

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ

СОГЛАСОВАНО

Генеральный директор  
ЗАО «АйТи»

  
Бакнев О.Р.  
"29" января 2011 г.

УТВЕРЖДАЮ

Ректор НИУ ИТМО

  
Васильев В.Н.  
"29" января 2011 г.

МНОГОПРОФИЛЬНАЯ ИНСТРУМЕНТАЛЬНО-  
ТЕХНОЛОГИЧЕСКАЯ ПЛАТФОРМА СОЗДАНИЯ  
И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННОЙ СРЕДОЙ  
ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ CLAVIRE

ПРОГРАММНЫЙ КОМПОНЕНТ - БАЗА РЕСУРСОВ  
CLAVIRE/RESOURCEBASE

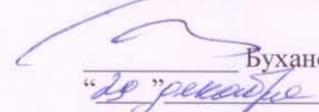
ОПИСАНИЕ ПРОГРАММЫ

ЛИСТ УТВЕРЖДЕНИЯ

RU.СНАБ.80066-06 13 33-ЛУ

Представители  
Организации-разработчика

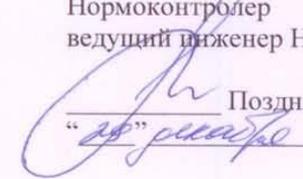
Руководитель разработки,  
профессор НИУ ИТМО

  
Бухановский А.В.  
"29" января 2011 г.

Ответственный исполнитель,  
с.п.с. НИУ ИТМО

  
Луценко А.Е.  
"29" января 2011 г.

Нормоконтролер  
ведущий инженер НИУ ИТМО

  
Позднякова Л.Г.  
"29" января 2011 г.

2011

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

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

---

**УТВЕРЖДЕН**

**RU.СНАБ.80066-06 13 Ошибка! Источник ссылки не найден.-ЛУ**

**МНОГОПРОФИЛЬНАЯ ИНСТРУМЕНТАЛЬНО-  
ТЕХНОЛОГИЧЕСКАЯ ПЛАТФОРМА СОЗДАНИЯ  
И УПРАВЛЕНИЯ РАСПРЕДЕЛЕННОЙ СРЕДОЙ  
ОБЛАЧНЫХ ВЫЧИСЛЕНИЙ CLAVIRE**

**ПРОГРАММНЫЙ КОМПОНЕНТ - БАЗА РЕСУРСОВ  
CLAVIRE/RESOURCEBASE**

**ОПИСАНИЕ ПРОГРАММЫ**

**RU.СНАБ.80066-06 13 ОШИБКА! ИСТОЧНИК ССЫЛКИ НЕ НАЙДЕН.**

**ЛИСТОВ 16**

2011

<b>Ине.№ подл.</b>	
<b>Подп. и дата</b>	
<b>Взам. ине. №</b>	
<b>Ине. № дубл.</b>	
<b>Подп. и дата</b>	

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

## **АННОТАЦИЯ**

Документ содержит описание программного компонента – базы ресурсов CLAVIRE/ResourceBase RU.СНАБ.80066-06 01 **Ошибка! Источник ссылки не найден.**, реализующего необходимый для работы других компонентов CLAVIRE функционал по работе с конфигурацией вычислительных ресурсов системы. Программный компонент база ресурсов разработан в ходе выполнения проекта «Создание распределенной вычислительной среды на базе облачной архитектуры для построения и эксплуатации высокопроизводительных композитных приложений» (Договор № 21057 от 15 июля 2010 г., шифр 2010-218-01-209) в рамках реализации постановления Правительства РФ № 218 «О мерах государственной поддержки развития кооперации российских высших учебных заведений и организаций, реализующих комплексные проекты по созданию высокотехнологичного производства».

