

Министерство образования и науки Российской Федерации

Федеральное государственное бюджетное образовательное учреждение
высшего профессионального образования
«Оренбургский государственный университет»

Кафедра информационных систем и технологий

В.В. Извозчикова

КОМПИЛЯЦИЯ И ЗАПУСК ПРОГРАММ В КЛАСТЕРНЫХ СИСТЕМАХ

Рекомендовано к изданию Редакционно-издательским советом федерального государственного бюджетного образовательного учреждения высшего профессионального образования «Оренбургский государственный университет» в качестве методических указаний для студентов, обучающихся по программам высшего образования по направлению подготовки 09.03.02, 09.04.02 Информационные системы и технологии

Оренбург
2015

УДК 004.3/.4'23(075.8)

ББК 32.973-018.я73

ИЗ4

Рецензент – профессор, доктор экономических наук, В.Н. Шепель

ИЗ4

Извозчикова, В.В.

Компиляция и запуск программ в кластерных системах: методические указания к лабораторным работам / В.В. Извозчикова. –Оренбургский гос. ун.-т. – Оренбург: ОГУ, 2015. – 36 с.

Лабораторный практикум состоит из трех лабораторных работ и посвящен изучению вопросов компиляции и запуска программ на кластерных системах. Каждая лабораторная работа содержит подробную методику выполнения работы с наглядными иллюстрациями.

Методические указания к лабораторным работам позволяют изучить и приобрести навыки в работе с интегрированной средой разработки Microsoft Visual Studio 2008, библиотекой для разработки параллельных программ Microsoft MPI и системой управления кластером Microsoft High Performance Computing Server 2008. Предназначены для выполнения лабораторных работ по дисциплинам «Высокопроизводительные вычислительные системы» для бакалавров по направлению 09.03.02 Информационные системы и технологии очной и заочной форм обучения и «Современные инструментальные средства в научных исследованиях» для магистров по направлению 09.04.02 Информационные системы и технологии магистерской программы «Информационные системы в научных исследованиях».

УДК 004.3/.4'23(075.3)

ББК 32.973-018.я73

© Извозчикова В.В., 2015

© ОГУ, 2015

Содержание

Введение	4
1 Лабораторная работа №1. Компиляция программы для запуска в HPC 2008	6
1.1 Установка Microsoft High Performance Computing Pack 2008 R2 Client	6
1.2 Установка Microsoft High Performance Computing Pack 2008 R2 MS-MPI.....	8
1.3 Установка Microsoft High Performance Computing SDK.....	9
1.4 Настройка интегрированной среды разработки Microsoft Visual Studio 2008 ..	11
2 Лабораторная работа №2. Установка и настройка кластера.....	13
2.1 Подготовка к развертыванию кластера.....	13
2.2 Установка головного узла	14
2.3 Создание шаблона для автоматической установки и настройки вычислительных узлов.....	22
2.4 Установка вычислительных узлов.....	24
3 Лабораторная работа №3. Запуск задач на кластере	27
3.1 Запуск последовательной задачи.....	27
3.2 Запуск параллельной задачи	31
Список использованных источников	36

Введение

В настоящее время приходится решать все более сложные задачи, которые требуют для своего решения мощных вычислительных ресурсов, в то время дальнейшее наращивание скорости работы индивидуальных устройств становится практически невозможным. В связи с этим развитие вычислительной техники пошло по экстенсивному пути, основанному на использовании вычислительных устройств, которые способны выполнять параллельно этапы одной и той же общей задачи. Вместе с этим родилось параллельное программирование, призванное дать возможность эффективно использовать параллельные архитектуры. Поэтому сегодня разработчики программных систем используют параллелизм на всех уровнях, начиная от нескольких конвейеров суперскалярных процессоров, и заканчивая параллельно работающими вычислительными узлами в GRID, которые представляют собой форму распределённых вычислений, в которой «виртуальный суперкомпьютер» представлен в виде кластеров, соединённых с помощью сети, слабосвязанных гетерогенных компьютеров, работающих вместе для выполнения огромного количества заданий.

Отдельный класс параллельных архитектур представляют кластерные системы, которые представляют собой совокупность вычислительных узлов, объединённых сетью, а параллельное приложение для кластерной системы представляет собой несколько процессов, которые общаются друг с другом по сети. Таким образом, если пользователь сумеет эффективно распределить свою задачу между несколькими процессорами на узлах кластера, то он может получить выигрыш в скорости работы, почти пропорциональный числу процессоров.

Для эффективной эксплуатации высокопроизводительных кластерных систем необходимо использовать сложный комплекс программных средств. Долгое время пользователям Windows кластеров приходилось одновременно использовать программное обеспечение нескольких производителей, что вызывало проблемы с совместимостью различных программ друг с другом. С

выходом системы управления кластером Microsoft High Performance Computing Server 2008 (HPC 2008) компания Microsoft предоставляет полный спектр программного обеспечения, необходимый для эффективной эксплуатации кластера и разработки программ, в полной мере использующих имеющиеся вычислительные мощности.

Для эффективного использования вычислительного ресурса кластера необходимо обеспечить не только непосредственный механизм запуска заданий на выполнение, но и предоставить среду управления ходом выполнения заданий, решающую, в том числе, задачу эффективного распределения ресурсов. Эти задачи эффективно решаются с использованием встроенных в HPC 2008 средств.

1 Лабораторная работа №1. Компиляция программы для запуска в HPC 2008

Цель работы: научиться устанавливать необходимое программное обеспечение и компилировать программы под управлением Microsoft High Performance Computing Server 2008 (HPC 2008).

Microsoft High Performance Computing Server 2008 (HPC 2008) позволяет управлять ходом выполнения как последовательных, так и параллельных задач. При этом параллельные MPI задачи не обязательно должны быть скомпилированы для MS MPI (хотя в случае MPI использование реализации от Microsoft является предпочтительным). Кроме того, возможно использование других технологий поддержки параллельного программирования (например, программирование с использованием OpenMP).

В данной работе необходимо выполнить компиляцию параллельных программ для MS MPI в Microsoft Visual Studio 2008.

1.1 Установка Microsoft High Performance Computing Pack 2008 R2 Client

Для компиляции параллельных программ, работающих в среде MS MPI, необходимо установить Microsoft High Performance Computing Pack 2008 R2 Client, для чего необходимо выполнить следующую последовательность действий:

а) откройте директорию со скачанной версией HPC Client и запустите программу установки, соответствующую Вашему процессору (32-битная или 64-битная версия) согласно рисунку 1.1;

б) в открывшемся окне нажмите кнопку " Next " для начала установки;

г) внимательно прочитайте лицензионное соглашение. Выберите пункт " I accept the terms in the license agreement " в случае согласия с лицензионным соглашением и нажмите кнопку " Next ";

д) выберите директорию, в которую будет установлен HPCClient. Для изменения стандартной директории нажмите кнопку " Change ". Нажмите кнопку " Next ";

е) в следующем окне Вас попросят выбрать обновления Microsoft. Исходя из вашего выбора, поставьте галочку в нужном месте и нажмите «Next» согласно рисунку 1.2;

д) завершите установку HPCClient.

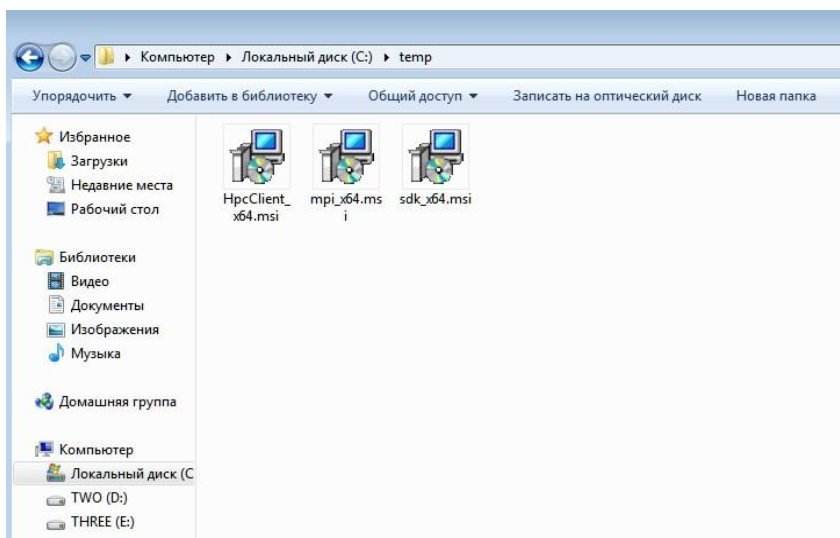


Рисунок 1.1 - Рабочая папка

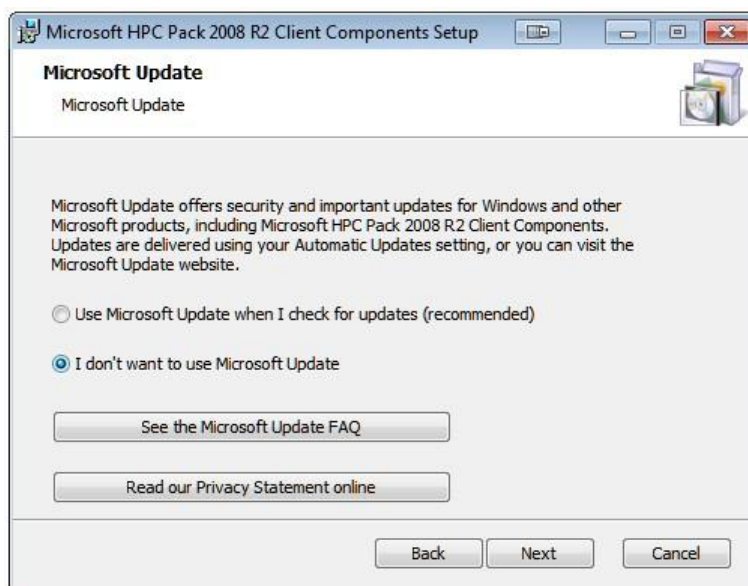


Рисунок 1.2 - Установка HPCPack

1.2 Установка Microsoft High Performance Computing Pack 2008 R2

MS-MPI

Для компиляции параллельных программ, работающих в среде MS MPI, также необходимо установить Microsoft High Performance Computing Pack 2008 R2 MS-MPI, выполнив следующую последовательность действий:

- а) откройте директорию со скачанной версией MS-MPI и запустите программу установки, соответствующую Вашему процессору (32-битная или 64-битная версия);
- в) в открывшемся окне нажмите кнопку " Next " для начала установки;
- г) внимательно прочитайте лицензионное соглашение. Выберите пункт " I accept the terms in the license agreement " в случае согласия с лицензионным соглашением и нажмите кнопку " Next ";
- д) выберите директорию, в которую будет установлен MS-MPI . Для изменения стандартной директории нажмите кнопку " Change ". Нажмите кнопку " Next ";
- е) в следующем окне Вас попросят выбрать об обновлениях Microsoft. Исходя из вашего выбора поставьте галочку в нужном месте и нажмите «Next»;
- ж) завершите установку MS-MPI и на экране появится изображение, приведенное на рисунке 1.3 .

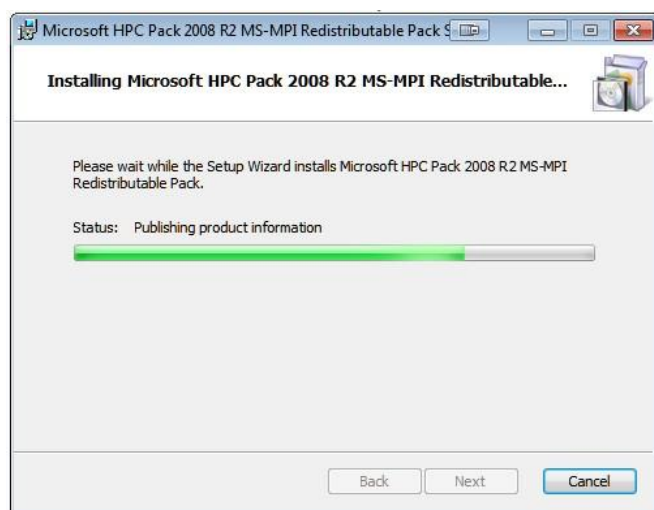


Рисунок 1.3 - Установка HPC Pack

1.3 Установка Microsoft High Performance Computing SDK

Для компиляции параллельных программ, работающих в среде MS MPI, необходимо установить SDK (Software Development Kit) - набор интерфейсов и библиотек для вызова MPI-функций, согласно следующей последовательности действий:

а) откройте директорию со скачанной версией SDK и запустите программу установки, соответствующую Вашему процессору (32-битная или 64-битная версия);

в) в открывшемся окне нажмите кнопку " **Next** " для начала установки;

г) внимательно прочитайте лицензионное соглашение. Выберите пункт " **I accept the terms in the license agreement** " в случае согласия с лицензионным соглашением и нажмите кнопку " **Next** ";

д) выберите директорию, в которую будет установлен SDK. Для изменения стандартной директории нажмите кнопку " **Change** ". Нажмите кнопку " **Next** ", согласно рисунку 1.4;

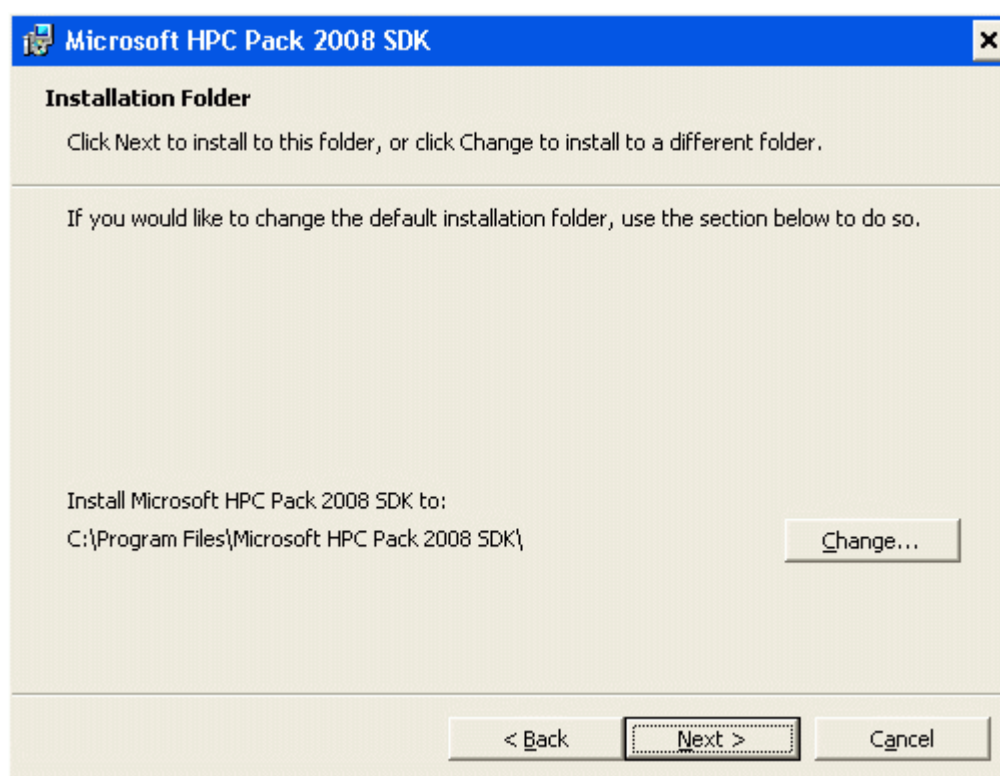


Рисунок 1.4 - Установка SDK

е) в открывшемся окне нажмите кнопку " **Install** " для начала установки SDK, согласно рисунку 1.5;

