this task is given to the kind of I/O control processor that is physically outside a CM-5 **Device Cabinet**, because of the need to cable the external I/O devices to it. See product code CM-IOCP-VME-1.

fat-tree

If we were planning to say much about the topology of the Data Network, we would use the term Hypertree rather than fat-tree. But we're not. As a rule, we prefer to state performance characteristics of the Data Network and Control Network, such as latency, throughput, and the ratio of best case to worst case, rather than get bogged down in "religious wars" about which topology is best.

File System Control Processor

An **I/O Control Processor** that specifically has functional responsibility for maintaining a file system on an I/O device or a set of I/O devices. Typically, though not always, this task is given to the kind of I/O control processor that is physically contained within a CM-5 **Device Cabinet**. See product code CM-IOCP-1.

Height-3,4 Cabinet

This terminology is considered internal to TMC. It is a synonym for a **Device Cabinet**, emphasizing the fact that a Device Cabinet contains the logic that implements levels 3 and 4 of the network structure.

Height-5 Cabinet

This terminology is considered internal to TMC. It means a **Network Cabinet** containing the logic to implement level 5 of the network structure. A large CM-5 system might also need *Height-6 Cabinets* or even *Height-7 Cabinets.*

Hypertree

See *fat-tree.*

I/O Control Processor, IOCP

This term refers to a **Control Processor** in its role as the manager of an I/O device or a set of I/O devices. Contrast this term with **I/O Processor**. Note that this is a functional term, rather than identifying a particular kind of hardware. Any control processor can function as an I/O Control Processor; however, in practice certain control processors will be designated as I/O control processors for purposes of system configuration. See also **External I/O Processor, File System Control Processor, Partition Manager**, and **System Console Processor**.

I/O Partition

No longer used. See **Partition.**

I/O Processor, IOP

An I/O processor is on the same board as the Network Interface chips that connect an I/O device to the internal networks. It typically performs processing that is specific to one I/O device. In contrast, an **I/O Control Processor** is a **Control Processor** that may be physically or logically quite distant from the Network Interfaces for an I/O device but is responsible for processing I/O requests and managing filesystems.

I/O Support Processor, IOSP

Use **I/O Control Processor** or **IOCP**.

MIMD

Multiple Instruction, Multiple Data.

"SIMD", "MIMD", and "vector machines" are architectural terms, not programming terms. Programs and programming styles may be characterized as data parallel (which is not the same as *SIMD*), message-passing, *SPMD* (although that's a rather ugly term), and also as coarse-grained or fine-grained. Architecturally, the CM-2 is SIMD; the CM-5 is MIMD. Both support data parallel programming; the CM-5 does so more efficiently in some cases. The CM-5 supports data parallel programming more efficiently than many other MIMD architectures, thanks in large part to the careful design of its internal networks. The CM-5 also supports message-passing programs. Sometimes we say that CM-5 has a universal parallel architecture.

We don't speak of "SIMD programs" and "MIMD programs". Lots of people outside TMC will use these terms, and we should treat their usage with some respect; but be aware that by this they usually mean "a program written in the style I am used to using on a SIMD (or MIMD) architecture". There are lots of ways to write a program for a MIMD machine—the data parallel style is one, message-passing is another, and shared memory with semaphores (spin-locks) is yet another.

Network Address

Every **Network Interface** in a machine has a distinct physical address. Each message sent through the Data Network is directed to a particular Network Interface by specifying the address of that interface. Many Control Network operations (such as parallel prefix) assume that network interfaces are linearly ordered according to their network addresses.

A specific byte of memory anywhere in the machine can be addressed by a 64-bit pointer; the high-order 32 bits indicate the network address, and the low-order 32 bits are the memory address of the byte within whatever is connected to that network interface. (Such pointers must be processed by suitable software, however; remember that there is currently no direct hardware support for memory access through such a pointer.)

Network Cabinet

A cabinet that contains the network logic necessary to integrate multiple **Device Cabinets** into a fully operational CM-5 system. The number and type of network cabinets in a system is dictated by the number of Device Cabinets. Within TMC, different kinds of Network Cabinets are distinguished as *Height-5 Cabinets*, *Height-6 Cabinets*, and *Height-7 Cabinets*.