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник  
ссылки не найден.**

## СОДЕРЖАНИЕ

1.	ОБЩИЕ СВЕДЕНИЯ.....	4
2.	ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ.....	4
3.	ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ .....	4
3.1.	Принципы функционирования базы ресурсов .....	4
3.2.	Основные классы базы ресурсов .....	8
3.2.1.	Класс ResourceBaseService .....	8
3.2.2.	Класс Resource .....	9
3.2.3.	Класс ResourceNode .....	10
3.2.4.	Класс NodeState .....	11
3.2.5.	Класс PackageOnNode .....	11
3.2.6.	Класс NodeDataFolders.....	12
3.2.7.	Класс NodeServices.....	12
4.	ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА.....	13
5.	ВЫЗОВ И ЗАГРУЗКА .....	13
6.	ВХОДНЫЕ ДАННЫЕ .....	14
7.	ВЫХОДНЫЕ ДАННЫЕ.....	15
	ПЕРЕЧЕНЬ СОКРАЩЕНИЙ .....	16

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

## 1. ОБЩИЕ СВЕДЕНИЯ

Программный компонент – база ресурсов CLAVIRE/ResourceBase RU.СНАБ.80066-06 01 **Ошибка! Источник ссылки не найден.** – элемент МИТП, предоставляющий информацию о доступных вычислительных ресурсах, их структуре, установленных на них вычислительных пакетах, их настройках и способах взаимодействия. Данный компонент разработан на языке С# в виде web-сервиса с применением технологии Windows Communication Foundation (WCF) на основе стандарта SOAP. Развертывание сервиса производится на базе web-сервера Internet Information Services (IIS) версии не ниже 6.0.

## 2. ФУНКЦИОНАЛЬНОЕ НАЗНАЧЕНИЕ

Основные функции, выполняемые базой ресурсов:

- 1) сбор и проверка корректности информации о составе и конфигурации вычислительных ресурсов;
- 2) предоставление унифицированного доступа к сведениям об имеющихся в составе платформы ресурсах и установленных на них пакетах;
- 3) обеспечение возможности изменения состава и настроек используемых платформой вычислительных ресурсов.

## 3. ОПИСАНИЕ ЛОГИЧЕСКОЙ СТРУКТУРЫ

### 3.1. Принципы функционирования базы ресурсов

База ресурсов является web-сервисом, предоставляющим унифицированный доступ к распределенным сведениям о составе и структуре вычислительных ресурсов.

Под ресурсом понимается логически целостная совокупность вычислителей (возможно, разнородных), называемых вычислительными узлами. Каждый узел может иметь собственную архитектуру и различный набор вычислителей (как правило, ядер центрального процессора), однако в рамках одного ресурса все узлы имеют общие механизмы управления и доступа. Отдельными вычислительными ресурсами могут являться кластеры, набор персональных компьютеров или целый сегмент Грид-сети.

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

Важно понимать, что способ интерпретации большинства параметров, задаваемых в базе ресурсов, зависит от типа ресурса и возлагается на модули МИТП, взаимодействующие с системой управления конкретным ресурсом. Соответственно идентичность трактовки отдельных параметров и даже само их наличие для различных ресурсов в общем случае не гарантируются, а в некоторых случаях и невозможна ввиду различий в технологии функционирования разных типов вычислительных ресурсов. Тем не менее база ресурсов спроектирована таким образом, чтобы обязательные и необязательные регулярные параметры ресурсов могли трактоваться единообразно вне зависимости от типа ресурса, а все переменные ресурсозависимые параметры могли быть описаны в специальных расширяемых нетипизированных словарях параметров вычислительного ресурса и его узлов.

Общая схема работы базы ресурсов и ее взаимодействия с прочими компонентами системы показана на рис. 3.1.