Рисунок 6 - Установка SDK

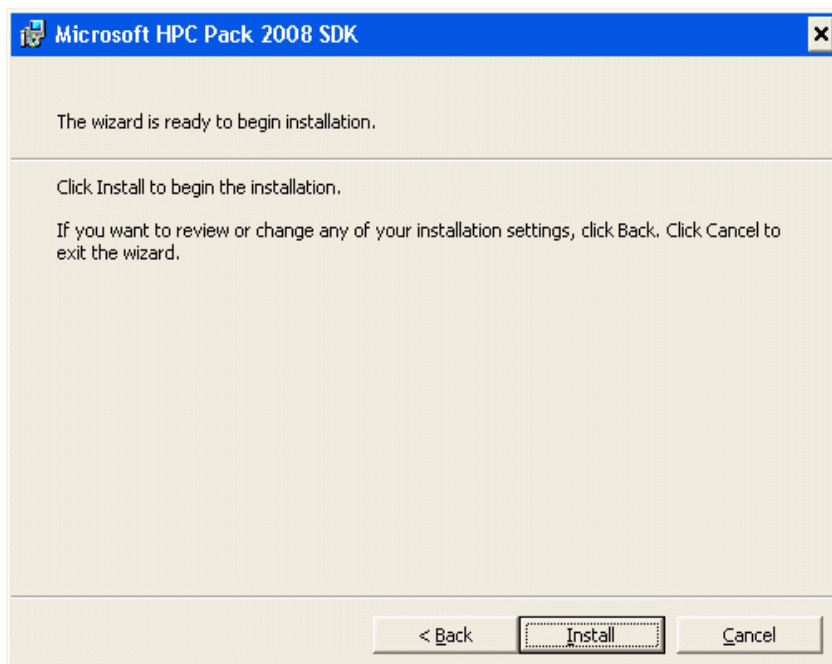


Рисунок 1.5 - Установка SDK

ж) дождитесь пока программа установки SDK скопирует требуемые файлы.

1.4 Настройка интегрированной среды разработки Microsoft Visual Studio 2008

Для того, чтобы скомпилировать Вашу программу для использования в среде MS MPI, необходимо изменить следующие настройки проекта по умолчанию в Microsoft Visual Studio 2008:

Для установки пути до заголовочных файлов объявлений MPI выберите пункт меню **Project->Project Properties**. В пункте **Configuration Properties->C++->General->Additional Include Directories** введите путь до заголовочных файлов MS MPI: **C:\Program Files\Microsoft HPC Pack 2008 R2\Inc**, как показано на рисунке 1.6.

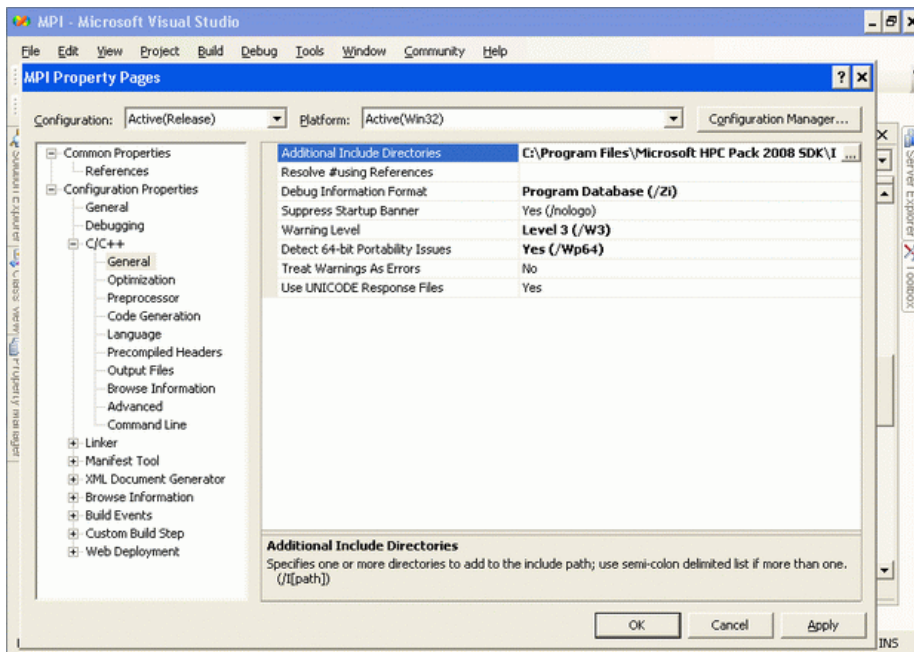


Рисунок 1.6 - Установка пути до заголовочных файлов MPI

Для подключения библиотечного файла с реализацией функций MPI выберите пункт меню **Project->Project Properties**. В пункте **Configuration Properties->C++->Linker->Input->Additional Dependencies** введите название библиотечного файла `mmpi.lib`, как показано на рисунке 1.7.

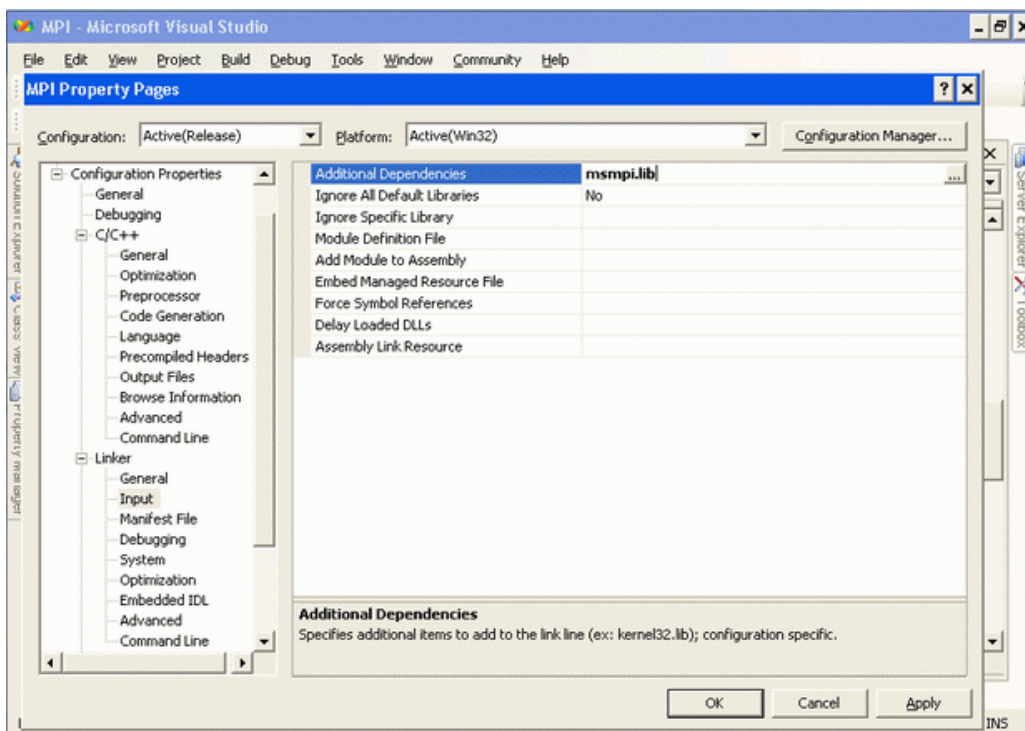
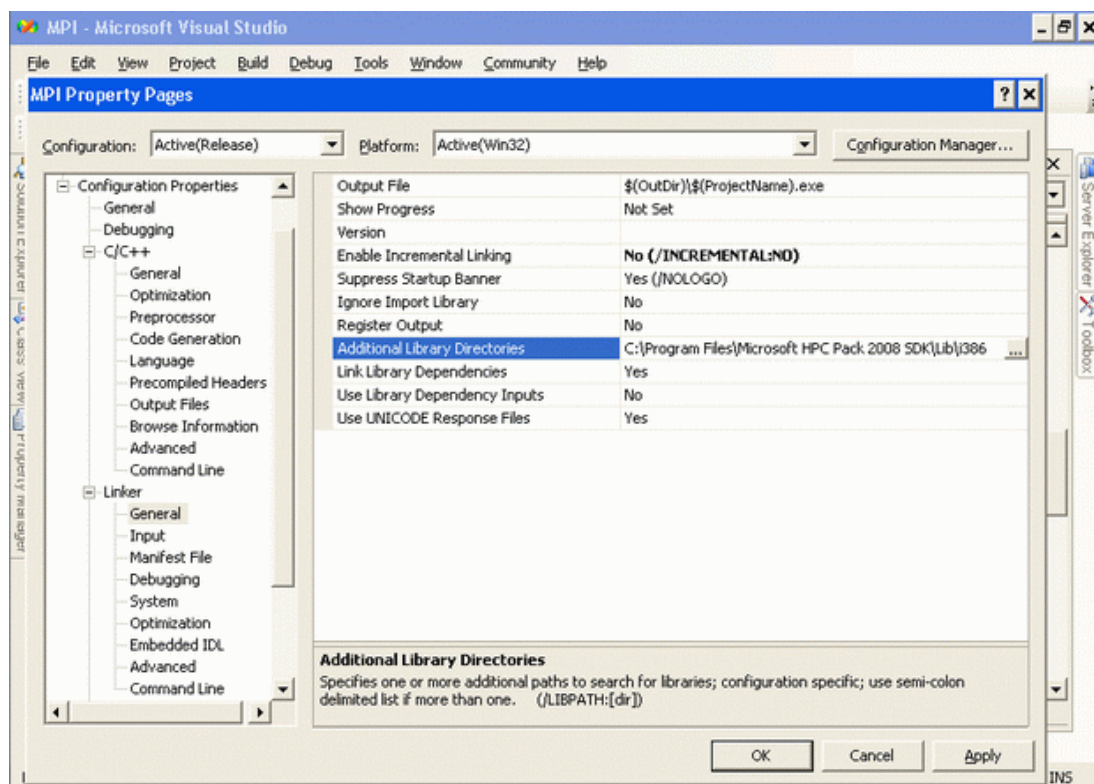


Рисунок 1.7- Подключение библиотечного файла

Для установки пути до библиотечного файла **msmpi.lib** выберите пункт меню **Project->Project Properties**. В пункте **Configuration Properties->C++->Linker->General->Additional Library Directories** введите путь до библиотечного файла **msmpi.lib**: **C:\Program Files\Microsoft HPC Pack 2008 R2\Lib\i386** или **< Директория установки HPC SDK>\Lib\AMD64** в зависимости от используемой Вами версии **Microsoft Visual Studio 2008(x86 или x64)**, как показано на рисунке 1.8.



Рисуну

к 1.8 - Установка пути до библиотечного файла **msmpi.lib**

2 Лабораторная работа №2. Установка и настройка кластера

Цель работы: Установка и настройка узлов кластера

2.1 Подготовка к развёртыванию кластера

Чтобы реализовать простейший кластер, необходимо иметь как минимум две рабочие станции, одна из которых будет главным узлом. Все эти станции необходимо объединить в локальную сеть. Для реализации кластера лучше всего подходят две топологии:

- вычислительные узлы и рабочие станции, с которых будут использовать кластер, расположены в разных сетях;
- все рабочие станции, в том числе и узлы кластера, находятся в одной сети.

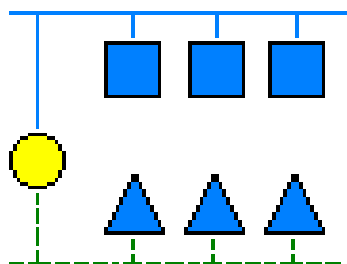


Рисунок 2.1 – Первая топология

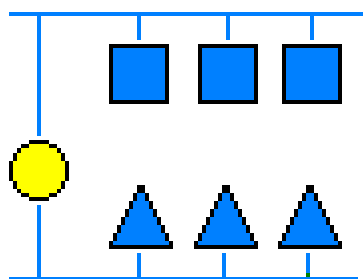


Рисунок 2.2 – Вторая топология

В случае наличия нескольких сетевых карт на компьютере, который будет являться главным узлом, можно использовать первую топологию. Если же на компьютере всего один сетевой адаптер – выбираем вторую. В данной инструкции все рабочие станции объединены в одну сеть (вторая топология).

После выбора топологии необходимо определиться с диапазоном IP-адресов. К примеру, адреса 192.168.10.1 – 192.168.10.20 будем использовать для узлов кластера, а 192.168.10.21 – 192.168.10.254 для клиентских машин.

2.2 Установка головного узла

Для работы кластера необходимо установить Microsoft HPC Pack 2008 R2. Установка запускается с файла setup.exe, который находится в папке HPCPack.Express. В процессе установки необходимо выбрать версию HPC Pack 2008 R2 Express, тип узла - главный (head node), указать директории для баз данных и т.д. Процесс установки представлен на рисунках с 2.3 по 2.11.

