

Лабораторная работа № 2

Технологии распределенной обработки данных

Тема: Разработка распределённой системы обработки данных

Инв. № подл.	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата

Проверил:

Гай В. Е.

Выполнил:

Студент гр. 14-В-2

Носов А. В.

## 1. Цель и порядок выполнения работы

Разработка распределённой системы обработки данных.

Разработанный программный комплекс должен состоять из Сервера и Клиента.

Функции сервера: хранение удалённого объекта, предоставляющего доступ к заданиям для обработки и результату обработки.

Предусмотреть на сервере возможность одновременного доступа к критической секции кода нескольких клиентов (с помощью оператора lock). Критическая секция кода - та, к которой гипотетически одновременно могут обратиться несколько клиентов.

Функции клиента (на сервере хранится список клиентов - эта функция уже предусмотрена исходным кодом библиотеки RemoteBase):

1) управляющие функции (выполняет только один клиент из всего множества клиентов, выполнение данной функции должно выполняться через вызов методов удалённого объекта (удалённый объект хранится на сервере)):

1.1) формирование и ведение списка заданий (под ведением понимается удаление уже обработанных и предоставление клиенту задания по запросу);

1.2) получение, объединение и вывод результатов вычислений (результаты вычислений должны выводиться в каждом клиенте, для этого необходимо проверять окончание обработки всех данных по таймеру; объединение

результатов вычисление также можно реализовать с использованием таймера);

1.3) устанавливает флаг того, что управляющий клиент назначен, на сервере сохраняется идентификатор клиента;

2) вычислительные функции

2.1) запрос задания с сервера (клиент должен запросить задание только после того, как эти задания были сформированы);

2.2) обработка данных;

2.3) отправка результатов обработки на сервер.

Требования к системе:

1) предусмотреть возможность отключения одного из клиентов, получившего задание на обработку.

2) предусмотреть возможность отключения управляющего клиента (для этого можно хранить время последней операции на сервере).

## Лабораторная работа № 2

Разработка распределённой системы обработки данных

14-B-2

Содержание

- 1. Листинг RemoteClient
- 2. Листинг RemoteBase
- 3. Листинг RemoteServer

Инв. № подл	Подп. и дата				Инв. № дубл.	Взам. инв. №				Подп. и дата				
Ли					Изм.					№ докум.	Подп.	Дат		Лист

## 2. Выполнение лабораторной работы

### 2.1. Листинг RemoteClient

```
using System;
using System.Windows.Forms;

using Library;

// Connect namespace for .NET Remoting
using System.Runtime.Remoting.Channels.Tcp;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting.Services;
using System.Runtime.Remoting;

namespace Client
{
    class Program
    {
        private static Lib remoteObject = null;

        public static int SuperMethod(int par1, int par2)
        {
            return par1 * par2;
        }

        static void Main(string[] args)
        {
            // Opening channel
            TcpChannel channel = new TcpChannel();

            // Registering channel
            ChannelServices.RegisterChannel(channel, true);

            // Getting reference for instance of library
            remoteObject = (Lib)Activator.GetObject(typeof(Lib), "tcp://localhost:11114/equ");

            Console.WriteLine("Entered main routine...");

            int id = remoteObject.Connect();

            while (!remoteObject.IsFinished())
            {
                int[] parameters = remoteObject.GetParameters();
                Console.WriteLine("Client multiplying values: {0}, {1}", parameters[0], parameters[1]);
                remoteObject.ReturnResultFromClient(SuperMethod(parameters[0], parameters[1]));

                // Console.ReadKey();
            }

            remoteObject.Disconnect(id);

            Console.WriteLine("Client exited.");
        }
    }
}
```

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						Лист 3
Ли	Изм.	№ докум.	Подп.	Дат	Лабораторная работа № 2					

```
Console.ReadKey();
```

```
}  
}  
}
```

## 2.2. Листинг RemoteBase

```
using System;  
using System.Collections.Generic;  
  
namespace Library  
{  
    // SharedLibrary class  
    public class Lib : MarshalByRefObject  
    {  
  
        public const int ARRAY_SIZE = 20000;  
        public int needResult = 0;  
  
        public Lib()  
        {  
            Random r = new Random();  
  
            for (int i = 0; i < ARRAY_SIZE; ++i)  
            {  
                vectorA[i] = r.Next(100);  
                vectorB[i] = r.Next(100);  
                needResult += vectorA[i] * vectorB[i];  
            }  
  
            /*  
            for (int i = 0; i < ARRAY_SIZE; ++i)  
                Console.Write("{0} ", vectorA[i]);  
  
            Console.WriteLine();  
  
            for (int i = 0; i < ARRAY_SIZE; ++i)  
                Console.Write("{0} ", vectorB[i]);  
            * */  
  
            Console.WriteLine();  
  
            Console.WriteLine("Needed result: {0}", needResult);  
        }  
  
        int[] clientParameter = new int[2];  
  
        int[] vectorA = new int[ARRAY_SIZE];  
        int[] vectorB = new int[ARRAY_SIZE];  
        int[] dataProcessed = new int[ARRAY_SIZE];  
  
        int result = 0;  
  
        List<int> clients = new List<int>();  
        int ID = 0;  
  
        public int Connect()  
        {  
            lock ("connect")  
            {  
                ID++;  
                clients.Add(ID);  
            }  
        }  
    }  
}
```

Подп. и дата

Взам. инв. №

Инв. № дубл.