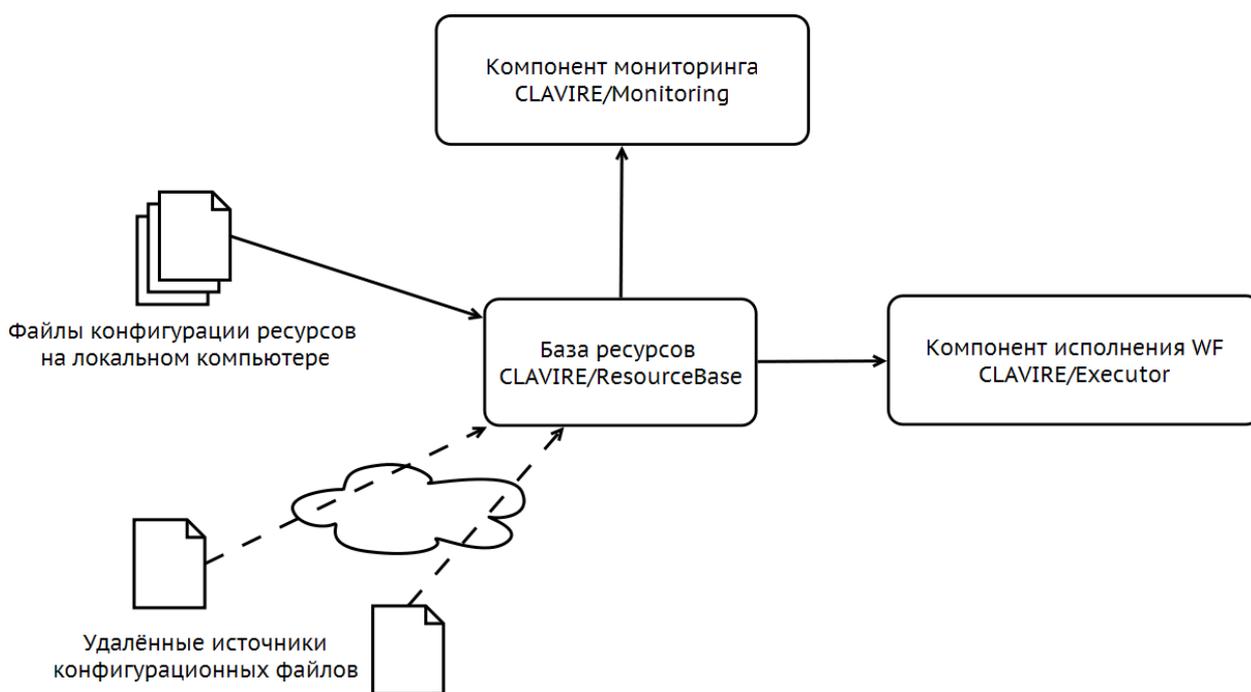


Рисунок 3.1 – Схема работы базы ресурсов

Главным потребителем информации о составе и структуре ресурсов является компонент исполнения WF (CLAVIRE/Executor), который использует полученную информацию для взаимодействия различными компонентами комплекса, включая компоненты планирования, учета использования ресурсов и доступа к вычислительным ресурсам.

**RU.СНАБ.80066-06 13 Ошибка! Источник ссылки не найден.Ошибка! Источник  
ссылки не найден.**

Загрузка файлов конфигурации может осуществляться как с компьютера, на котором используется база ресурсов, так и с удаленных источников. В последнем случае возможно хранить описание ресурса на самом ресурсе или ином компьютере, к которому есть доступ локального администратора вычислительного ресурса. Это позволяет поддерживать информацию о ресурсе в актуальном состоянии, изменяя ее при установке или удалении вычислительных пакетов, отключении вычислительных узлов в связи с проводимыми профилактическими работами, при изменении параметров доступа к узлам (например, при смене адресов вычислителей или сервисов запуска и мониторинга); при этом подобные изменения не требуют доступа к основным сервисам системы и могут быть осуществлены администратором ресурса самостоятельно.

Само описание вычислительного ресурса и входящих в его состав узлов задается в нотации JSON (JavaScript Object Notation). Пример содержимого файла описания ресурса показан на рис. 3.2.

```
cluster_niinkt_1 =
{
  "ResourceName": "cluster_niinkt_1",
  "ResourceDescription": "Кластер НИИ НКТ (HP 2x8)",
  "SupportedArchitectures": ["MPP", "SMP"],
  "ProviderName": "Cluster",

  "Nodes":
  [
    { "NodeName": "i-master", "NodeAddress": "192.168.1.51" },
    { "NodeName": "i-node", "NodeAddress": "192.168.1.52" },
  ],

  "NodeDefaults":
  {
    "SupportedArchitectures": ["SMP"],

    "Services":
    { "ExecutionUrl": "http://192.168.1.189/Integrator/IntegratorService.asmx" },

    "DataFolders":
    {
      "ExchangeUrlFromSystem": "ftp://192.168.1.189/{task}/{phase}",
      "ExchangeUrlFromResource": "ftp://192.168.1.189/{task}/{phase}",
      "LocalFolder": "/mnt/share_from_head/{task}"
    },

    "CoresCount": 8,

    "Packages":
    [
      { "Name": "TestP", "Version": "v1", "AppPath": "ntestp.sh" },
      { "Name": "SEMP", "Version": "v1", "AppPath": "zindo1.sh" },
      { "Name": "ORCA", "Version": "v1", "AppPath": "orca" },
      { "Name": "GAMESS", "Version": "v1", "AppPath": "gms" },
      { "Name": "JAggregate", "Version": "v1", "AppPath": "runMPI.sh" }
    ]
  }
}
```