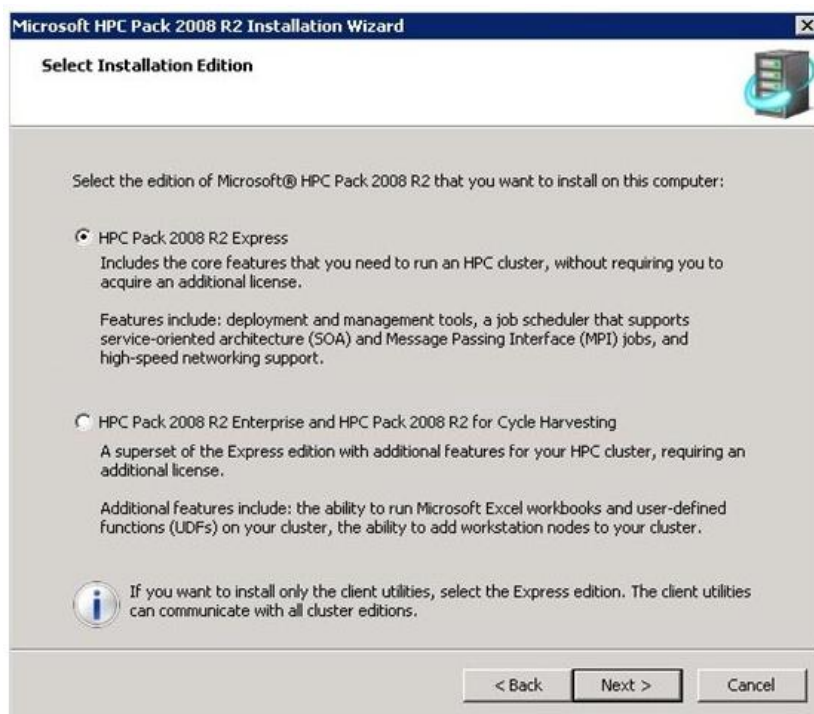


Рисунок 2.3 – Выбор версии

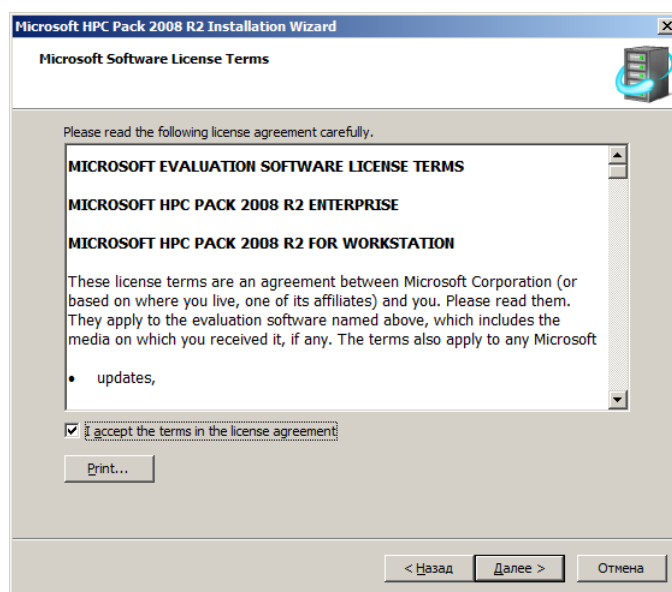


Рисунок 2.4 - Соглашение с условиями лицензии

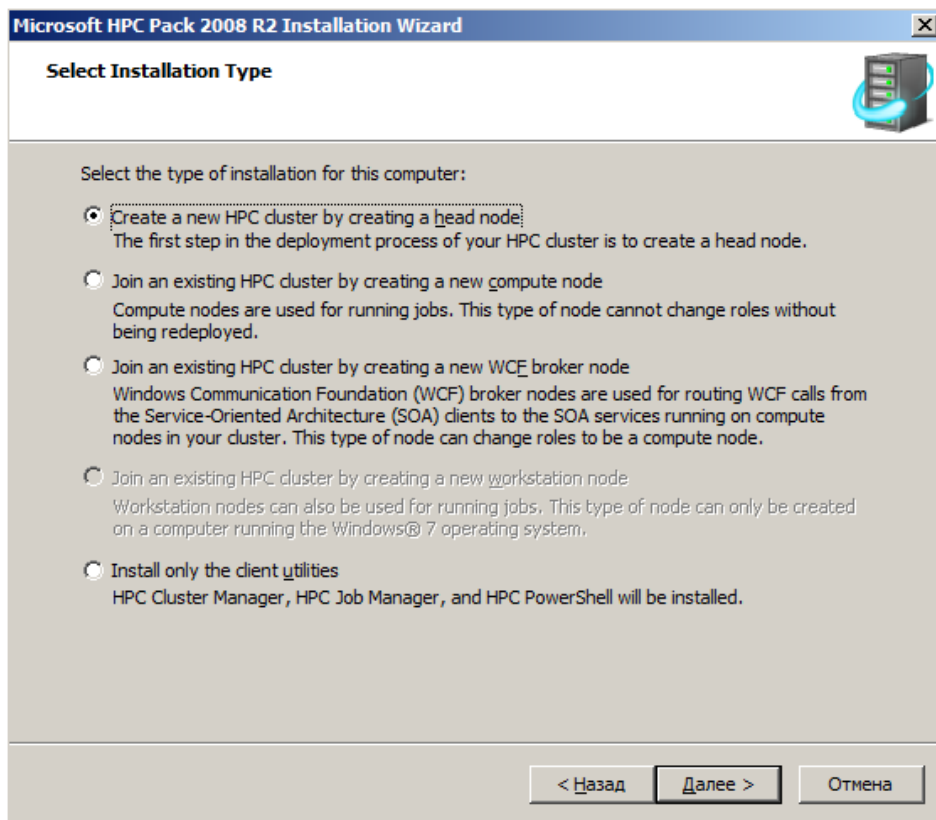


Рисунок 2.5 - Выбор типа узла

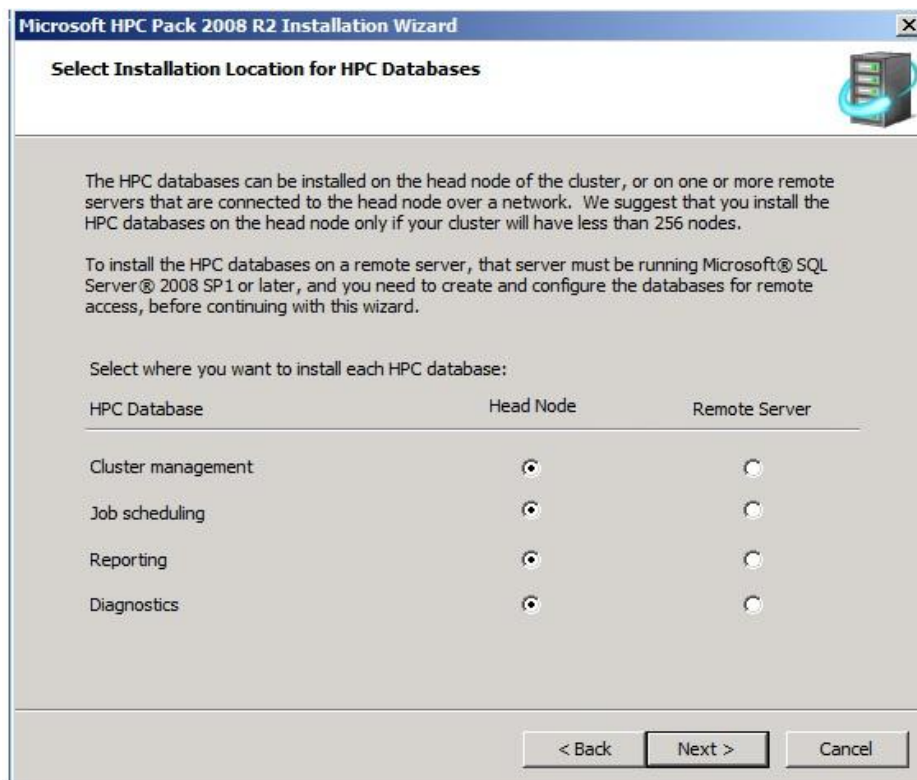


Рисунок 2.6 – Выбор места расположения HPC database

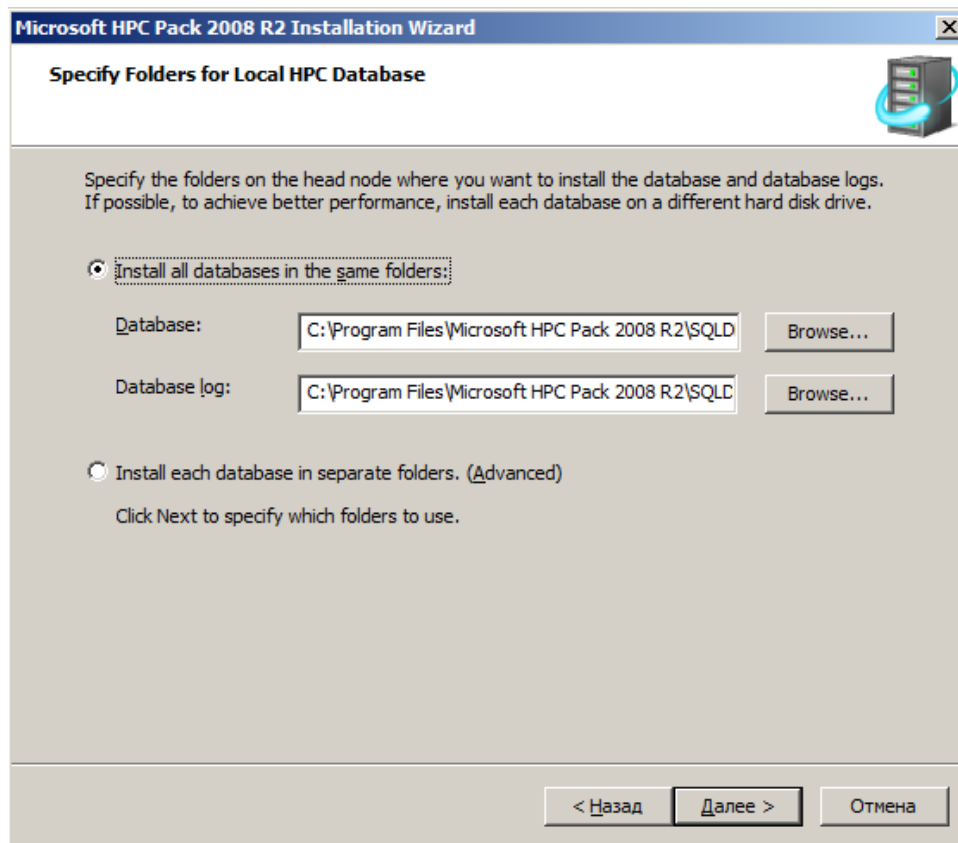


Рисунок 2.7 - Выбор директорий для баз данных

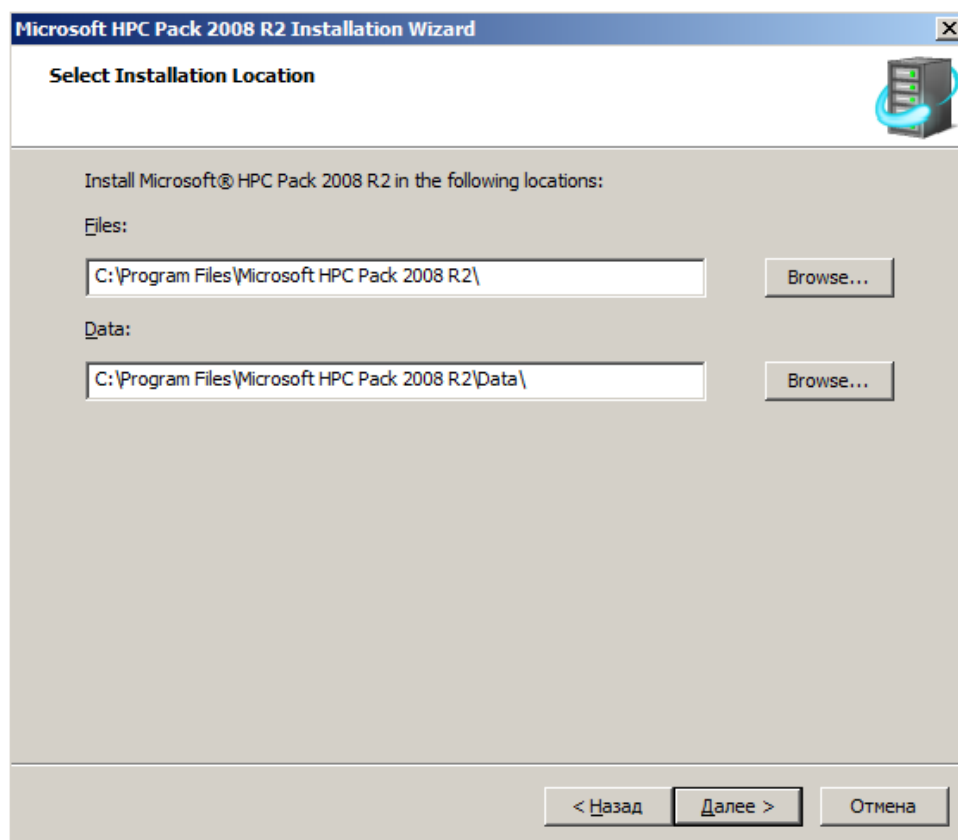


Рисунок 2.8 - Выбор директории для установки HPC Pack

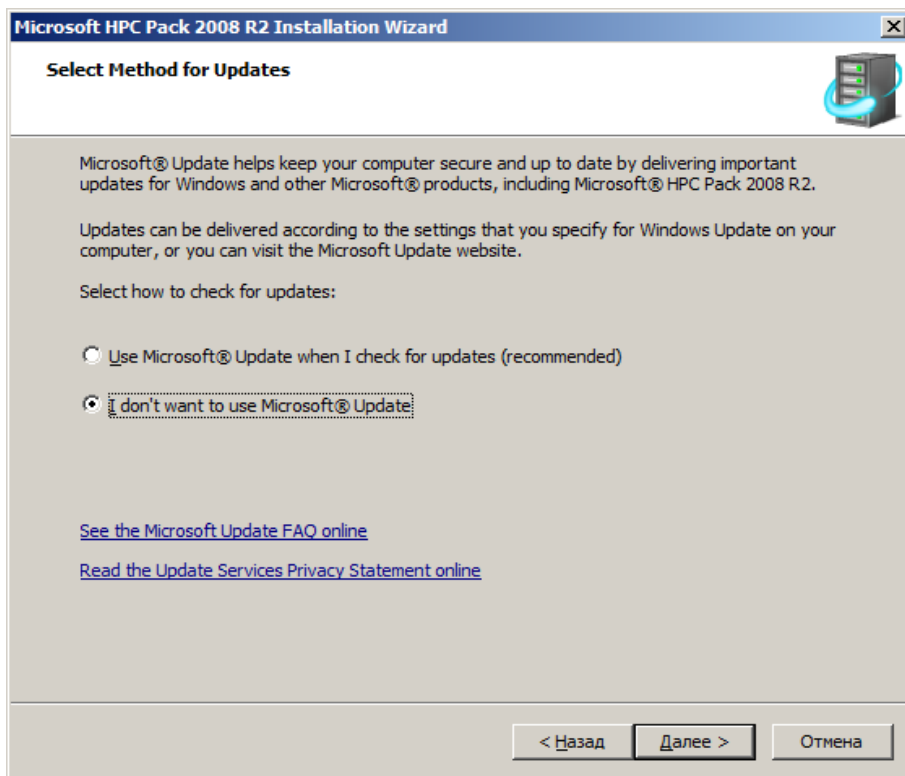


Рисунок 2.9 - Использование обновления Microsoft

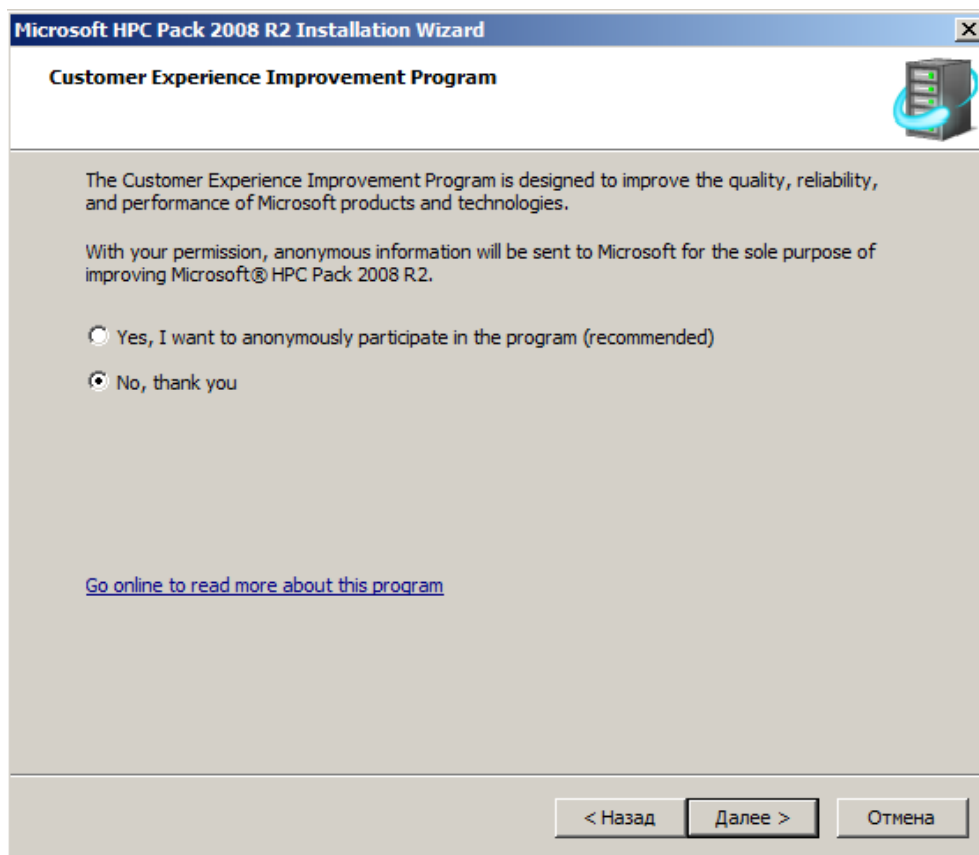


Рисунок 2.10 - Получение информации от Microsoft

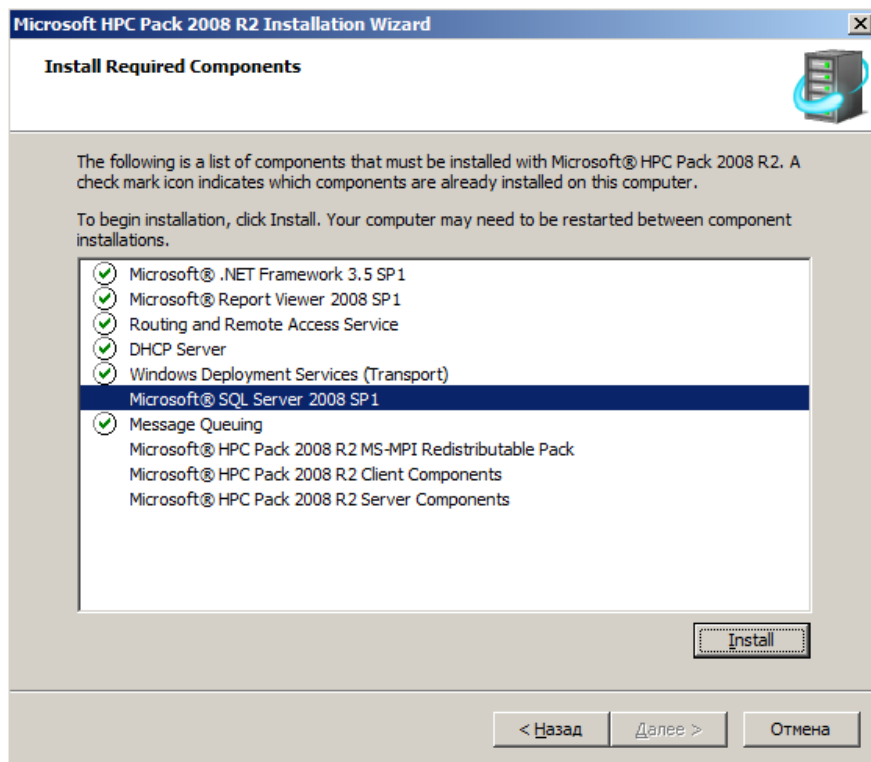


Рисунок 2.11 – Список устанавливаемых компонентов