Подп. и дата

Инв. № подл

Ли Изм. № докум. Подп. Дат

Лабораторная работа № 2

Лист

4

```

return ID;
}
}

public void Disconnect(int id)
{
lock ("disconnect")
{
clients.Remove(id);

if (isFinished() && (clients.Count == 0))
{
Console.WriteLine("Result: " + this.result);
Console.WriteLine("Pre-calculated server result (for testing): " + this.needResult);
}
}
}

public int[] GetParameters()
{
lock ("get_task")
{
int i = 0;

for (i = 0; i < ARRAY_SIZE; ++i)
if (dataProcessed[i] == 0)
break;

dataProcessed[i] = 2;
clientParameter[0] = vectorA[i];
clientParameter[1] = vectorB[i];

Console.WriteLine("Processed: {0}", i / (ARRAY_SIZE / 100.0));

```

```

return clientParameter;
}
}

public bool isFinished()
{
lock ("finished_query")
{
for (int i = 0; i < ARRAY_SIZE; ++i)
if (dataProcessed[i] == 0)
return false;

return true;
}
}

public void ReturnResultFromClient(int _result)
{
lock ("collecting_result")
{
this.result += _result;
}
}
}
}

```

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата	<div>Лабораторная работа № 2</div>	Лист
						5
Ли	Изм.	№ докум.	Подп.	Дат		

2.3. Листинг RemoteServer

```
using System;
using System.Runtime.Remoting.Channels.Tcp;
using System.Runtime.Remoting.Channels;
using System.Runtime.Remoting;

namespace Server
{
    class Program
    {
        static void Main(string[] args)
        {
            Console.WriteLine("Open port {0}...", 11114);

            TcpChannel channel = new TcpChannel(11114);

            Console.WriteLine("Register channel...");
            ChannelServices.RegisterChannel(channel, true);

            Console.WriteLine("Load cross-server lib...");
            RemotingConfiguration.RegisterWellKnownServiceType(typeof(Library.Lib), "equ", WellKnownObjectMode.Singleton);

            Console.WriteLine("\nPress <ENTER> to shutdown server.");
            Console.ReadLine();

            Console.WriteLine("Unregister channel...");
            ChannelServices.UnregisterChannel(channel);

            return;
        }
    }
}
```

2.4. Тестирование работоспособности системы

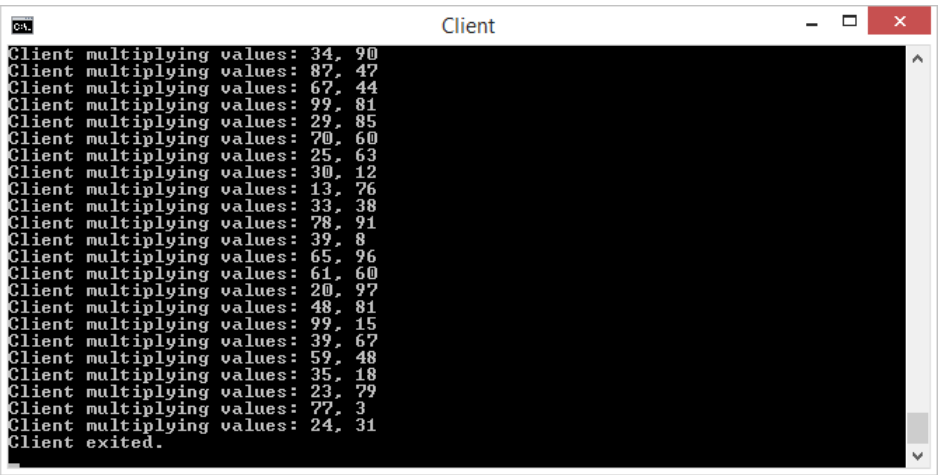


Рисунок 2.1. Пример работы клиента

Подп. и дата	
Взам. инв. №	
Инв. № дубл.	
Подп. и дата	
Инв. № подл	

Ли	Изм.	№ докум.	Подп.	Дат

Server

```
Processed: 99,89
Processed: 99,895
Processed: 99,9
Processed: 99,905
Processed: 99,91
Processed: 99,915
Processed: 99,92
Processed: 99,925
Processed: 99,93
Processed: 99,935
Processed: 99,94
Processed: 99,945
Processed: 99,95
Processed: 99,955
Processed: 99,96
Processed: 99,965
Processed: 99,97
Processed: 99,975
Processed: 99,98
Processed: 99,985
Processed: 99,99
Processed: 99,995
Result: 48818386
Pre-calculated server result <for testing>: 48818386
```

Рисунок 2.2. Пример работы сервера

2.5. Вывод

В результате выполнения данной лабораторной работы был изучен компонент созданный фирмой Microsoft для межпроцессно-го взаимодействия .NET Remoting. На основе этого компонента была реализована простейшая сиситема распределенной обработки данных реализующая нахождение скалярного N-мерного произведения векторов.

Инв. № подл	Подп. и дата	Инв. № дубл.	Взам. инв. №	Подп. и дата						
Ли	Изм.	№ докум.	Подп.	Дат	Лабораторная работа № 2					Лист
										7