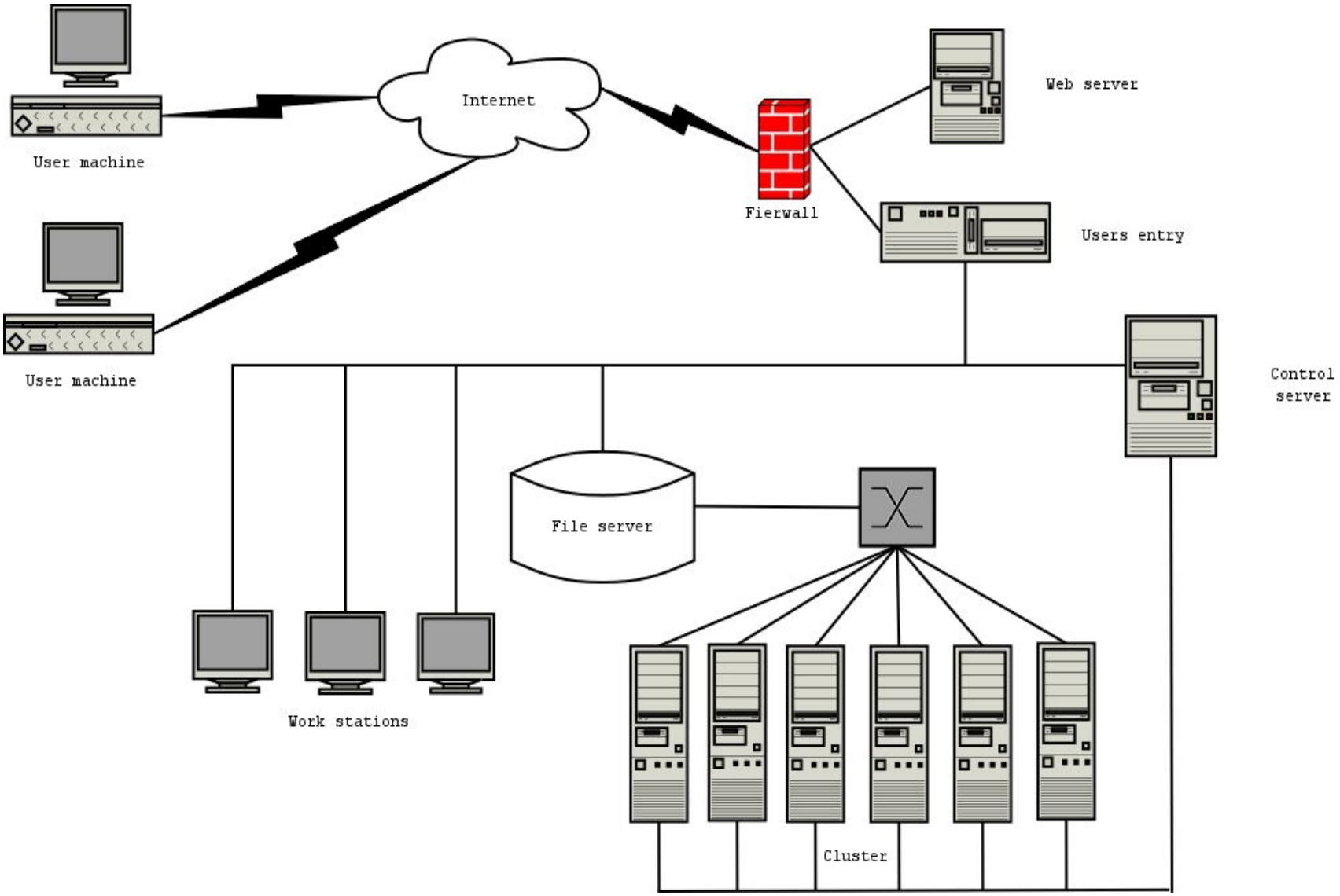
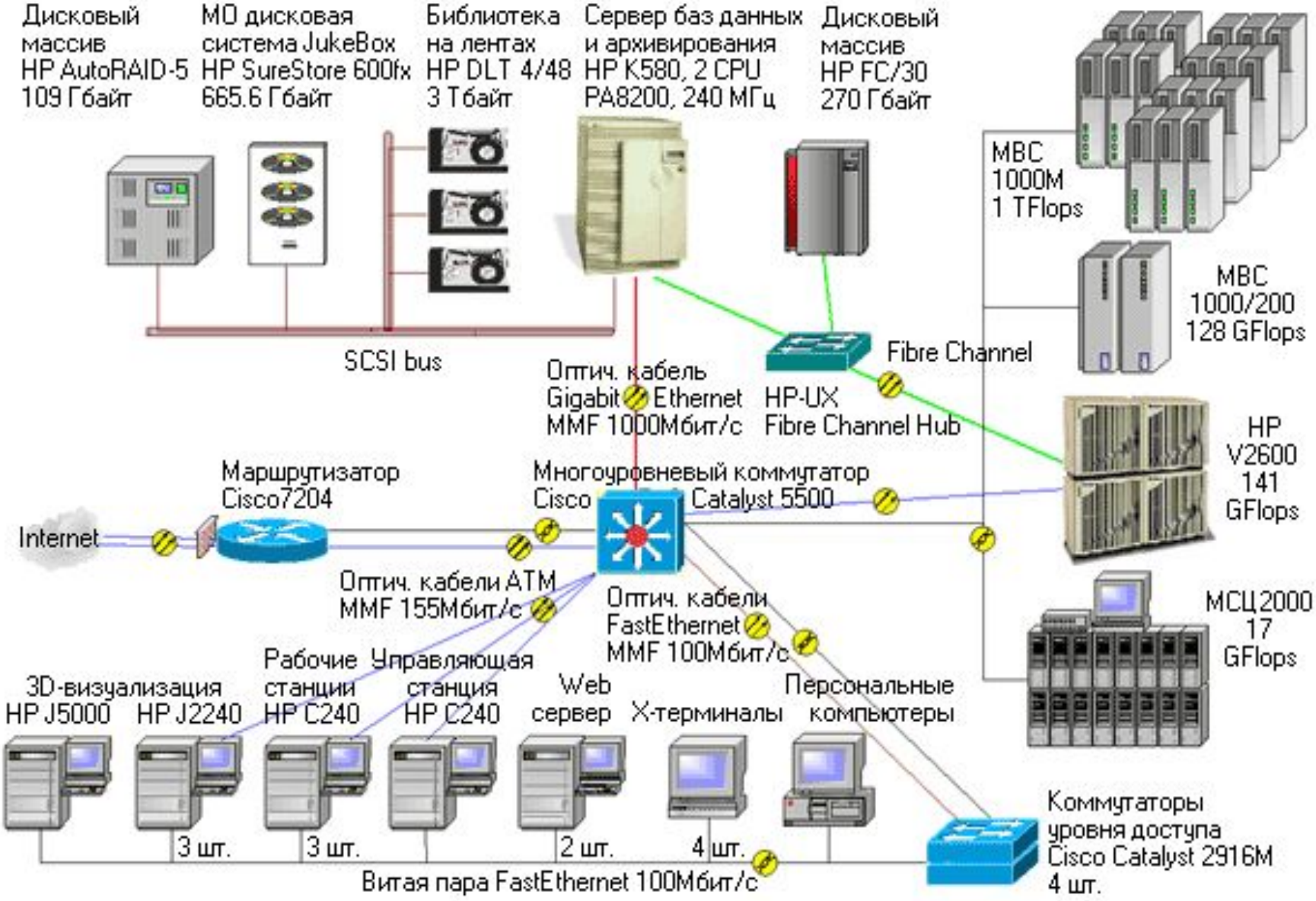
# Пользовательская среда многопроцессорного комплекса.