После установки необходимо произвести первоначальную настройку кластера. В окне To-do List нажимаем Configure your network, согласно рисунку 2.12.

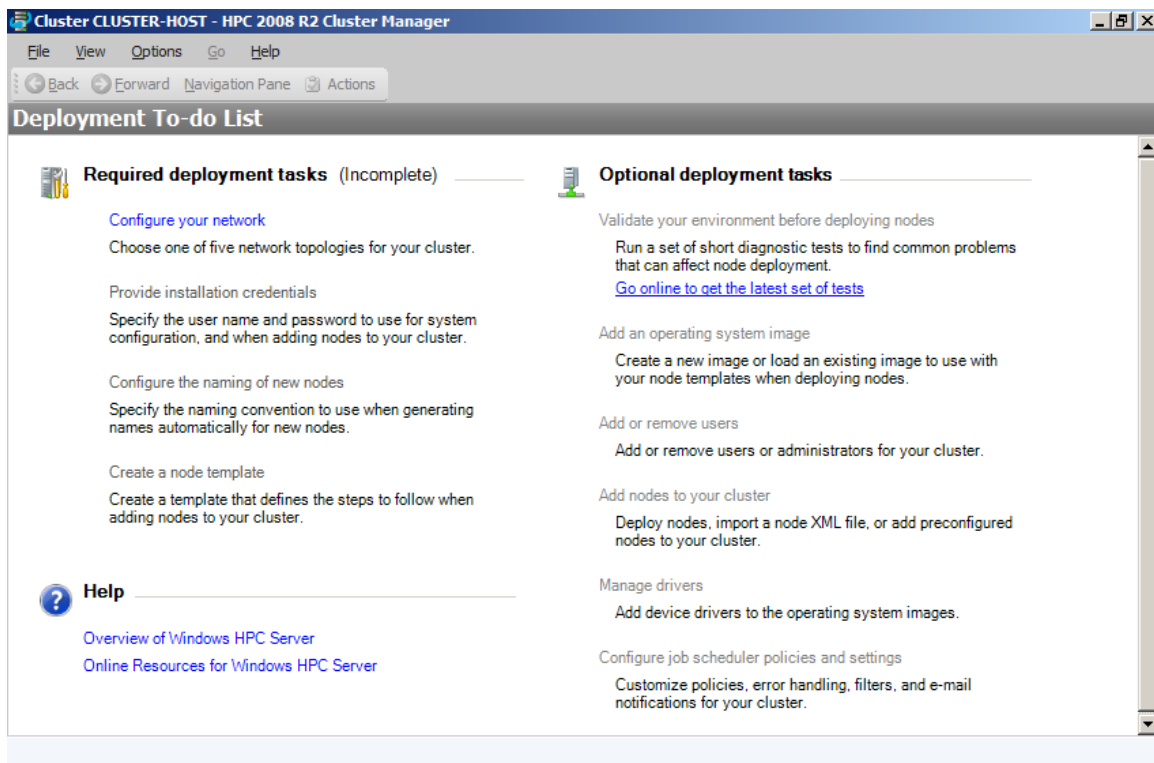


Рисунок 2.12 - Окно To-do list

Выбираем последнюю топологию, так как наши компьютеры объединены простой локальной сетью, как показано на рисунке 2.13.

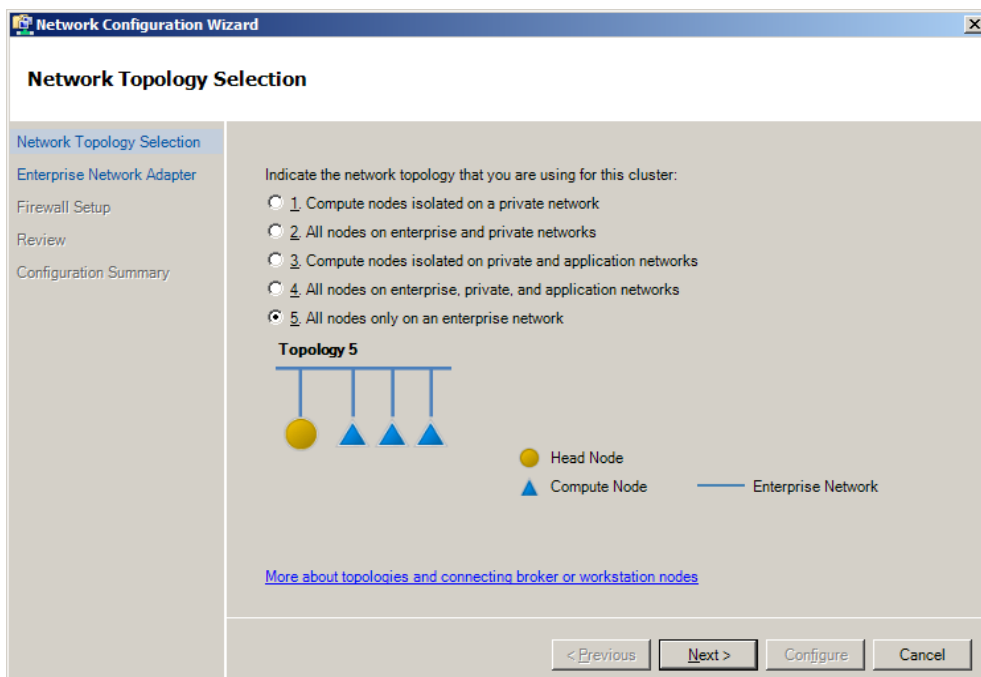


Рисунок 2.13 - Выбор топологии

Выбираем сетевой адаптер нашей локальной сети. Перед выбором, его необходимо включить и проверить настройки, как показано на рисунке 2.14.

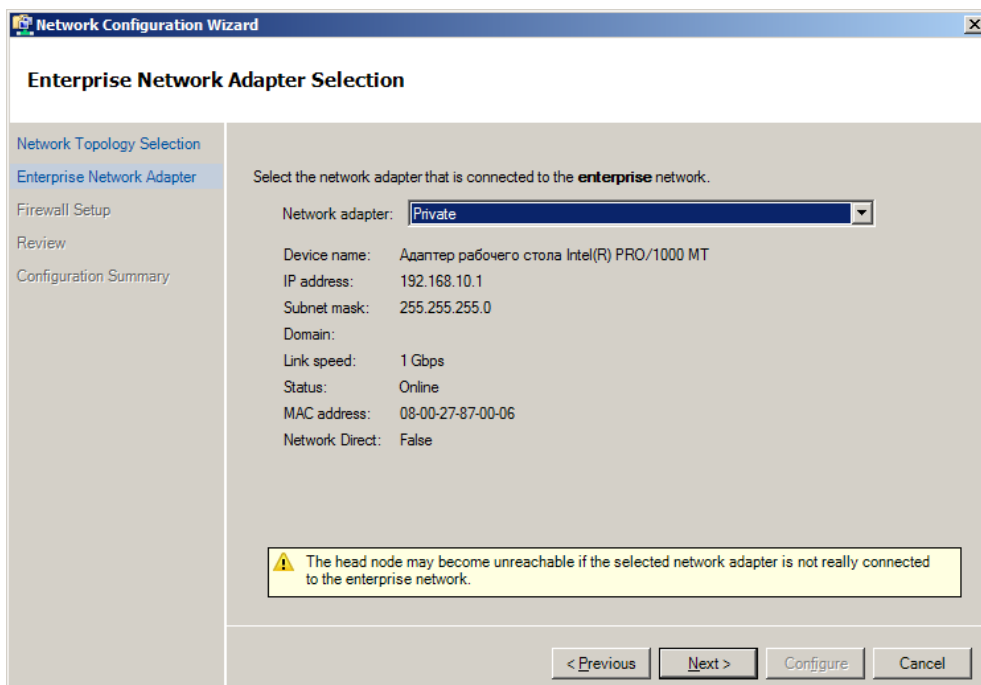


Рисунок 2.14 - Выбор сетевого адаптера

Выбираем настройки фаервола, как на рисунке 2.15.

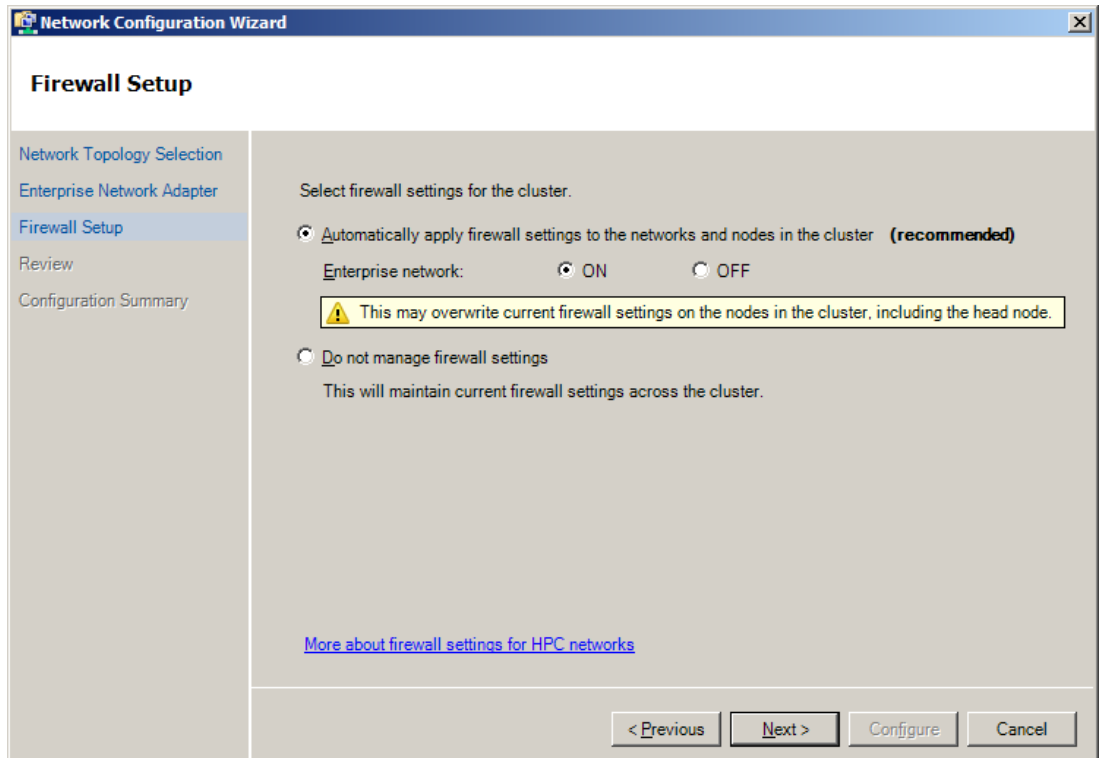


Рисунок 2.15 - Настройки фаервола

Ознакомившись с выходной информацией, представленной на рисунке 2.16, нажимаем Configure.

