

July 8, 2001

## Contents

<b>1</b>	<b>Общие сведения</b>	<b>2</b>
1.1	Обзор	2
1.1.1	Назначение пакета	2
1.1.2	Платформы	2
1.1.3	Поддержка	2
1.2	Компоненты	2
1.2.1	Предмет поставки	2
1.2.2	Инсталляция	3
1.2.3	Компоненты, автоматически создающиеся в процессе использования пакета	4
<b>2</b>	<b>Порядок инсталляции пакета</b>	<b>4</b>
2.1	Необходимое ПО	4
2.2	Порядок инсталляции под UNIX	4
2.3	Порядок инсталляции под Windows NT	5
<b>3</b>	<b>Тестирование</b>	<b>5</b>
3.0.1	Общие сведения	5
3.0.2	Примерный порядок тестирования	6
<b>4</b>	<b>Использование</b>	<b>6</b>
4.1	Права доступа	6
4.2	Запуск сервера	7
4.2.1	Общие сведения	7
4.2.2	Опции настройки сервиса	8
4.2.3	Два способа запустить сервер под Windows NT	9
4.2.4	Примерный порядок запуска	10
4.2.5	Сообщения об ошибках, которые могут появиться при запуске сервера	11
4.3	Компиляция клиентских приложений	11
4.4	Запуск клиентских приложений	11

4.5	Риски безопасности . . . . .	12
<b>5</b>	<b>Удаление</b>	<b>12</b>
<b>6</b>	<b>История документа</b>	<b>12</b>

## **1 Общие сведения**

### **1.1 Обзор**

#### **1.1.1 Назначение пакета**

HostProcControl - это CORBA сервис, предоставляющий API для удаленного управления операционной системой. С помощью HostProcControl вы можете создавать CORBA-объекты для таких вещей как:

1. *доступ к файловой системе сервера:*
  - (a) чтение и запись файлов;
  - (b) удаление файлов;
  - (c) проверка статуса файлов;
2. *доступ к процессам на сервере:*
  - (a) получение списка процессов;
  - (b) получение параметров выделенных процессов;
  - (c) запуск процессов;
  - (d) посылка сигналов.

#### **1.1.2 Платформы**

Пакет HostProcControl компилируется с такими моделями ORB как TAO, omniORB и ORBacus для UNIX, а также (пока) с ORBacus и только для Windows NT.

#### **1.1.3 Поддержка**

HostProcControl разрабатывается и поддерживается компанией GradSoft, домашняя страница GradSoft <http://www.gradsoft.com.ua>. Представленная версия продукта - **HostProcControl-1.0**.

### **1.2 Компоненты**

#### **1.2.1 Предмет поставки**

В дистрибутив пакета входят:

1. описание реализованного нами API в виде модуля HostControl.idl

2. компоненты в исходном коде:
  - (a) сервер **HostControlServer** ;
  - (b) вспомогательное ПО:
    - i. инсталлер **NtInstaller** для установки HostControlServer как Windows NT сервиса;
    - ii. примеры клиентских приложений;
3. средства управления компиляцией под Windows NT
4. средства управления компиляцией под UNIX
5. Документация, в т.ч.:
  - (a) Руководство программиста;
  - (b) Руководство администратора (этот файл);

Все перечисленные компоненты собраны в архив, который называется **HostProcControl-1.0** и может быть архивом типа **.zip** или **.tar.gz** в зависимости от сделанного Вами выбора.

### 1.2.2 Инсталляция

Инсталлированный пакет содержит:

1. idl-модуль HostControl.idl
2. заголовочный файл стаба
3. статические библиотеки C++ :

платформа	имя библиотеки	содержание библиотеки
UNIX	libHostControlClient.a	стаб
UNIX	libHostControl.a	стаб, скелетон, серванты
Windows NT	HostControlClient.lib	стаб
Windows NT	HostControl.lib	стаб, скелетон, серванты

4. исполняемые файлы:
  - При инсталляции под UNIX : сервер HostControlServer
  - При инсталляции под Windows NT : сервер HostControlServer.exe и, дополнительно, инсталлер сервиса NtInstaller.exe

Эти средства автоматически генерируются из исходного кода (все за исключением HostControl.idl), после чего копируются в фиксированные подкаталоги специального "каталога инсталляции", имя которого определяется пользователем. Имена подкаталогов, в которых будут размещены idl-модуль, .h-файл, библиотеки и выполняемые файлы фиксированы: это "idl", "include", "lib" и "bin" соответственно. Данные подкаталоги должны быть созданы пользователем до начала инсталляции.