Рисунок 3.2 – Пример файла описания ресурса

**RU.СНАБ.80066-06 13 Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

В конфигурацию вычислительного ресурса входят следующие параметры:

- 1) уникальное имя ресурса;
- 2) строка комментария, показываемая пользователю системы;
- 3) поддерживаемые ресурсом вычислительные архитектуры;
- 4) тип ресурса – имя провайдера для компонента исполнения WF;
- 5) координаты, указывающие географическое положение ресурса;
- 6) описание входящих в состав ресурса вычислительных узлов.

Главной составляющей описания вычислительных узлов является список установленных пакетов и их параметров, зависящих от конкретной машины, на которой они установлены. К таким параметрам относятся, например, версия программы (необходима для установления корректных параметров базой пакетов), путь к запускаемому приложению, адреса сервисов, осуществляющих пакетный запуск и мониторинг состояния как запущенных задач, так и всего узла в целом.

Такие параметры узла, как поддерживаемые вычислительные архитектуры (например, SMP или GPGPU) и количество вычислительных ядер, используются в процессе планирования для сравнения вычислительных ресурсов и оптимизации параметров запуска задачи.

Некоторые параметры узлов ответственны за описание взаимодействия между ресурсом и остальными частями платформы. Так, помимо указания параметров аутентификации возможно указание пути к директории на ресурсе, в которой будут собираться результаты вычислений, а также URL-адреса папки обмена данными, если ресурс не поддерживает непосредственной записи и загрузки файлов расчета, а производит их копирование из промежуточного источника самостоятельно или посредством провайдеров.

Отдельные вычислительные пакеты слабо приспособлены к одновременному запуску из различных рабочих каталогов, например, вследствие записи в файлы, находящиеся в директории приложения. Для этого в базе ресурсов для отдельных пакетов можно указать директиву CopyOnStartup копирования определенных файлов и каталогов перед запуском программы. Для предотвращения копирования ненужных файлов больших размеров в хранилище (являющихся временными или полученными вследствие копирования файлов приложения в рабочий каталог) поддерживается параметр Cleanup,

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

содержащий список файлов и директорий, которые необходимо удалить до загрузки выходных данных в хранилище.

Часть параметров в описании узла (таких как значения переменных окружения, выставление которых необходимо для запуска конкретного пакета) может быть опущена, другая обязательна для заполнения (имена узлов и установленных на них пакетов). При загрузке описания очередного ресурса проверяется корректность заполнения всех параметров. Отрицательный результат проверки заносится в журнал для возможности последующей корректировки.

Часто большинство устанавливаемых в базе ресурсов параметров совпадает для всех или многих узлов. Поэтому в базе ресурсов поддерживается установка настроек по умолчанию для всех узлов внутри вычислительного ресурса. Для каждого конкретного узла данные параметры могут быть перезаписаны или дополнены.

### **3.2. Основные классы базы ресурсов**

Структура основных классов, используемых в компоненте-базе ресурсов, показана на рис. 3.3.