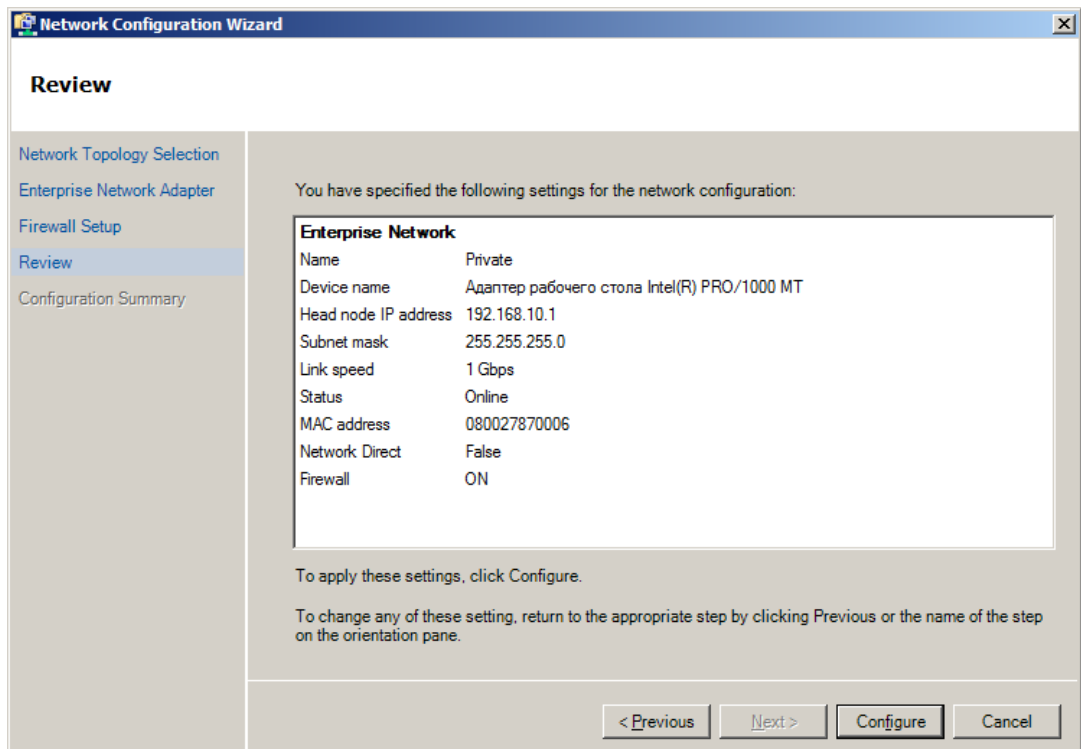


Рисунок 2.16 - Сводка

После настройки сети кластера мы снова увидим окно To-do List. Теперь необходимо указать учётную запись администратора кластера. Нажимаем Provide installation credentials. В появившемся окне, представленном на рисунке 2.17, указываем учётную запись и вводим пароль.

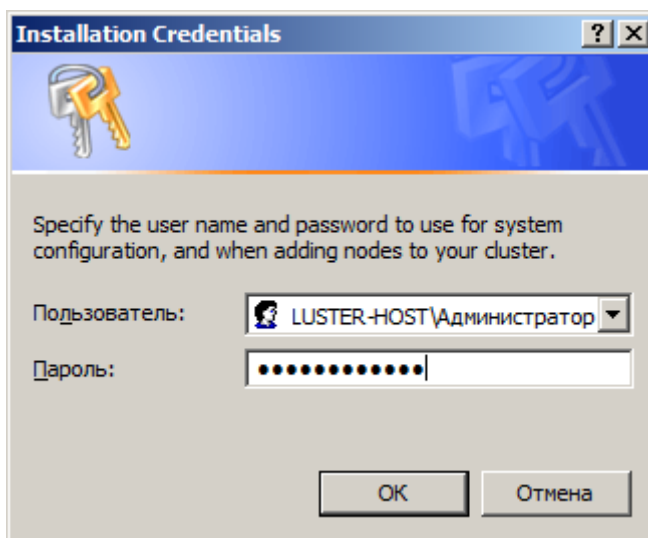


Рисунок 2.17 - Выбор учётной записи

После указания учётной записи администратора необходимо выбрать спецификацию, по которой будут именоваться узлы кластера. В окне To-do List нажимаем Configure the naming of new nodes. Вводим спецификацию в поле Naming Series согласно рисунку 2.18 и ниже будет показан пример формирования имён по введённым данным.

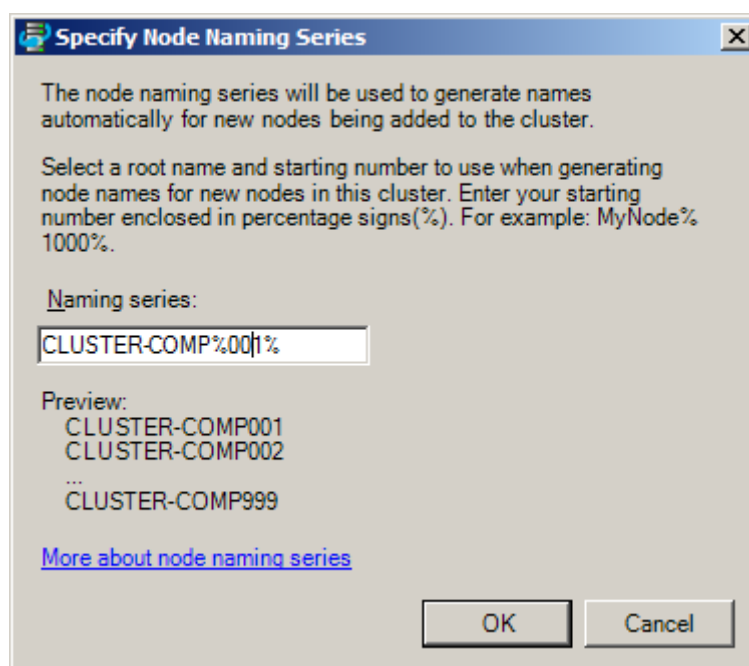


Рисунок 2.18 - Окно To-do list

2.3 Создание шаблона для автоматической установки и настройки вычислительных узлов

Теперь остаётся только создать шаблон для автоматической установки и настройки вычислительных узлов. В окне To-do List нажимаем «Create a node template». Появится мастер создания шаблонов, показанный на рисунке 2.19. В первом окне необходимо выбрать тип шаблона для вычислительных узлов.

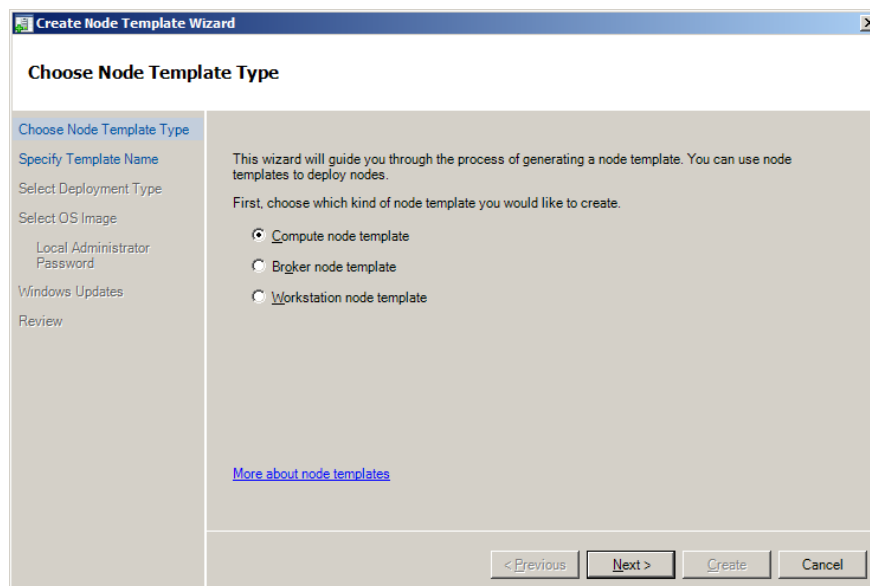


Рисунок 2.19 - Мастер создания шаблона

Вводим имя шаблона и его описание. Пример представлен на рисунке 2.20.

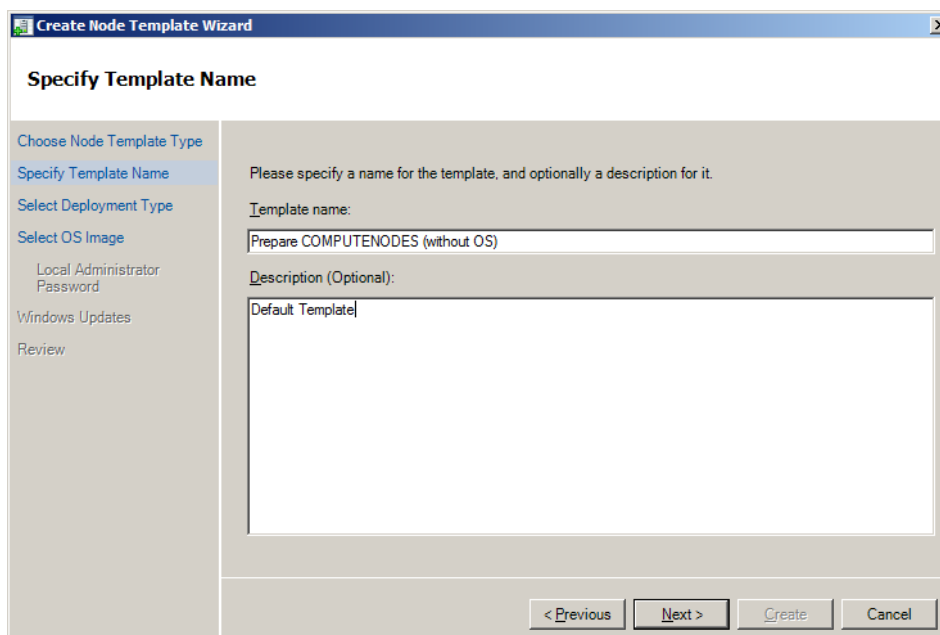


Рисунок 2.20 - Указание имени шаблона и описания

Далее указываем, будет ли устанавливаться с шаблона операционная система или нет. Первый пункт следует выбирать в случае автоматической установки всего необходимого ПО, в том числе и операционной системы. Если предполагается установка ОС вручную - выбираем второй пункт. В случае выбора первого пункта - нужно будет указать образ диска с ОС и ввести данные учётной записи администратора. Данная информация приведена на рисунке 2.21.

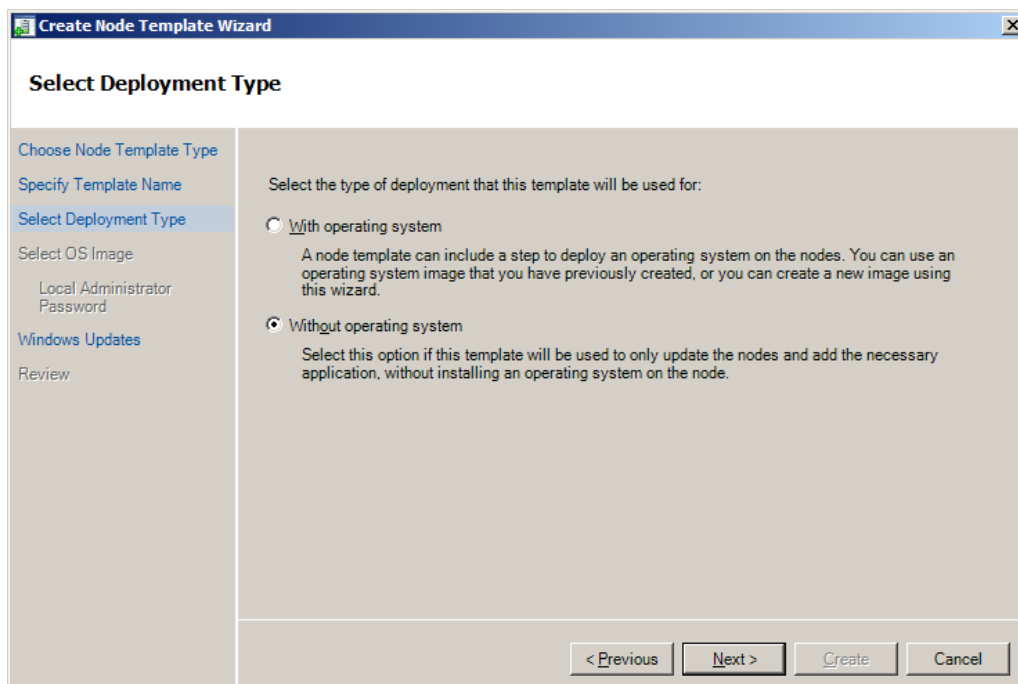


Рисунок 2.21 - Мастер создания шаблона

Так же указываем, стоит ли включать в шаблон обновления для ОС или нет. Выбор представлен на рисунке 2.22.

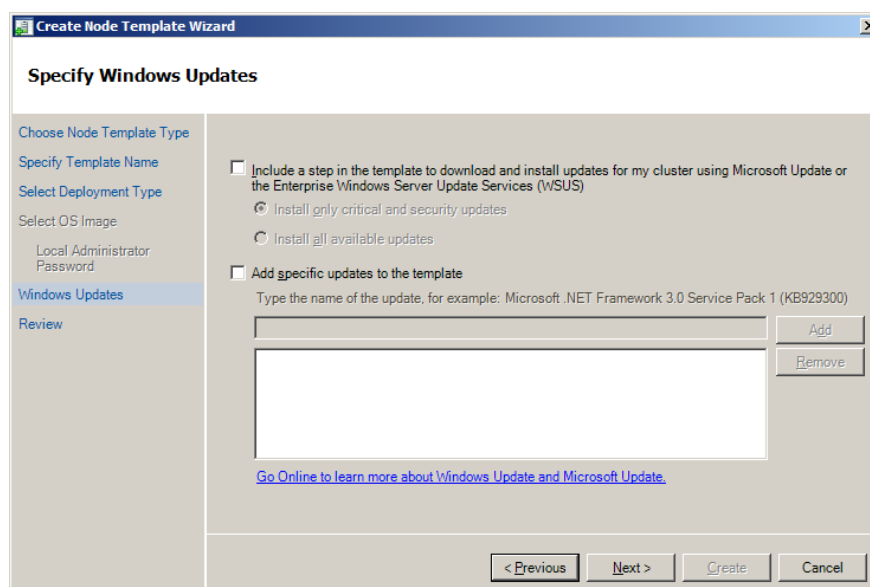


Рисунок 2.22 - Обновления для ОС

После ознакомления с выходной информацией нажимаем Create. В результате получим выходную информацию, показанную на рисунке 2.23.