Network Interface

The standard interface by which nodes and I/O devices are connected to the **Control Network** and the **Data Network**. Each network interface provides one Control Network connection and one Data Network connection and occupies one **Network Address**.

NI

Official abbreviation for **Network Interface** for all purposes.

Node

This is the approved term for what has sometimes in the past been called a *Processing Element* or *PE*. It contains a microprocessor, a node bus, memory, and a network interface, and may optionally contain a vector accelerator consisting of four vector units.

The simple word **Node** is recommended for use in oral presentations; *please* try not to say "*PE*". The term Processing Node may be used for emphasis. In block diagrams, a small box representing a processing node may be labelled with the single letter **P**. For software labels and so on, the abbreviation is "node" (although the abbreviation "PE" will be grandfathered for the sake of existing software—but avoid using it in new software). In other contexts the abbreviation **PN** may be used where necessary.

Machine sizes should be measured in nodes; we speak of a "512-node machine", for example. (The Minnesota machine was a 544-node machine. Note how helpful it is not to be committed, at this level of description, to whether it has vector units.) Note, too, that the number of nodes can be distinguished from the fact that its network spans 1024 network addresses (or, equivalently, that it has a height-5 router, though that is terminology more likely to be used internally than outside TMC).

Node Bus

The bus, internal to the node, which connects the microprocessor, the network interface, and the memory controller or vector units. This is currently implemented as a 64-bit bus supporting the SPARC MBUS protocol. (This is not the same as the 64-bit bus that connects a vector unit to its associated memory bank.)

Node Microprocessor, Node Processor

The processing unit in a **Node** that fetches and executes instructions, issues instructions to the vector units, and issues commands to the Network Interface for that node. In the initial implementation of the CM-5, the node microprocessor is implemented as a SPARC chip set. We are holding open the possibility that, in the future, the node microprocessor might be implemented in other ways. See *SPARC*.

There is nothing wrong, particularly in oral presentations, in referring to "the SPARC", *provided* that one is absolutely certain no confusion will result from the fact that the CP also has a SPARC. When in doubt, clarify.

In block diagrams, the abbreviation **μP** may be used to label the node microprocessor. (On a Macintosh, the character "μ" is Option-m.)

The term "Node Processor" may be used instead of "Node Microprocessor".

Partition

The terms "I/O partition" and "system partition" were used at one time, but have been declared defunct; the regions of the system to which they referred are not really partitions. We will use the term "partition" to refer to a group of nodes, under the control of a Control Processor, used for executing user tasks. A partition that has been dedicated to a particular role by the system administrator may be described as a timesharing partition, batch partition, or dedicated partition.

Partition Manager, PM

This term refers to a **Control Processor** in its role as the manager of a partition (a group of nodes that can execute user code). Note that this is a functional term, rather than identifying a particular kind of hardware. Any control processor can function as a partition manager. See also **I/O Control Processor** and **System Console Processor**.

PE

Do not use! **Node** is the preferred term. See *Processing Element* for reasons.

PE memory bus

See **Node Bus**.

PM

See **Partition Manager**.

PN

See **Node, Processing Node**.

Processing Element

Do not use! **Node** is the preferred term.

While"*processing element*" and especially its acronym "*PE*" roll lightly off the tongue, it has proved confusing to many outsiders and even to some within TMC, because it is not clear whether a PE is a single vector unit or the grouping of a microprocessor plus four vector units. Indeed, we have had considerable confusion in certain customer presentations because of differing usage between, say, the hardware design group and the compiler group. We intend to eliminate the term "Processing Element" and its abbreviation "PE", *except* insofar as it is already embedded into software (such as names of functions) and its associated documentation, especially that shared with the CM-2. The term is deprecated for further use.

We refer to the unit containing a microprocessor, a bus, memory, a network interface, and optionally a vector accelerator (consisting of four vector units) as a **Processing Node** or simply a **Node**. The term "node" also fits in well with Intel or NCUBE terminology: a node is one autonomous computational engine, capable of fetching and executing its own instructions. This also leaves us some wiggling room for the future in case the architecture of a node changes; we aren't committed to a "node" being an "element".

Processing Node

See **Node**.

Scalar

See *Scalar Processor.*

Scalar Processor

Use **Control Processor.**

The term "Scalar Processor" has gotten us into trouble in several ways.