### 1.2.3 Компоненты, автоматически создающиеся в процессе использования пакета

При инсталляции сервера в качестве NT сервиса без спроса создается конфигурационный файл **hpcservice.ini**, в директории Windows (обычно C:\WinNT). Этот файл необходим для автоматического запуска сервиса при старте системы.

## 2 Порядок инсталляции пакета

### 2.1 Необходимое ПО

1. CORBA ORB:
  - Unix: omniORB-3.0 или выше, *или* TAO-5.9b или выше, *или* ORBacus 4.0.2 или выше
  - Windows NT: ORBacus версии 4.x или выше.
2. Компилятор C++ :
  - Unix: gcc-2.95.2 или выше, *или* SunProc C++ 4.2
  - Windows NT: Microsoft Visual C версии 6.0 или выше.
3. make:
  - Unix: gnu make
  - Windows NT: nmake из поставки MSVC++
4. Специальное программное обеспечение: пакет GradC++ToolBox версии 1.0.3 или выше в конфигурации, включающей компоненты ProgOptions и ServiceOptions

### 2.2 Порядок инсталляции под UNIX

- Убедиться, что необходимое ПО установлено и работает в текущем окружении пользователя.
- Развернуть архив `HostProcControl.tar.gz` в избранном Вами каталоге (в дальнейшем этот каталог мы будем называть "корневым каталогом проекта", `<project_root>` ).
- Перейти в каталог `<project_root>`
- Запустить `configure`, командой `./configure`
- При необходимости указать опции конфигурации. Список возможных опций конфигурации выводится при запуске `configure` с опцией `--help`.
- Для компиляции пакета воспользоваться командой `gmake` (или `make`)

- Для инсталляции пакета воспользоваться командой `gmake install`
- Для удаления файлов, созданных в процессе компиляции перейти в каталог `<project_root>` и воспользоваться командой `make clean`

## 2.3 Порядок инсталляции под Windows NT

- Убедиться, что необходимое ПО установлено и работает. Прописать пути к утилитам `nmake`, `cl` и `xcopy` в переменной среды пользователя `PATH`.
- Развернуть архив `HostProcControl.tar.gz` в избранном Вами каталоге (в дальнейшем этот каталог мы будем называть "корневым каталогом проекта", `<project_root>` ).
- Отредактировать файл `env_inc.nt.mak` в подкаталоге `<project_root>\config`; При редактировании установить значения следующих `nmake`-переменных:

<i>имя переменной</i>	<i>описание</i>
<code>PROJECT_ROOT</code>	<code>&lt;project_root&gt;</code> (корневой каталог проекта)
<code>INSTALL_DIR</code>	Каталог инсталляции
<code>ORB_DIR</code>	Корневой каталог ORB
<code>MSVC_DIR</code>	Корневой каталог Microsoft Visual Studio
<code>GEN_DIR</code>	Каталог инсталляции пакета <code>GradC++ToolBox</code>

- Для компиляции пакета перейти в каталог `<project_root>` и воспользоваться командой `make`
- Для инсталляции пакета перейти в каталог `<project_root>` и воспользоваться командой `make install`
- Для удаления файлов, созданных в процессе компиляции перейти в каталог `<project_root>` и воспользоваться командой `make clean`

## 3 Тестирование

### 3.0.1 Общие сведения

Мы предлагаем 6 тестовых примеров, расположенных в подкаталогах `1,2,...6` каталога `<project_root>/test` и озаглавленных `step1, step2, ... step6` соответственно. Описание каждого примера находится в соответствующем ему файле `readme<number>.txt` (где `<number>=1,2,...6`), расположенном в том же подкаталоге.

### 3.0.2 Примерный порядок тестирования

1. Откомпилировать пакет HostProcControl как описано в подразделе "Порядок инсталляции" для Вашей платформы.
2. Собрать примеры клиентских приложений, для чего перейти в каталог `<project_root>/test` и воспользоваться командой `make`
3. Перейти в каталог `<project_root>/src/main` и запустить HostControlServer со следующими опциями:

```
--userlist <project_root>/test/HostControlUserlist.ini -OApport 16001
```

(смысл этих опций смотри в разделе "Использование" 4)

4. Убедиться в работоспособности примеров, для чего последовательно перейти в каждый из подкаталогов 1,2,...6 каталога `<project_root>/test` и осуществить там следующие действия:
  - прочитать описание примера в файле `readme<number>.txt` где `<number>` - имя подкаталога;
  - запустить исполняемый файл `step<number>` (для UNIX) либо `step<number>.exe` (для Windows NT) с опцией  

```
-ORBInitRef HostControlService=corbaloc::127.0.0.1:16001/HostControlService
```
  - убедиться, что работа примера соответствует описанию.
5. Удалить бинарные файлы, для чего перейти в каталог `<project_root>/test` и воспользоваться командой `make clean`