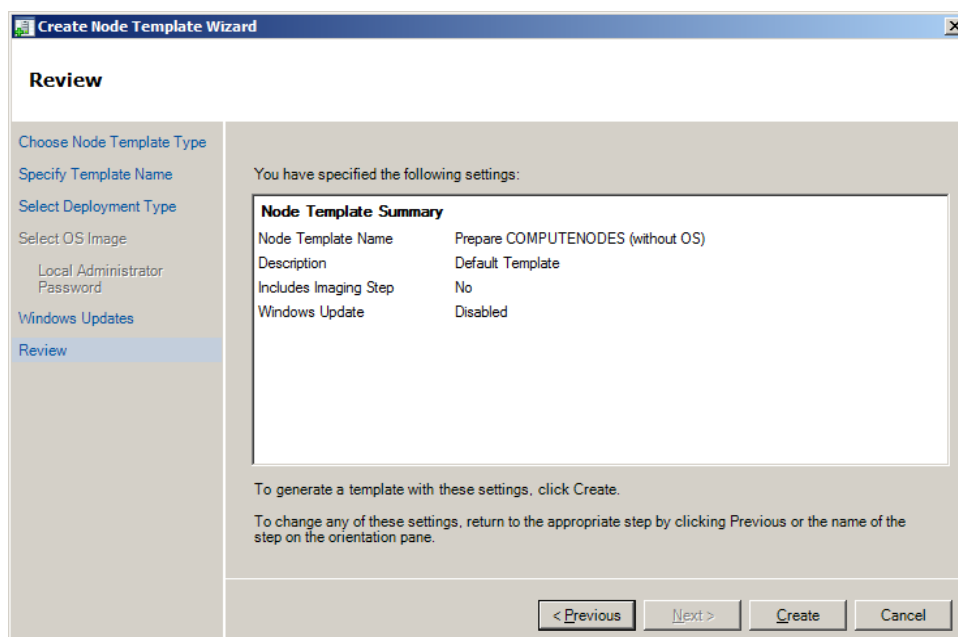


Рисунок 2.23 - Выходная информация

2.4 Установка вычислительных узлов

Теперь, когда главный узел кластера настроен, нужно ввести вычислительные узлы. На панели слева выбираем Node Management. Мы увидим таблицу, в которой содержится список узлов кластера. Данная таблица представлена на рисунке 2.24.

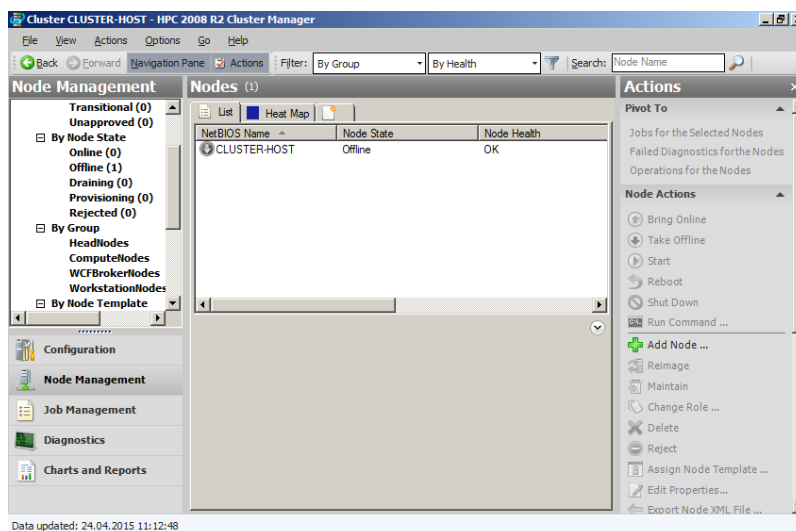


Рисунок 2.24 - Окно Cluster Manager

На данный момент у нас всего один узел - это главный. Включить узлы в кластер можно двумя способами, первый - просто включить компьютер, подключенной к сети и, если на компьютере доступна среда PXE, он появится в таблице в состоянии Unknown. Узлам необходимо присвоить шаблон (команда "AssignNodeTemplate"), после чего начнется установка (статус вычислительных узлов сменится на "Provisioning"). По окончании установки узлы будут добавлены в кластер. Второй способ - установить необходимое ПО вручную. Установка практически не отличается от установки главного узла. В процессе необходимо будет указать имя компьютера, который является главным узлом. Вычислительный узел, на котором идёт установка необходимого ПО, представлен на рисунке 2.25.

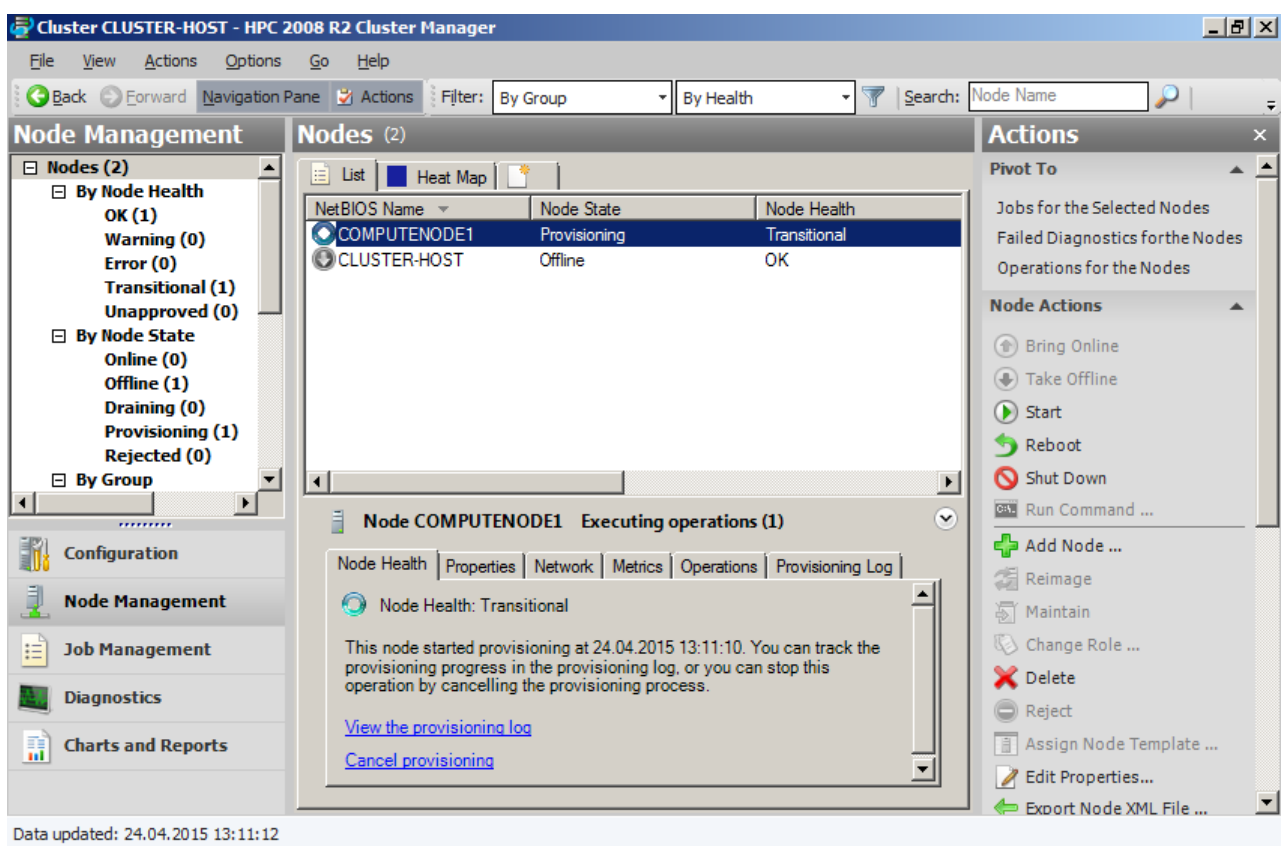


Рисунок 2.25 - Список узлов

После установки ПО, статус узлов станет Offline, а Node Health - ОК. Теперь для того, чтобы узлы могли участвовать в вычислениях, их нужно включить. Нажимаем правой кнопкой мыши на нужный узел и нажимаем Bring online, как показано на рисунке 2.26. С этого момента узел может участвовать в вычислениях.

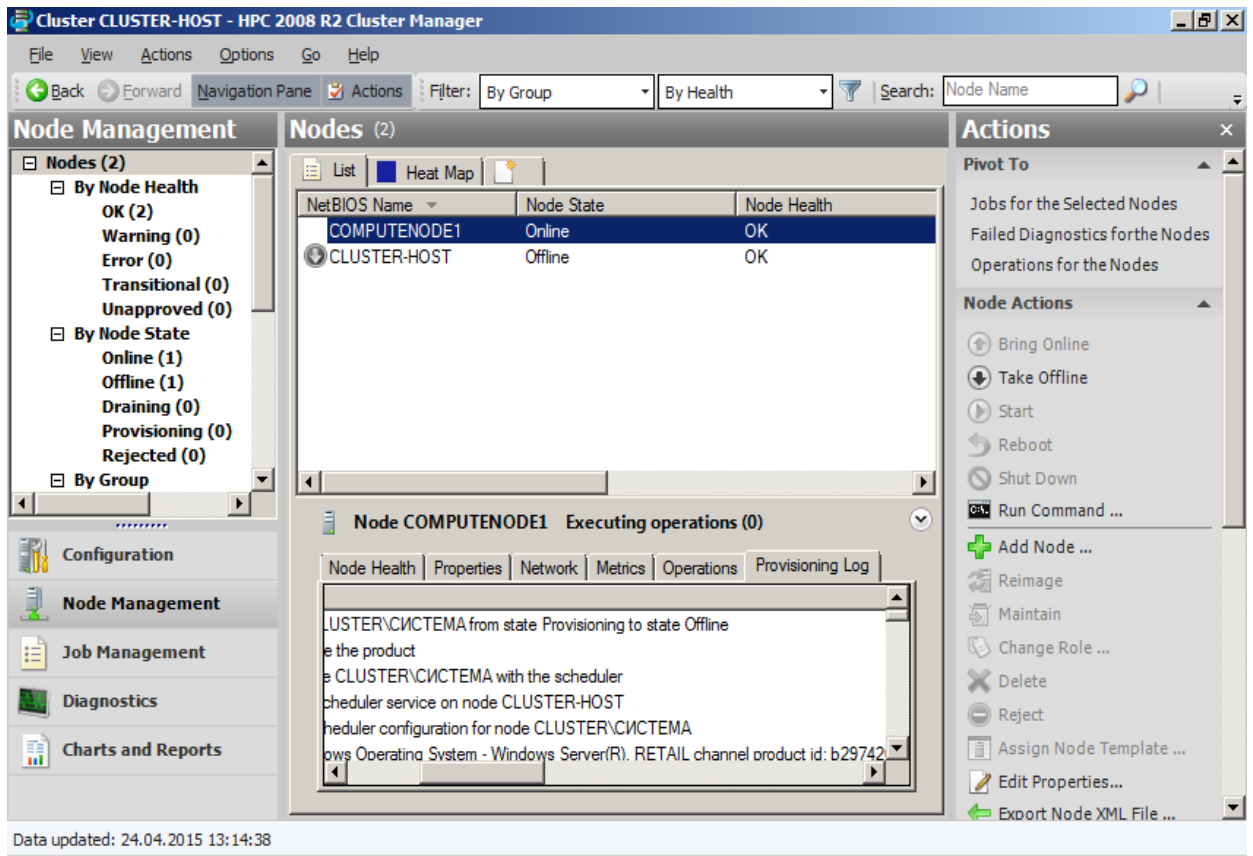


Рисунок 2.26 - Список узлов

3 Лабораторная работа №3. Запуск задач на кластере

Цель работы :научиться запускать программы под управлением Microsoft High Performance Computing Server 2008 (HPC 2008).

3.1 Запуск последовательной задачи

Последовательной называется задача, которая используется для своей работы ресурсы только одного процессора. Порядок компиляции последовательной программы (а также параллельной программы с использованием технологии OpenMP) для использования на кластере под управлением Microsoft HPC 2008 не отличается от обычного и не требует использования дополнительных библиотек. В данном задании рассматривается порядок запуска последовательной задачи на кластере. На рисунок 3.1 показано окно настройки шаблона заданий.

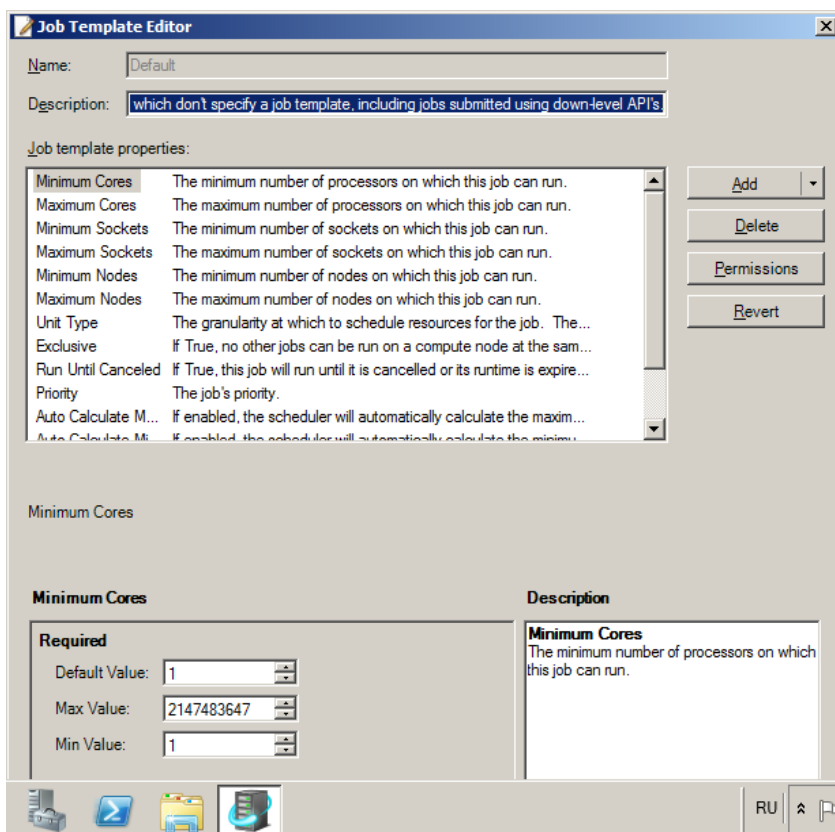


Рисунок 3.1 - Настройка шаблона заданий

Перед использованием кластера необходимо на главном узле создать общие папки, куда будут помещаться исполняемые модули программы, файлы с результатами и отчёты об ошибках.

Для запуска программы на кластере, открываем на клиентской рабочей станции HPC Job Manager. Окно программы представлено на рисунке 3.2.