First, it leads some customers to think that it necessarily executes all parts of the code that are "scalar." This is not necessarily so, because the microprocessor in each node will actually bear some of the burden of executing such code (such as loop control and address calculations); thus some computations will be carried out redundantly, but that may be cheaper than doing them in one place and then broadcasting the results.

Second, those same customers frequently jump to the conclusion that the "Scalar Processor," not being terribly powerful at scalar performance (as compared to a Cray), will be a bottleneck. We need to emphasize that while part of a user program will indeed run on this central processor, its primary concern is with operating systems tasks, I/O, user interface, and so on. Our initial language implementations will indeed rely on this processor for scalar computations as a means of making a smotth transition from the CM-2, but as time goes on more of these computations will be moved down into the nodes.

Third, the term "scalar processor" frequently degenerates in oral presentations to simply "the scalar," which is confusing because "scalar" already has an important technical meaning, namely a scalar data item, as opposed to a vector.

ScaleArray

There is some controversy over whether this term is appropriate. One problem is that the first two syllables are pronounced "scalar", which is The Wrong Message. On the other hand, some people don't have a problem with this, and it is catchy.

An alternative name that has been suggested for the successor to DataVault is the "Scalable Disk Array" or "SDA". This will not necessarily be the final product name, so stay tuned; or this might turn into the generic name for a series of products.

In any case, the point is moot for now because it is not an announced product, so please don't use the term outside the company until further notice.

scan network

Use **Control Network**. Scanning (parallel prefix operations) is only one of the functions of the Control Network. See *broadcast network*.

scatter/gather

As I understand it, "gather" has meant two slightly different things in the Cray world: (1) indirect addressing performed as a vector operation, using a vector of indexes that are each added to a given base address; (2) compressing a vector under control of a vector mask so that items where the mask is true are placed contiguously at the front of the result vector. "Scatter" has had two meanings also, with the obvious correspondence. Apparently early Cray-1 systems had hardware only for (2), which was then used by a library routine to accomplish (1). Some customers with Cray experience have expressed confusion at the claim that our vector units have "scatter/gather hardware", and some have not. All agree, however, that our vector units have "indirect addressing hardware".

So I recommend that we speak always of "indirect addressing hardware" when describing that particular facility, but we can also make the more general claim, in more general contexts, that scatter/gather operations are supported.

SIMD

Single Instruction, Multiple Data. This term should be used carefully; it describes a style of machine architecture, not a style of programming. The CM-2 is a SIMD machine, but we don't speak of SIMD programs. See *MIMD*.

single precision

Avoid assuming that "single precision" means 32 bits. To many of our customers, single precision is 64 bits and double precision is 128 bits, IEEE Standard 754 for Floating-Point Arithmetic notwithstanding. It saves a lot of confusion to speak of 32-bit and 64-bit floating-point arithmetic. See *word*.

SPARC

This is not a forbidden term, but it should be used carefully. While the use of SPARC has certain engineering and marketing advantages, it is not crucial to the overall architecture, and we want the freedom to ride the technology curve as new processors emerge. So in official architectural specifications we will avoid referring to "the SPARC". This also eliminates another source of confusion that stems from the fact that SPARC chips are used in the Control Processors and I/O processors as well as in the nodes. Instead, we will refer to the **Node Microprocessor** or simply "the microprocessor" (once the node has been established as the context for discussion).

On the other hand, the first generation of CM-5 will indeed use SPARC, and we don't intend to hide that fact. There is nothing horribly wrong, particularly in oral presentations, in referring to "the SPARC", *provided* that one is absolutely certain no confusion will result from the fact that the CP also has a SPARC. When in doubt, clarify.

On the third hand, we would like to emphasize that we are using RISC technology in our processing nodes, even though we don't necessarily want to be tied to the SPARC architecture. The best strategy, then, if you must be specific about the node processor, is to refer to "the RISC processor" rather than to "the SPARC processor".