# Типичная структура центра.



# Структура МСЦ



# Реализации SSH протокола

- Под windows
  - Putty ( http://www.chiark.greenend.org.uk/~sgtatham/putty/download.html )
  - WinScp ( http://winscp.sourceforge.net/eng )
- Под Unix
  - OpenSSH (http://www.openssh.org )

# Система контроля запуска приложений на МВС-1000/М

# Концепции

- *Параллельная программа (задача)* – программа (задача), состоящая из нескольких взаимодействующих *процессов*, каждому из которых необходим для выполнения отдельный процессор.

- Принцип пакетного режима работы.

# Пакетный режим

- Весь производимый задачей в процессе выполнения стандартный вывод (печать на экран), а также стандартный вывод сообщений об ошибках будут перенаправлены в специальные *выходные файлы*.

- Задача пользователя выполняется на МВС-1000/М определенное время (по умолчанию – 5 часов). Пользователь имеет возможность самостоятельно специфицировать время выполнения. По истечении специфицированного времени задача будет снята со счета.

# Пакетный режим

- Каждая запущенная задача получает уникальное имя, состоящее из символьного имени задачи и ее номера, что позволяет запускать одну и ту же задачу одновременно в нескольких экземплярах.

- Каждая запущенная задача контролируется специальным процессом управляющей ЭВМ – менеджером задачи.

# Пакетный режим

- Запуск задач на счёт производится только менеджером очередей. Пользователю предоставляется возможность поставить свою задачу в очередь.

- Предполагается, что работа пользователя будет происходить только на сервере доступа.

# Классы задач.

- ***Отладочные задачи*** – это короткие по времени задачи, которые запускаются исключительно в целях отладки.

- ***Пакетные задачи*** – это средние по времени задачи, которые производят реальные расчеты и выполняются, не прерываясь.

- ***Фоновые задачи*** – задачи с большим временем счета, которые могут прерываться системой. Для фоновой задачи пользователь должен явно указать *квант* – минимальное время счета фоновой задачи, в течение которого задачу прерывать нельзя.

# Приоритеты пользователей

Задача пользователя с высоким приоритетом может посчитаться раньше, чем задача пользователя с низким приоритетом. Приоритет пользователя определяется по указанной шкале и напрямую зависит от суммарного времени счета пользователя за учетный период. Например, если шкала имеет следующий вид:

(120,300,600,1200,0)

то это означает, что наивысшим приоритетом будут обладать задачи пользователей, которые за учетный период считали менее 120 минут, низшим приоритетом будут обладать задачи пользователей, считавших более 1200 минут. Последний ноль означает «конец списка».