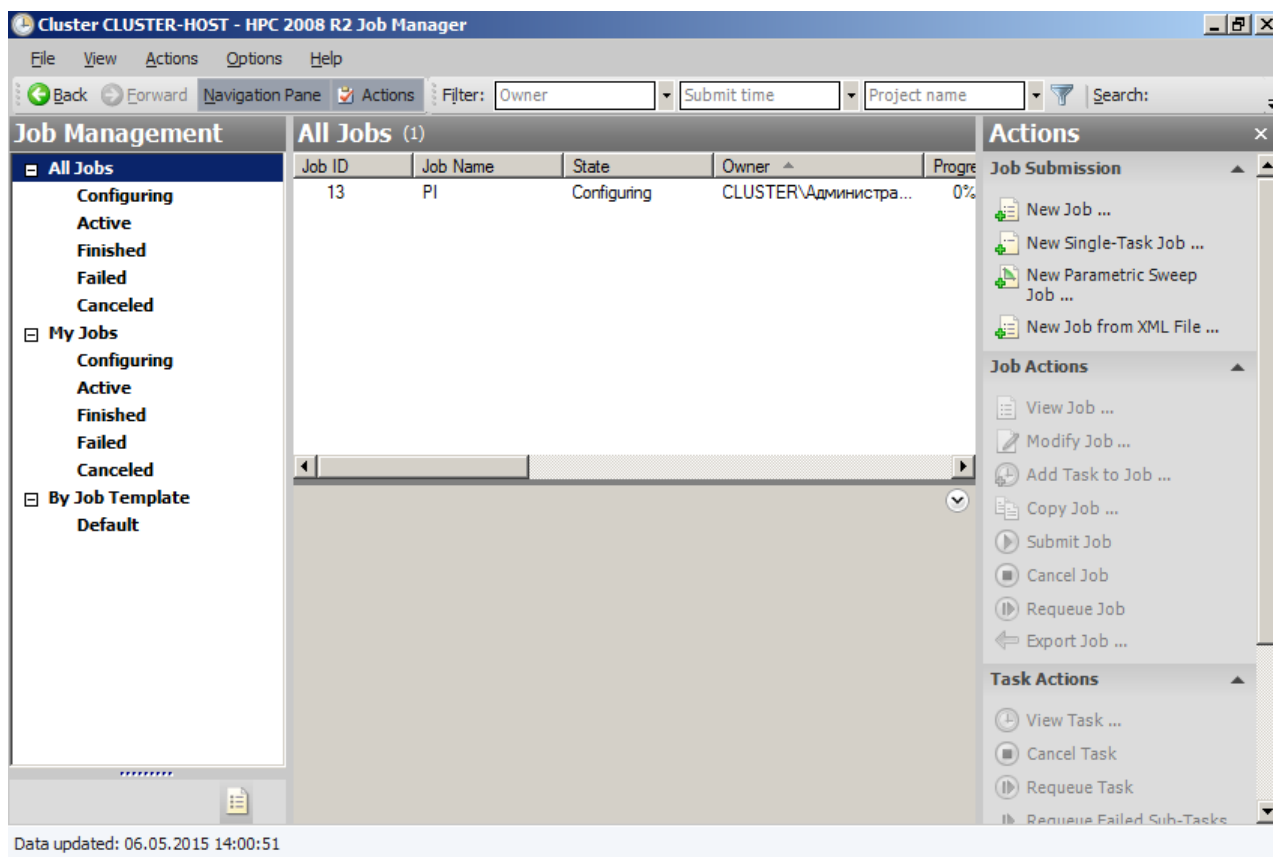


Рисунок 3.2 - Окно Job Manager

Для создания новой задачи в меню Job Submission справа нажимаем New Job. В поле Job name вводим имя задания, выбираем шаблон. Также можно ограничить время выполнения программы. Пример создания нового задания приведен на рисунке 3.3.

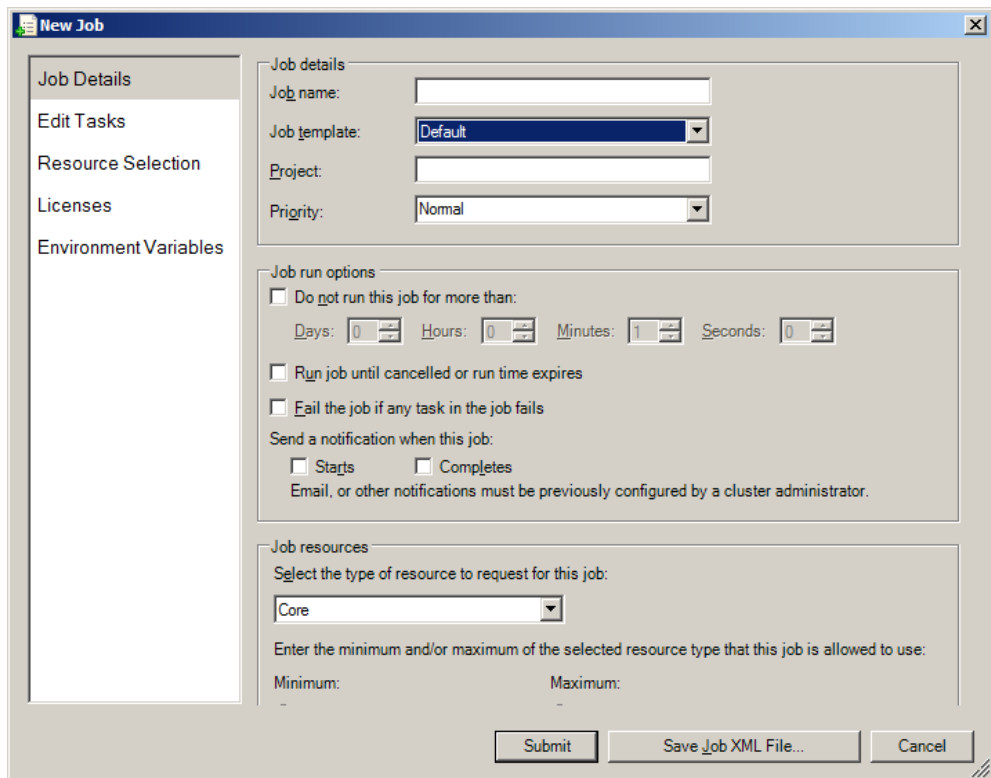


Рисунок 3.3 - Создание нового задания

Далее слева открываем вкладку « Edit Task» и нажимаем кнопку Add. Вкладка Edit Task представлена на рисунке 3.4.

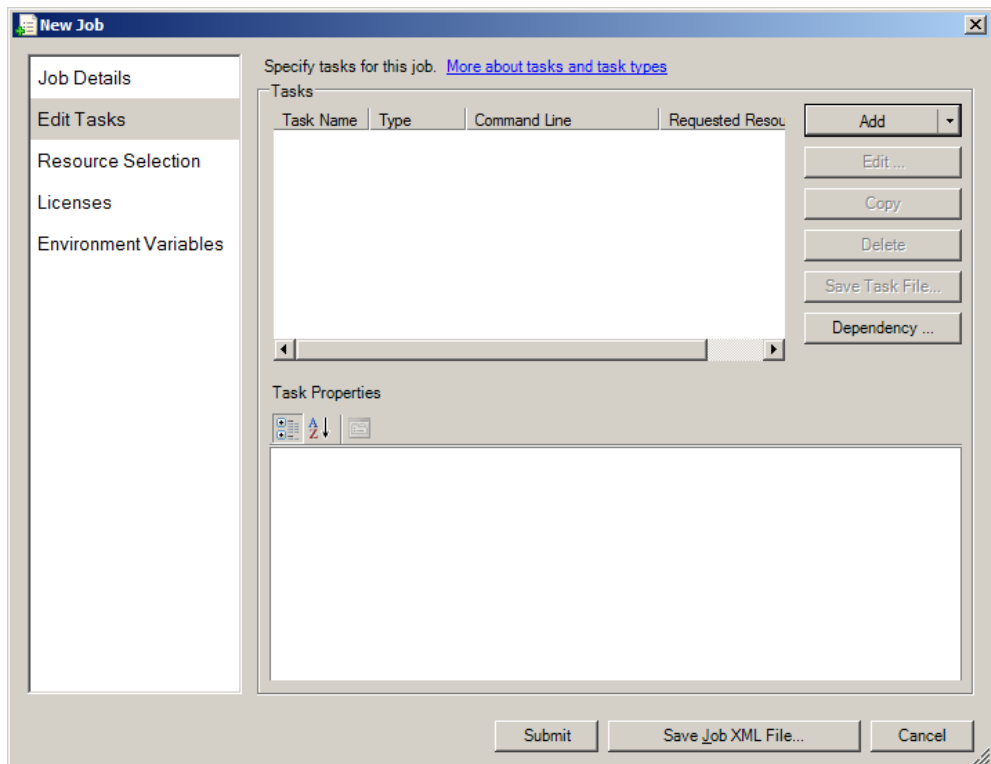


Рисунок 3.4 - Вкладка Edit Task

В открывшемся окне указываем имя задачи, строчку командной строки, которая отвечает за запуск программы, указываем директорию, где лежат исполняемые файлы (сетевая папка на главном узле), указываем файл, в котором содержится информация для ввода, указываем файлы, в которые будут выводиться результаты работы программы и отчёты об ошибках. Заполнение полей представлено на рисунке 3.5.

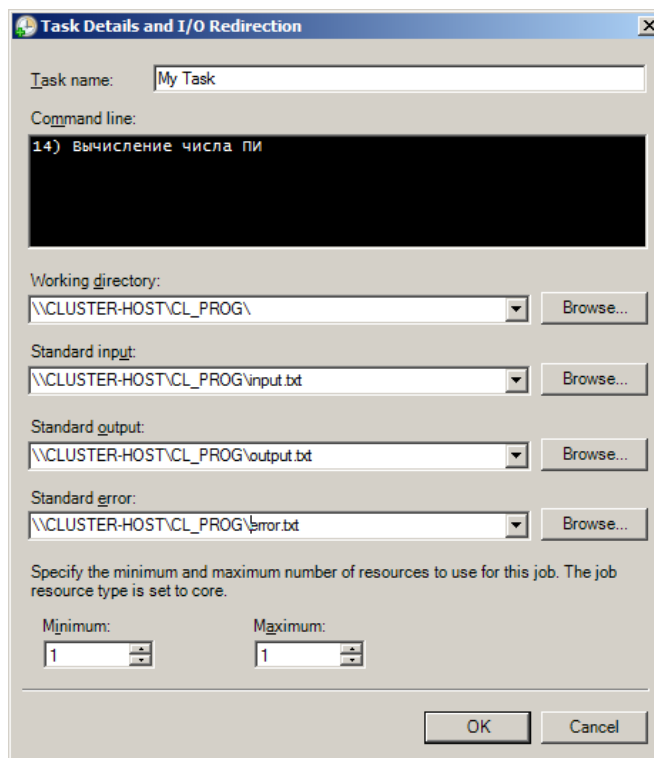


Рисунок 3.5 - Настройка параметров ввода и вывода

После заполнения полей нажимаем Submit. Теперь программа находится на выполнении, за состоянием можно следить в таблице Active или All Jobs (в меню слева). Если программа выполнена успешно, она появится в таблице Finished, иначе она появится в таблице Failed. Результат выполнения программы будет выведен в файл, который был указан при постановке задачи на выполнение. Результат работы программы представлен на рисунке 3.6.

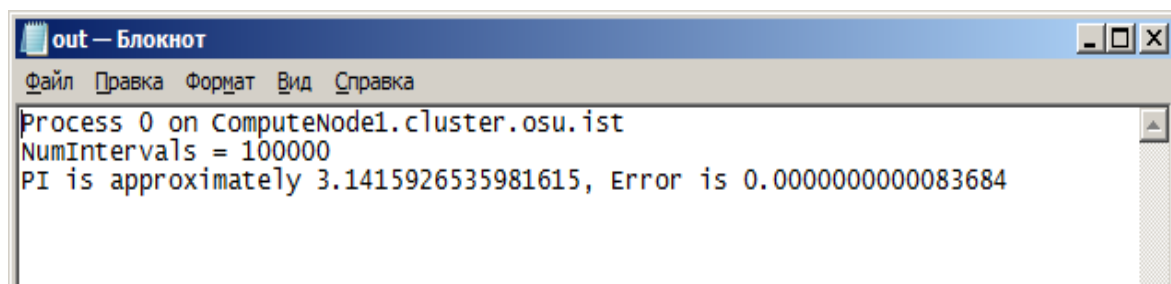


Рисунок 3.6- Результат работы программы

3.2 Запуск параллельной задачи

Для сборки программы под кластер необходимо установить библиотеки MPI. Установка осуществляется из файла `mpi_x64.msi`, который находится в папке `setup`. Во время установки указываем директорию, куда будут скопированы файлы библиотек и включаемых файлов. Для компиляции программ желательно использовать Microsoft Visual Studio, но компилировать можно и на любом компиляторе C и C++.

1. Запускаем Visual Studio. В меню Файл раскрываем подменю.

2. Создаем и выбираем Проект. В появившемся окне, представленном на рисунке 3.7, выбираем консольное приложение, вводим имя проекта, имя решения и путь к папке, где будет храниться решение.

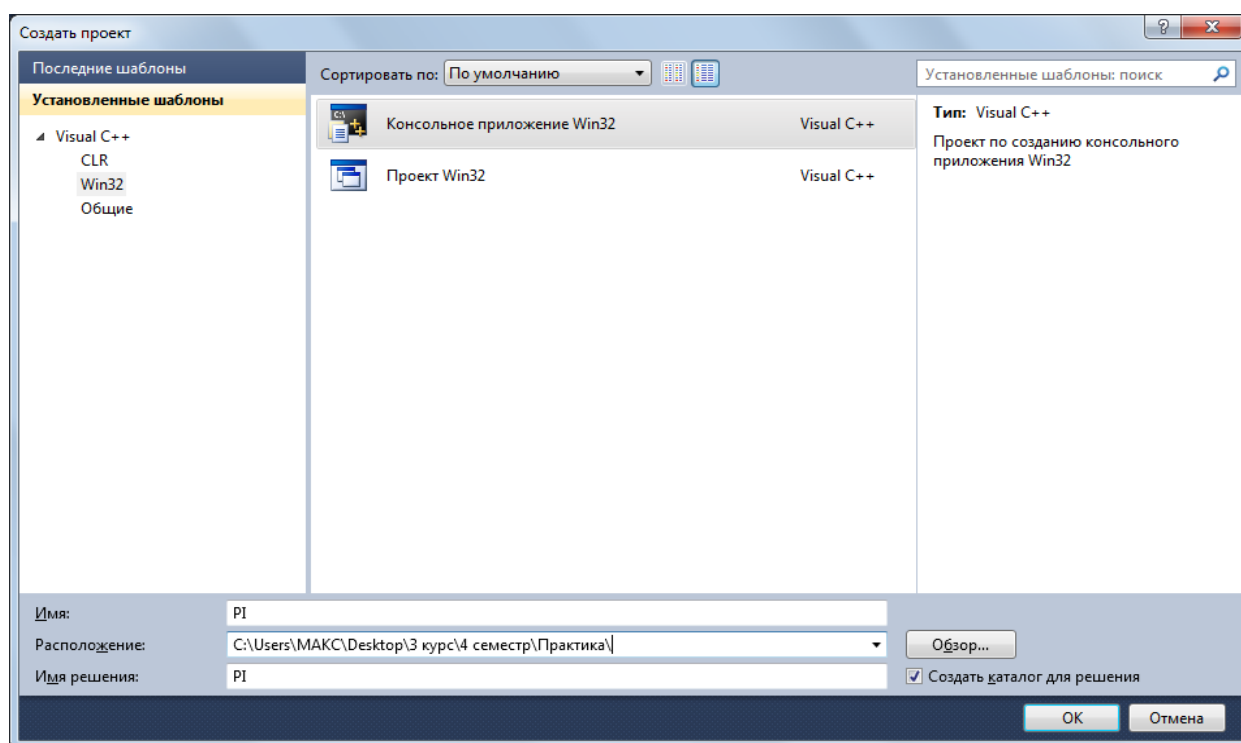


Рисунок 3.7 - Создание проекта

3. Далее в появившемся окне, представленном на рисунке 3.8, в параметрах приложения убираем галочку напротив «Предварительно скомпилированный заголовок» и ставим галочку напротив «Пустой проект».