## 4 Использование

### 4.1 Права доступа

В HostProcControl авторизация доступа осуществляется так:

1. Существует т.н. "список пользователей" сервиса. Это файл, который создается *перед* запуском сервера, может редактироваться в процессе его работы и, вообще говоря, состоит из строк следующего формата:

```
[space]<name>[space]:[space]<password>[space]
```

где `<name>` и `<password>` - "имя" пользователя и его "пароль".

- Пример списка пользователей:

```
John:Johnson
nsh:gsh
another_name : another_password
```

- Замечание: общее количество символов в отдельной строке файла (включая пробелы) не должно быть больше 255.

2. Для того, чтобы получить доступ к функциональности сервера, клиентское приложение должно предоставить комплексный "ID клиента", состоящий из имени пользователя и его пароля. По получении этого ID, аналогичное ID ищется в списке пользователей сервиса. Доступ предоставляется, если аналогичное ID будет найдено и не предоставляется, если нет.

Важно: Путь к списку пользователей должен быть зафиксирован при старте сервера; рекомендуется использовать полный путь.

## 4.2 Запуск сервера

### 4.2.1 Общие сведения

- Что надо сделать при запуске сервера:

Три вещи:

1. указать путь к списку пользователей;
2. согласовать способ, которым клиентские приложения будут получать инициальную ссылку сервиса;
3. при необходимости - передать параметры ORB.

- Как это можно сделать:

При помощи опций настройки сервиса 4.2.2, которые следует передать в `HostControlServer` одним из двух способов:

1. в командной строке,
2. в файле конфигурации.

- Общие сведения об опциях командной строки:

В командной строке должны находиться ЛИБО опции настройки сервиса, ЛИБО особые (управляющие) опции. Управляющих опций две: `--config <filename>` и `--help`. Опция `--config <filename>` указывает, что опции настройки сервиса находятся в файле конфигурации `<filename>`, опция `--help` предотвращает запуск сервера и обеспечивает вывод на экран стандартного help-сообщения.

- Общие сведения о файле конфигурации:

1. Файл конфигурации используется если командная строка пуста или содержит управляющую опцию  
`--config <filename>`  
где `<filename>` - имя файла.

2. Если командная строка пуста, имя файла назначается по умолчанию:  
`HostControlConfig.ini`
3. Всегда, когда имя файла НЕ является полным (в частности, всегда когда командная строка пуста), файл конфигурации ищется относительно "базы". По умолчанию, "базой" считается фиксированный каталог диска: каталог Windows для Windows NT (как правило, `C:\WINNT`), либо каталог `/etc` для UNIX. Для того, чтобы сменить базу, нужно выбрать другой каталог и сохранить его имя в системной переменной `HPC_CONFIG`.
4. В любом случае файл конфигурации должен содержать полный набор опций, которые необходимы для настройки механизма авторизации доступа и конфигурирования ORB.
5. Формат файла конфигурации описан в руководстве программиста к пакету ProgOptions (<http://www.gradsoft.kiev.ua/rus/Products/ToolBox/ProgOptions/ProgGuid>). Пример файла конфигурации см. ниже 3.

#### 4.2.2 Опции настройки сервиса

1. Имя и положение списка пользователей:

Имя файла-списка пользователей и передаются серверу как аргумент опции

```
--userlist <filename>
```

Если это имя НЕ является полным, список пользователей ищется относительно той же базы, что и конфигурационный файл 4.2.1.

2. Инициальная ссылка:

Публикация инициальной ссылки нашего сервиса осуществляется при помощи специального программного средства ServiceOptions, входящего в состав пакета GradC++ ToolBox. ServiceOptions автоматически обеспечивает доступ к серванту через объектную ссылку в стиле corbaloc и, дополнительно, позволяет опубликовать IOR тремя способами:

- (a) если задана опция `--with-naming`, то инициальный объект отображаются в `NamingService`;
- (b) если задана опция `--ior-stdout`, то объектная ссылка преобразуется в строку и печатается на стандартном выводе программы;
- (c) если задана опция `--ior-file-HostControlService <filename>`, то объектная ссылка преобразуется в строку и помещается в файл с именем `<filename>`.