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

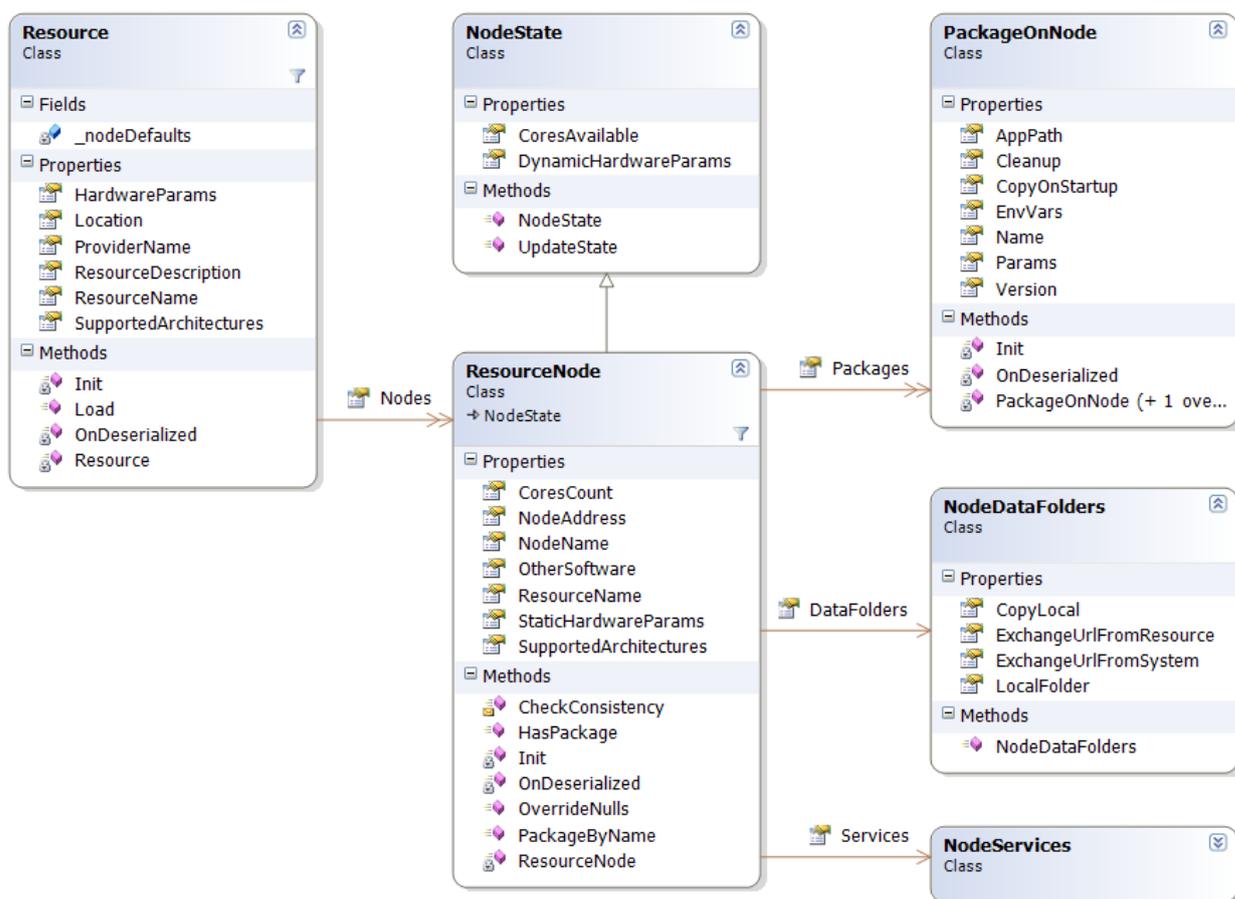


Рисунок 3.3 – Структура основных классов базы ресурсов

Ниже приводятся сокращенные описания структуры и методов основных классов компонента.

### 3.2.1. Класс *ResourceBaseService*

Главный класс базы ресурсов, реализующий ее интерфейс.

#### Основные открытые методы:

- **GetAllResources** – получить описание всех ресурсов, доступных платформе. Ввиду большого количества возвращаемых и пересылаемых результатов при использовании данного метода более предпочтительно использование других методов сервиса для получения информации о ресурсах
  - а) входных параметров нет;
  - б) возвращает массив описаний ресурсов (`Resource[]`).
- **GetResourceNames** – получить список имен имеющихся в системе ресурсов
  - а) входных параметров нет
  - б) возвращает массив строковых имен ресурсов (`string[]`).

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

- `GetResourceByName` – получить описание конкретного ресурса с указанным именем
  - a) входной параметр: имя ресурса, строка (`string`);
  - b) возвращается описание требуемого ресурса (`Resource`).
- `GetResourceNodeByName` – получить описание конкретного узла вычислительного ресурса
  - a) входные параметры: имя ресурса (`string`) и имя узла (`string`);
  - b) возвращается описание запрашиваемого вычислительного узла (`ResourceNode`).

### 3.2.2. Класс *Resource*

Основной класс, используемый для загрузки и хранения конфигурации ресурсов.