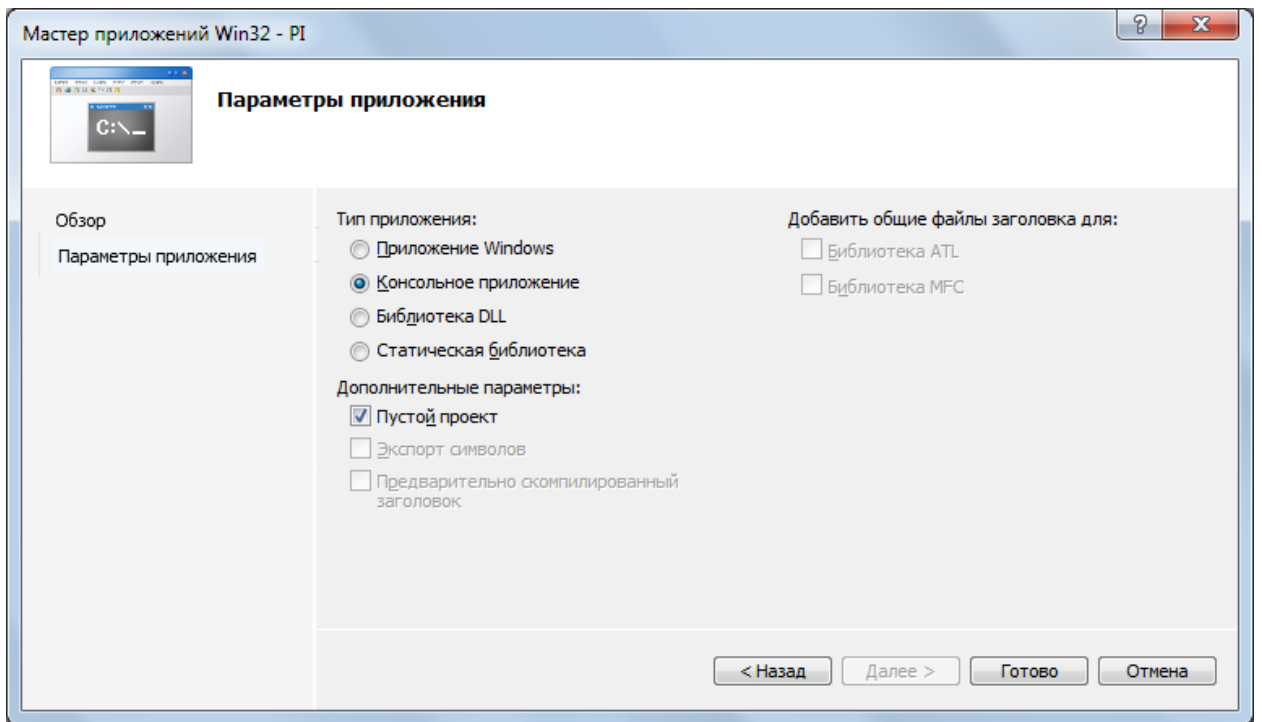


Рисунок 3.8 - Параметры создаваемого проекта

4. Теперь в обозревателе решений выбираем наш проект и нажимаем правой кнопкой мыши на «Файлы исходного кода» и в раскрывшемся меню наводим на «Добавить» и нажимаем «Создать элемент». Выбираем тип файла и указываем имя, нажимаем «Добавить», согласно рисунку 3.9.

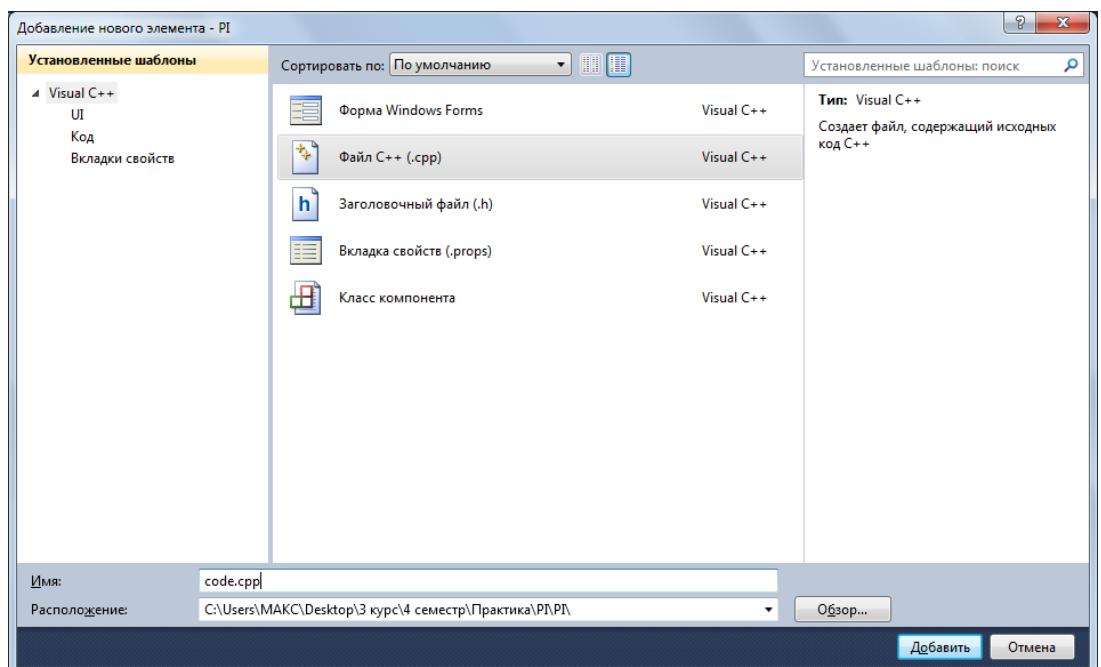


Рисунок 3.9 - Выбор типа создаваемого элемента

5. Создаем исходный файл.

В качестве примера можно скомпилировать код программы по вычислению числа пи:

```
#include <stdlib.h>
#include <stdio.h>
#include <math.h>
#include <mpi.h>

void main(int argc, char *argv[])
{
    int    NumIntervals    = 0; // num intervals in the domain [0,1]
    double IntervalWidth  = 0.0; // width of intervals
    double IntervalLength = 0.0; // length of intervals
    double IntrvlMidPoint = 0.0; // x mid point of interval
    int    Interval       = 0; // loop counter
    int    done           = 0; // flag
    double MyPI           = 0.0; // storage for PI approximation results
    double ReferencePI = 3.141592653589793238462643; // value for comparison
    double PI;
    char    processor_name[MPI_MAX_PROCESSOR_NAME];
    char    (*all_proc_names)[MPI_MAX_PROCESSOR_NAME];
    int     numprocs;
    int     MyID;
    int     namelen;
    int     proc = 0;

    MPI_Init(&argc,&argv);
    MPI_Comm_size(MPI_COMM_WORLD,&numprocs);
    MPI_Comm_rank(MPI_COMM_WORLD,&MyID);
    MPI_Get_processor_name(processor_name,&namelen);

    all_proc_names = (char(*)[128]) malloc(numprocs *
MPI_MAX_PROCESSOR_NAME);

    MPI_Gather(processor_name, MPI_MAX_PROCESSOR_NAME,
MPI_CHAR,
    all_proc_names, MPI_MAX_PROCESSOR_NAME, MPI_CHAR, 0,
MPI_COMM_WORLD);
    if (MyID == 0)
    {
```

```

    for (proc=0; proc<numprocs; ++proc)
        printf("Process %d on %s\n", proc, all_proc_names[proc]);
    }

IntervalLength = 0.0;
if (MyID == 0)
    {
    if (argc > 1)
        {
        NumIntervals = atoi(argv[1]);
        }
    else
        {
        NumIntervals = 100000;
        }
    printf("NumIntervals = %i\n", NumIntervals);
    }

// send number of intervals to all procs
MPI_Bcast(&NumIntervals, 1, MPI_INT, 0, MPI_COMM_WORLD);

if (NumIntervals != 0)
    {
    //approximate the value of PI
    IntervalWidth = 1.0 / (double) NumIntervals;

    for (Interval = MyID+1; Interval <= NumIntervals; Interval += numprocs)
        {
        IntrvlMidPoint = IntervalWidth * ((double)Interval - 0.5);
        IntervalLength += (4.0 / (1.0 + IntrvlMidPoint*IntrvlMidPoint));
        }
    MyPI = IntervalWidth * IntervalLength;

    // Calculating the sum of all local alues of MyPI
    MPI_Reduce(&MyPI, &PI, 1, MPI_DOUBLE, MPI_SUM, 0,
MPI_COMM_WORLD);

    //reportapproximation
    if (MyID == 0)

```

```

    {
        printf("PI is approximately %.16f, Error is %.16f\n",
            PI, fabs(PI - ReferencePI));
    }
}

```

```

MPI_Finalize();
}

```

6. После создания файла исходного кода можно сразу же набирать его, или же для начала настроить проект. Стоит отметить, что настроить проект можно в любое время. Нажимаем правой кнопкой мыши на имени проекта в обозревателе решений и нажимаем «Свойства». Раскрываем вкладку Свойства конфигурации и выбираем «Каталоги C/C++». В полях «Каталоги включения» и «Каталоги библиотек» указываем пути к файлам включения и библиотекам. Пример настройки каталогов приведен на рисунке 3.10.

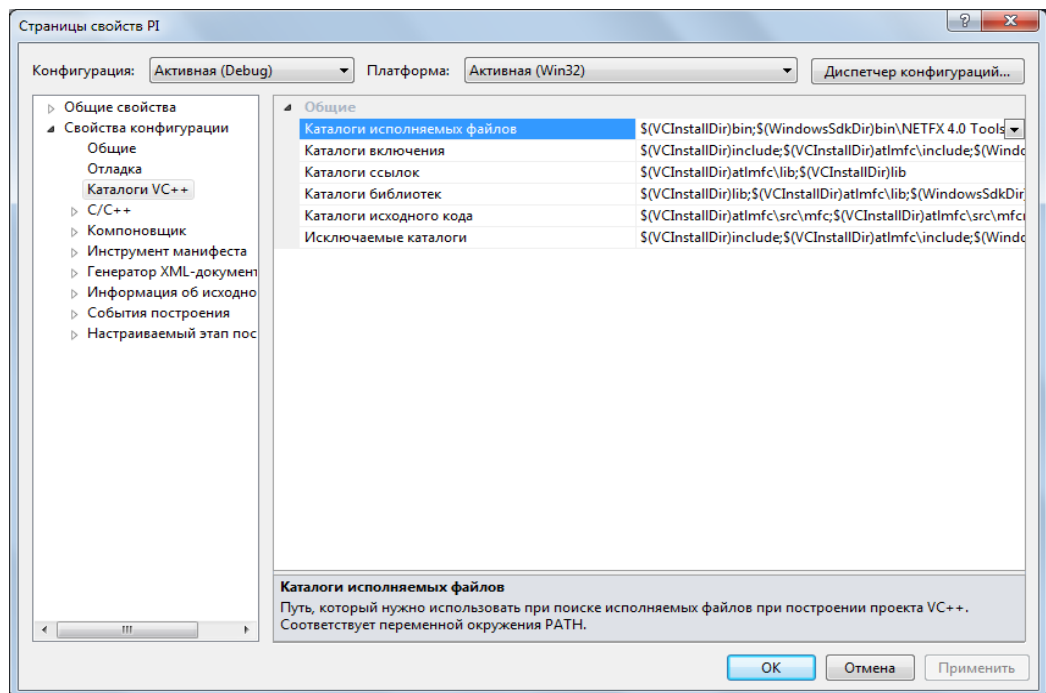


Рисунок 3.10 - Настройка каталогов

7. Выполните команду компиляции и линковки проекта:

8. Поздравляем! Компиляция параллельной программы для MS MPI успешно завершена.

Список использованных источников

1 Цилькер, Б. Я. Организация ЭВМ и систем: учебник для вузов / Б. Я. Цилькер, С. А. Орлов . - СПб. : Питер, 2006. - 668 с. - (Учебник для вузов). - Библиогр.: с. 638-652. - Алф. указ.: с. 653-667 .

2 Гергель, В. П. Основы параллельных вычислений для многопроцессорных вычислительных машин: учеб.пособие / В. П. Гергель, Р. Г. Стронгин . - 2-е изд., доп. - Нижний Новгород : Изд-во Нижегородского ун-та, 2003. - 184 с. - Библиогр.: с. 179.

3 Богданов, А.В. Архитектуры и топологии многопроцессорных вычислительных систем: учеб.пособие для вузов / А. В. Богданов [и др.]. - М.: Интернет-Ун-т Информ. Технологий, 2009. - 175 с. - (Основы информационных технологий). - Библиогр.: с. 167-170.

4 Немнюгин, С.А. Параллельное программирование для многопроцессорных вычислительных систем / С. А. Немнюгин, О. Л. Стесик . - СПб. : БХВ-Петербург, 2002. - 400 с. : ил.. - Библиогр.: с. 385-396.

5 Lupin, S. A. Технологии параллельного программирования: учеб. пособие / С. А. Lupin, М. А. Посыпкин . - М. : Форум, 2008. - 206 с. : ил.. - (Высшее образование). - Библиогр.: с. 147. - Прил.: с. 148-205.

6 Воеводин, В. В. Параллельные вычисления: учеб. пособие для вузов / В. В. Воеводин, Вл. В. Воеводин . - СПб. : БВХ-Петербург, 2004. - 600 с. : ил.. - Библиогр.: с. 588-591. - Библиогр.: с. 593-599.

7 Хорошевский, В. Г. Архитектура вычислительных систем: учеб. пособие для вузов по направлению "Информатика и вычисл. техника" / В. Г. Хорошевский . - 2-е изд., перераб. и доп. - М. : МГТУ им. Н.Э. Баумана, 2008. - 520 с. : ил.. - (Информатика в техническом университете). - Прил.: с. 511-518. - Библиогр.: с. 519.

8 Гергель, В.П. Технологии построения и использования кластерных систем.- Режим доступа: http://www.intuit.ru/goods_store/ebooks/8456.