# Запуск MPI программ.

- **mpirun –np <число_используемых_процессоров> [параметры_mpirun...] <имя_программы> [параметры_программы...]  [-host <host>]**

- **-h** интерактивная подсказка по параметрам команды **mpirun**.

- **-maxtime <максимальное_время>** Максимальное время счета. От этого времени зависит положение задачи в очереди. После истечения этого времени задача принудительно заканчивается.

- **-np <число_процессоров>** Число процессоров, требуемое программе.

- **-quantum <значение_кванта_времени>** Этот параметр указывает, что задача является фоновой, и задает размер кванта для фоновой задачи.

# Запуск MPI программ

- **-restart** Указание этого ключа приведет к тому, что после своего завершения задача будет вновь поставлена в очередь. Для удаления из очереди такой задачи пользуйтесь стандартной командой **mqdel,** а для ее завершения – командами **mkill** или **mterm**.

- **-stdiodir <имя_директории>** Этот параметр задает имя каталога стандартного ввода/вывода, в который будут записываться протокол запуска задачи, файл стандартного вывода и имена модулей, на которых запускалась задача.

- **-stdin <имя_файла>** Этот параметр задает имя файла, на который будет перенаправлен стандартный ввод задачи.

# Запуск MPI программ

- **-stderr <имя_файла>** Этот параметр задает имя файла, на который будет перенаправлен стандартный вывод сообщений об ошибках задачи.

- **-stdout <имя_файла>** Этот параметр задает имя файла, на который будет перенаправлен стандартный вывод задачи.

- **-transform <имя_командного_файла>** Этот параметр задаёт скрипт отличный от скрипта по умолчанию, который производит преобразование параметров запуска к параметрам запуска, которые понимает mpirun mpich-a.

# Запуск MPI программ

- **-width** Использовать альтернативный способ нумерации процессоров. По умолчанию процессы задачи распределяются по процессорам выделенных модулей в следующем порядке: 1-й процесс - на 1-й процессор 1-го модуля, 2-й процесс - на 1-й процессор 2-го модуля, 3-й процесс - на 1-й процессор 3-го модуля и т.д. После занятия всех 1-х процессоров всех выделенных модулей занимаются 2-е процессоры в том же порядке.

# Запуск MPI программ

- -host Параметр задаёт управляющую машину. Разные управляющие машины задают реально разные многопроцессорные системы на которых работает данная система очередей.

- -s Параметр задающий дополнительную многопроцессорную систему.

# Список команд

- **mpirun** – запуск MPI приложений
- **mrunf <имя_файла-паспорта>** - запуск задачи по паспорту задачи.
- **mfree** – число свободных процессоров.
- **mps [имя_задачи.номер_задачи]** – просмотр статуса задач(и), в том числе находится ли в очереди.

# Список комманд

- **mkill [имя_задачи.номер_задачи]** – убивает запущенную или стоящую в очереди задачу. (только данную итерацию.) “*” – означает все задачи данного пользователя.

- **mterm [имя_задачи.номер_задачи]** – завершение задачи вместе со всеми указанными итерациями.

- **mout [имя_задачи.номер_задачи [out|err|log]] –** выдаёт на экран соответствующие потоки вывода.

# Команды работы с очередями.

- **mqdel <имя_задачи.номер_задачи>** - удаление задачи из очереди.
- **mqinfo** – просмотр очереди (наиболее полная информация.)
- **mqtest <имя_задачи>** - проверка нахождения задачи в очереди.

# Монопольный доступ к вычислительным модулям.

- **getnodes –np <число_модулей> -maxtime <время> <имя_запроса>** - запрос ставится в очередь, затем после прохождения очереди соответствующие узлы будут доступны по rsh. Прекратить их использовать можно по команде **mkill.**

# Кластер ВМК
# cluster.cmc.msu.ru

# Базовые концепции

- Имеются 2 очереди для задач откомпилированных с использованием LAM и задач с использованием MPICH
- Статические приоритеты задач пользователей

# Запуск задач.

**mpirun -np N [-q queue][-maxtime lim][-l lim][-p pri] command**

- **-np** Количество процессоров
- **-q** Название очереди, в которую ставится задача
- **-maxtime** Лимит времени счета в минутах
- **-l** maxtime в секундах
- **-p** Приоритет задачи в очереди

# Просмотр состояния

**tasks [-q queue][-l][-f][-o][-m mask][-u userlist][-b]**

- **-q** Название очереди
- **-l** Показывать дополнительную информацию
- **-f** Учитывать чужие задачи
- **-o** Учитывать свои задачи
- **-m mask** Использовать маску для выборки задач
- **-u list** Использовать список пользователей для выборки задач
- **-b** Показывать информацию о заблокированных узлах.

# Удаление задачи из очереди

**tasks –d id**

- Удаляет задачу с указанным id из очереди.
- Если в качестве id указанно all удаляются все задачи, возможно сочетание с используемой маской.