SPMD

Single Program, Multiple Data. An oddball term coined long after Flynn's original division of hardware architectures into SISD, SIMD, MIMD, and the rarely-heard-of MISD. It is important to understand what it means, because it has gained some currency in the outside world; but it is also important to understand that it describes neither architecture alone nor programming style alone, but an incomplete mixture of the two. It conveys the idea that the hardware is capable of operating on multiple data items but under the control of a single program. At that level of description it might seem that SIMD is a special case of SPMD, but in practice the implication is almost always that the hardware has a MIMD architecture and that the processors could in principle each contain a different program, but as a matter of software discipline each processor contains (a copy of) the same single program. The term leaves open the question of the method of communication (message passing, shared memory, or what have you), although some persons who use the term may make an implicit assumption when using the term SPMD. We are better off concentrating on more specific terms and emphasizing CM-5 support for data parallelism and message passing; but if someone asks whether CM-5 handles SPMD, "yes" is not an incorrect answer.

System Console Processor

This term refers to a **Control Processor** in its role as the system console, that is, the processor responsible for certain system administrative tasks such as reconfiguring partitions and driving the diagnostic network. Note that this is a functional term, rather than identifying a particular kind of hardware. Any control processor can function as a system console processor. See also **I/O Control Processor** and **Partition Manager**.

System Partition

No longer used. See **Partition**.

Timeshared Partition, Timesharing Partition

A **Partition** that has been designated by the system administrator for timeshared processing of user tasks. Note that batch processing may be active under timesharing. See **Batch Partition, Dedicated Partition**.

User Partition

No longer used, because there isn't any other kind. See **Partition**.

Vector Unit

The vector processing power associated with one bank of memory in a node. Formerly called a *datapath*. One Dash chip implements two vector units.

This may be abbreviated to "VU" where absolutely necessary (shouldn't be necessary in technical documentation), or to "V" in block diagrams.

Avoid *Dash*. Dash is an implementation name, the name of a chip. One Dash contains two vector units. One node contains four vector units (if any) but only two Dash chips. We should avoid using the word "Dash" outside the company except when discussing the chip per se. A vector unit also functions as a memory controller. A node that lacks vector units still has a memory controller.

We refrain from speaking of "floating-point units" or "FPU's" when referring to vector units for two reasons. First, there are other FPU's elsewhere in the machine (associated with the node microprocessor and in the Control Processors). Second, the vector units significantly accelerate integer arithmetic and logical operations, not just floating-point, and this is of particular importance to a number of customers.

One may speak of "a CM-5 with vector units" or a "CM-5 without vector units". The term "vector accelerator" refers to the aggregate of all vector units in a node or in a machine.

Volume Expansion Device Cabinet

A Volume Expansion Device Cabinet contains the same chassis as a **Device Cabinet** but no power supplies or network connections. It can provide inexpensive volume expansion when a Device Cabinet has insufficient physical space for all that is to be connected to its set of **Network Addresses**.

This is the official name for the cabinet, but it's a mouthful. For informal purposes, **Expansion Cabinet** will do.

word

This word is a troublemaker and should be qualified or avoided. See *single precision* and *double precision*, which also make trouble.

A SPARC word is 32 bits, and many of our technical staff refer to 32-bit quantities as words and 64-bit quantities as doublewords. Where some of our customers come from (such as the Cray world), a word is 64 bits and anything less than that is almost beneath notice. We also have some dealings with Apple, and recall that on a Motorola 680x0 a word is 16 bits; 32 bits is a longword.

One solution is to measure everything in bits or bytes; that, at least, is equally inconvenient for everyone! Another is to carefully qualify references as "32-bit words" or "64-bit words" until the appropriate context is established.

I believe that as a matter of philosophy we should establish the CM-5 in everyone's minds and hearts as a 64-bit architecture. It handles 64-bit floating-point; it handles 64-bit integers; it handles 64-bit addresses (32 bits of network address plus 32 bits of memory address); and it handles files whose size exceeds 2^{32} bytes. We simply have to use 64-bit data items wherever appropriate. Having a 64-bit architecture puts us in the supercomputer league.

That's not to say that we should assume that the word "word" itself unambiguously means 64 bits. Beware this pitfall and strive always for clarity. It may seem awkward or time-consuming always to specify the number of bits, but it sure saves time over having to unravel confusion after the fact.

CM-5 Product Codes

CM-CP-1 Control Processor w/ 64 MB memory, 840 MB disk, including interfaces to CPI (CM-SNI) & diagnostic network (CM-SDN). Can be used as Console and Diagnostic Control Processor. Requires available port on CPI. Each CM-5 system must have at least one CM-CP-1 for diagnostics.

CM-CP-2 Control Processor w/ 32 MB memory, 840 MB disk & interface to CPI (CM-SNI). Requires available port on CPI.