#### Основные открытые методы

- `Load` – загрузить и обработать описания ресурсов из файлов по указанным URL-адресам. Метод является статическим
  - a) входной параметр: список путей к конфигурационным файлам (тип: `IEnumerable<string>`);
  - b) возвращается список загруженных ресурсов (тип: `IEnumerable<Resource>`).

#### Основные открытые свойства

- `ResourceName` (тип: `string`) – уникальное имя ресурса.
- `ResourceDescription` (тип: `string`) – текстовое описание ресурса.
- `SupportedArchitectures` (тип: `IEnumerable<string>`) – список поддерживаемых ресурсов вычислительных архитектур (таких как SMP, MPP).
- `Location` (тип: `string`) – географическое местоположение, строковый параметр в формате «широта;долгота».
- `ProviderName` (тип: `string`) – тип ресурса, должен быть согласован со списком поддерживаемых ресурсов компонентом исполнения WF (`CLAVIRE/Executor`).
- `HardwareParams` (`IDictionary<string, string>`)
- `Nodes` (тип: `ReadOnlyCollection<ResourceNode>`) – список вычислительных узлов, принадлежащих ресурсу.

#### Основные закрытые поля

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

- `_nodeDefaults` (тип: `ResourceNode`) – значения по умолчанию для вычислительных узлов.

### 3.2.3. Класс *ResourceNode*

Класс, используемый для базовых операций работы и хранения информации о конфигурации узла вычислительного ресурса. Наследуется от класса `NodeState`.

#### Основные открытые свойства

- `nodeName` (тип: `string`) – имя узла, уникальное внутри ресурса.
- `resourceName` (тип: `string`) – имя ресурса, к которому принадлежит узел.
- `nodeAddress` (тип: `string`) – URL-адрес узла.
- `supportedArchitectures` (тип: `IEnumerable<string>`) – список поддерживаемых ресурсов вычислительных архитектур (таких как `SMP`, `GPGPU`).
- `services` (тип: `NodeServices`) – описание сервисов управления вычислительным узлом.
- `dataFolders` (тип: `NodeDataFolders`) – описание папок данных, которые задают процесс передачи файлов на узел при запуске задачи.
- `coresCount` (тип: `int`) – количество вычислительных ядер, которыми обладает данный узел.
- `otherSoftware` (тип: `ReadOnlyCollection<string>`) – список установленного дополнительного (по отношению к вычислительным пакетам) программного обеспечения. В качестве такого ПО могут выступать как различные библиотеки, используемые вычислительными пакетами, так и тип установленной операционной системы.
- `packages` (тип: `ReadOnlyCollection<PackageOnNode>`) – описание установленных на узле вычислительных пакетов.
- `staticHardwareParams` (тип: `IDictionary<string, string>`) – описание статических характеристик аппаратного обеспечения узла. Состав характеристик зависит от типа ресурса.

#### Основные открытые методы

- `HasPackage` – проверить, установлен ли на данном узле указанный пакет
  - а) входной параметр: имя пакета (тип: `string`);
  - б) возвращается результат проверки (тип: `bool`).

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

- **OverrideNulls** – перезаписать не установленные значения в описании пакета значениями по умолчанию
  - а) входной параметр: значения по умолчанию для вычислительных узлов (тип: ResourceNode);
  - б) возвращаемое значение отсутствует (void).
- **PackageByName** – получить описание установленного на узле пакета по его имени
  - а) входной параметр: имя пакета (тип: string);
  - б) возвращается описание пакета (тип: PackageOnNode).

#### **Основные закрытые методы**

- **CheckConsistency** – проверить корректность описания параметров узла.
  - а) входных параметров нет;
  - б) возвращаемое значение отсутствует (void). В случае некорректности описания узла выбрасывается исключение.

#### **3.2.4. Класс NodeState**

Базовый класс для хранения информации о состоянии узла.

##### **Открытые свойства**

- **CoresAvailable** (тип: int) – количество свободных на данный момент ядер на узле.
- **DynamicHardwareParams** (тип: IDictionary<string, string>) – описание динамических характеристик аппаратного обеспечения узла. Состав характеристик зависит от типа ресурса.

#### **3.2.5. Класс PackageOnNode**

Класс, используемый для описания конфигурации пакета на вычислительном узле.