CM-IOCP-1 I/O Control Processor w/ 32 MB memory, 840 MB disk & interface to CPI (CM-SNI). Requires available port on CPI.

CM-IOCP-VME-1 I/O Control Processor. SUN 4/330 (w/VME & SCSI interfaces) External to CM-5 w/ 32 MB memory, 669 MB disk & interface to CPI (CM-SVME). Requires available port on CPI.

CM-SVME VME board. Provides Sun 4 to CM-5 (CPI) connectivity to Data Network, Control Network & Diagnostic Network. This board is included in the product CM-IOP-VME-1.

CM-CPI Control Processor Interface. Connects up to 4 Control Processor devices. No more than 2 CP's are allowed per CPI.

CM-SNI S-bus Network Interface board. Provides SparcStation to CM-5 (CPI) connectivity, this board is included in the products CM-CP-1, CM-CP-2, CM-IOCP-1.

CM-SDN S-bus Diagnostic Network Interface board. Provides SparcStation to Diagnostic Network connectivity. Required for Console Processor. This board is included in the product CM-CP-1.

CM-DV20C 20 GB DataVault; Each DataVault requires one CM-IOBA and one CM-IOCP to function with the CM-5 system. (Existing CM-2 product.)

CM-DV30D 30 GB DataVault; Each DataVault requires one CM-IOBA and one CM-IOCP to function with the CM-5 system. (Existing CM-2 product.)

CM-DV40C 40 GB DataVault; Each DataVault requires one CM-IOBA and one CM-IOCP to function with the CM-5 system. (Existing CM-2 product.)

CM-DV60D 60 GB DataVault; Each DataVault requires one CM-IOBA and one CM-IOCP to function with the CM-5 system. (Existing CM-2 product.)

CM-HIPPI 100 MB/sec HIPPI Channel (but restricted to 40 MB/sec peak when connected to IOBA). Requires one CM-IOBA and one CM-HIPPI to function with the CM-5 system. (Existing CM-2 product.)

CM-TUD Storage Technology dual 3480 tape unit; STK 4980. Holds 20 tapes at a time (= 4GB). One CM-TUD requires one CM-IOPG Controller; two CM-TUD's require one CM-IOP2G. (Existing CM-2 product.)

CM-IOPG CM I/O Processor w/single 32 MByte VME board. One CM-IOPG is required to run one CM-TUD, and a CM-IOBA is required to connect this product to the CM-5 system. (Existing CM-2 product.)

CM-IOP2G

CM I/O Processor w/dual 32 MByte VME boards. One CM-IOP2G is required to run two CM-TUD's and a CM-IOBA is required to connect this product to the CM-5 system. (Existing CM-2 product.)

CM-IOBA

I/O Bus Adapter. A CM I/O bus connection for CM-HIPPI, DataVaults and tape units. Peak transfer rate of I/O Bus is approximately 40 MB/sec.

CM5-EX32V-32

Expansion of a CM-5 system by 32 processor nodes, each with 32 MB memory and a vector unit. Additional device cabinets and/or network cabinets may be necessary for use, depending upon base configuration.

CM-DC

Device Cabinet for CM-5 system with a total of 256 network addresses for processor nodes and/or I/O products.

CM-DC-VE

Volume Expansion Device Cabinet. This cabinet provides volume expansion for a CM-DC which has reached its maximum space limit, but not its network address limit. It uses the network addresses from the CM-DC it is supplementing.

CM5-NC1

Upper level routing; Connects two Device Cabinets (512 Network Addresses) and includes one Network Cabinet.

CM5-NC1.5

Upgrades upper level routing from CM5-NC1 to CM5-NC2 (to support up to 1024 network addresses).

CM5-NC2

Upper level Routing; Connects three to four Device Cabinets, (up to 1024 network addresses) and includes one Network Cabinet.

CM5-NC3

Upper level Routing; Connects two CM5-NC2's, (to support up to 2048 network addresses) and includes one Network Cabinet.

CM5-NC3.5

Upgrades upper level Routing from CM5-NC3 to CM5-NC4; (to support up to 4096 network addresses) and includes one Network Cabinet.

CM5-NC4

Upper level Routing; Connects three to four CM-NC2's (to support up to 4096 network addresses) and includes two Network Cabinets.