##### **Открытые свойства**

- **Name** (тип: string) – идентификатор пакета.
- **Version** (тип: string) – версия пакета.
- **AppPath** (тип: string) – путь к исполняемому файлу пакета на узле.
- **EnvVars** (тип: IDictionary<string, string>) – описание переменных окружения, установка которых необходима для запуска пакета.

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

- Params (тип: IDictionary<string, string>) – дополнительные параметры, необходимые для запуска пакета на ресурсе, состав которых зависит от типа ресурса и самого пакета.
- Cleanup (тип: IEnumerable<string>) – список путей, которые необходимо удалить перед записью файлов в хранилище.
- CopyOnStartup (тип: IEnumerable<string>) – список путей, которые необходимо скопировать в рабочую папку расчета перед запуском.

### **3.2.6. Класс NodeDataFolders**

Класс описания конфигурации папок данных, используемых для передачи файлов на вычислительный узел.

#### **Открытые свойства:**

- ExchangeUrlFromService (тип: string) – URL-адрес (как правило, FTP), через который системные сервисы могут получить доступ к папке обмена.
- ExchangeUrlFromResource (тип: string) – URL-адрес (как правило, сетевая папка или FTP) папки обмена, по которому может быть осуществлен доступ с вычислительного узла.
- CopyLocal (тип: bool) – параметр, определяющий, требуется ли производить копирование файлов из папки обмена в локальную папку на вычислительном ресурсе.
- LocalFolder (тип: string) – локальная папка на вычислительном ресурсе, в которой в процессе вычислений создаются выходные файлы; может совпадать с папкой обмена, если параметр CopyLocal принимает значение false.

### **3.2.7. Класс NodeServices**

Класс, используемый для описания сервисов управления вычислительным узлом.

#### **Открытые свойства:**

- ExecutionUrl (тип: string) – URL-адрес конкретного сервиса управления ресурсом.

## **4. ИСПОЛЬЗУЕМЫЕ ТЕХНИЧЕСКИЕ СРЕДСТВА**

Компонент CLAVIRE/ResourceBase реализован в виде web-сервиса на основе технологии WCF платформы Microsoft .Net версии 4.0. Для установки и запуска

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

компонента требуется установленный web-сервер IIS версии не ниже 6.0. Программный компонент поставляется в виде .Net-сборок и предназначен для использования в рамках платформы .Net. Требования к аппаратной платформе диктуются требованиями реализации платформы .Net:

- x86-совместимый процессор с тактовой частотой 1 ГГц или выше;
- 512 МБ оперативной памяти или больше;
- минимальное место на диске: 850 МБ для x86 версии и 2 ГБ – для x64 версии платформы .Net.

## 5. ВЫЗОВ И ЗАГРУЗКА

Компонент база ресурсов реализован в виде SOAP WCF-сервиса платформы .Net. Для запуска сервиса применяется стандартный механизм используемого web-сервера. Загрузка сервиса в этом случае выполняется web-сервером автоматически по мере поступления запросов от клиентов сервиса.

Вызов сервиса производится стандартным для технологии WCF способом: по опубликованному описанию сервиса (WSDL) создается прокси-класс, через который осуществляется взаимодействие с сервисом путем вызова необходимых методов. Процедура создания прокси-класса зависит от того, на базе каких технологий строится клиентское приложение. Если выбраны язык программирования C# и платформа .NET, построение клиента производится за счет вызова служебного программного средства svcutil.exe, распространяемого в составе платформы .NET, либо за счет создания ссылки на сервис в среде Microsoft VisualStudio.

Пример текста программы, использующей сервис базы ресурсов, показан на рис. 5.1.

```
using System;  
using System.Collections.Generic;  
using System.Linq;  
using System.Text;
```

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

```
namespace RBClient
{
    class Program
    {
        static void Main(string[] args)
        {
            var rbService = new ResourceBaseService.ResourceBaseServiceClient();
            string[] resourceNames = rbService.GetResourceNames();

            foreach (string name in resourceNames)
            {
                var res = rbService.GetResourceByName(name);
                string nodeNames = String.Join(", ",
                    res.Nodes.Select(n => n.NodeName));
                Console.WriteLine("{0}: {1}", name, nodeNames);
            }
        }
    }
}
```

Рисунок 5.1– Пример использования сервиса базы ресурсов в клиентской программе

## 6. ВХОДНЫЕ ДАННЫЕ

Входными данными для компонента CLAVIRE/ResourceBase является список адресов конфигурационных файлов с описанием ресурсов, сами эти файлы, а также входные параметры вызова интерфейсных методов.

Пример входного конфигурационного файла приведён на листинге 6.1.

Листинг 6.1. Файлы конфигурации для кластера в Грид ННС

```
{
  "ResourceName": "Grid NNN",
  "ResourceDescription": "Грид ННС",
  "SupportedArchitectures": ["MPP", "SMP"],
  "ProviderName": "Grid NNN",
  "NodeDefaults":
  {
    "SupportedArchitectures": ["SMP"],
    "Services": { "ExecutionUrl": "https://tb01.ngrid.ru:5053" },
    "DataFolders":
    {
      "ExchangeUrlFromSystem": "ftp:// 192.168.1.189/{task}/{phase}",
      "ExchangeUrlFromResource": "ftp://@194.85.163.231/srv/ftp/{task}/{phase}",
      "CopyLocal": "true",
      "LocalFolder": "gsiftp://nnn1.pnpi.nw.ru/home/gridui060/files/jobs/{task}"
    },
    "CoresCount": 0,
    "HardwareParams":
    [
      { "Key": "CoresPerformance", "Value": "[2000]" }
    ]
  },
  "Nodes":
  [
    {
      "NodeName": "hp-cl.escience.ifmo.ru",
      "NodeAddress": "hp-cl.escience.ifmo.ru",

```

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник ссылки не найден.**

```

    "Packages":
    [
      {
        "Name": "ORCA", "Version": "v1",
        "AppPath": "orca",
        "EnvVars": [ {"Key": "PATH", "Value": "/grid/software/orca-2.6.35-
nompri/bin/:%$PATH"} ]
      }
    ]
  }
}

```

## 7. ВЫХОДНЫЕ ДАННЫЕ

Выходными данными компонента является структурированный список доступных платформе ресурсов, их узлов и настроек.

Пример ответа сервиса базы ресурсов приведён на рисунке 7.1. Для отправки запроса и получения ответа использовано стандартное средство «WCF Test Client», идущее в поставке с Microsoft Visual Studio 2010.

The screenshot shows the WCF Test Client interface. The left pane displays the service configuration for 'My Service Projects' with the endpoint 'http://localhost/ResourceBase/ResourceBase'. The right pane shows the 'GetResourceByName' operation with a request parameter 'resourceName' set to 'b4'. The response is a structured object containing various resource parameters.

Name	Value	Type
(return)		MITP.Resource
HardwareParams	length=1	System.Collections.Generic.Dictionary<System.String, System.Collections.DictionaryEntry>
[1]		
Key	"LatencyClusterNode"	System.String
Value	"1.0"	System.String
Location	"59.942649;30.29706"	System.String
NodeDefaults		MITP.ResourceNode
CoresCount	4	System.Int32
Credentials		MITP.NodeCredentials
DataFolders		MITP.NodeDataFolders
ExchangeUriFromSystem	"ftp://192.168.4.1/Tasks/{task}/{phase}"	System.String
ExchangeUriFromResource	"\\192.168.4.1\ftp_exchange\Tasks\\"	System.String
CopyLocal	False	System.Boolean
LocalFolder	"C:\Temp\{task}"	System.String
HardwareParams	length=1	System.Collections.Generic.Dictionary<System.String, System.Collections.DictionaryEntry>
[1]		
Key	"CoresPerformance"	System.String
Value	"[2930]"	System.String
OtherSoftware	length=1	System.String[]

Рисунок 7.1 - Ответ компонента CLAVIRE/ResourceBase на запрос описания ресурса «b4»

RU.СНАБ.80066-06 13 **Ошибка! Источник ссылки не найден.Ошибка! Источник  
ссылки не найден.**

## **ПЕРЕЧЕНЬ СОКРАЩЕНИЙ**

WSDL	Web Services Description Language, язык описания web-сервисов
WCF	Windows Communication Foundation
IIS	Internet Information Services
МИТП	Многопрофильная инструментально-технологическая платформа