3. Параметры ORB:

Для того, чтобы клиент мог использовать ссылку в стиле corbaloc, Вы уже должны передать серверу как минимум один дополнительный параметр: Вы должны указать порт, через который будет осуществляться связь между компьютерами, при помощи обычной опции ORB:



- `-OAport`, если вы используете ORBacus
- `-ORBport`, если вы используете ТАО
- `-ORBpoa_iiop_port`, если вы используете omniORB

Другие параметры ORB можно также задать при помощи обычных для каждого случая опций, однако надо помнить, что даже в том случае, когда для запуска сервера используется конфигурационный файл, синтаксис опций должен быть таким же, как если бы они были заданы в командной строке. Определить параметры ORB при помощи тех синтаксических конструкций, которые используются в конфигурационном файле ORB-формата, можно при помощи дополнительного конфигурационного файла ORB-формата, который можно подключить при помощи опции ORB `"-ORBconfig"`.

Пример:

```
./HostControlServer --userlist HostControlUserlist.ini -ORBconfig orb.cf
```

где в `orb.cf`, допустим, следующая конструкция:

```
ooc.orb.server_timeout=2
ooc.orb.oa.port=1025
```

### 4.2.3 Два способа запустить сервер под Windows NT

Для работы под управлением Windows NT сервер `HostControlServer` можно запустить двумя способами:

1. Как консольное приложение
2. Как NT сервис

причем опции, с которыми эта программа должна быть запущена, в обоих случаях те же. Для запуска программы в качестве консольного приложения вы просто командуете `"HostControlServer"` с нужными опциями. Для того, чтобы инсталлировать и запустить `HostControlServer` как NT сервис используется специальная программа `NtInstaller.exe`. Порядок использования `NtInstaller.exe` следующий:

1. Запустить `NtInstaller.exe` – всплывёт диалоговое окно, в котором необходимо указать следующие вещи:
  - (a) имя сервиса, под которым его можно видеть из служб Windows NT;
  - (b) путь к бинарному файлу `HostControlServer.exe`;
  - (c) командную строку, с которой сервис будет запущен.

2. Для инсталляции сервиса нажать на кнопку **Install and Start** – сервер будет проинсталлирован как служба и запущен с указанными Вами параметрами.  
Детали:
  - (а) Если какая-то копия сервера уже была инсталлирована ранее, имеющийся сервис будет удален и инсталляция будет проведена заново.
  - (б) Сервис устанавливается в режиме автозапуска, т.е. он будет автоматически запускаться при каждой перезагрузке системы.
3. Для получения информации об инсталлированном сервисе нажать на кнопку **Service information** – выводится статус сервиса, а в случае его работы - список аргументов, с которыми он был запущен.
4. Для удаления сервиса нажать на кнопку **Uninstall**.

#### 4.2.4 Примерный порядок запуска

Пусть, для определенности, мы работаем под управлением Windows NT и сервер надо запустить как консольное приложение. Тогда оптимальный, на наш взгляд, способ запуска будет следующим:

1. Назначить базу - выбрать каталог `<base>`, в котором будет находиться конфигурационный файл `HostControlConfig.ini` и, в нашем случае, файл - список пользователей сервиса, и сохранить имя этого каталога в системной переменной `HPC_CONFIG`.
2. Создать файл `<base>\HostControlUserlist.ini` с именами и паролями пользователей (- отметим, что на этапе запуска этот файл может быть пустым, и может быть заполнен после запуска сервера).
3. Создать файл `<base>\HostControlConfig.ini` следующего формата (подробно см. <http://www.gradsoft.kiev.ua/rus/Products/ToolBox/ProgOptions/ProgGuide/Programmm>):

```
@ "ProgOptions config file" // Эта штука должна стоять в начале первой строки

# This is config file for HostControlService realized by executable:

    HostControlServer.exe    /* Первым "значащим" словом файла
                             должно быть имя программы */

# users list:

    --userlist HostControlUserlist.ini // Файл - список пользователей

# options for ORB:
```

```
-Oport 16001 /* Внимание: опция ORB в файле задана так же,  
как если бы она была задана в командной строке */
```

: это ПРИМЕР, но он может быть скопирован и использован без поправок

4. Запустить HostControlServer.exe без всяких опций.

#### 4.2.5 Сообщения об ошибках, которые могут появиться при запуске сервера

	<i>Сообщение</i>	<i>Описание ошибки</i>	<i>Действия программы</i>
1	error of StartServiceCtrlDispatcher	только для WinNT: Ошибка старта сервиса, запускаемого менеджером сервисов	завершение работы
2	error of parsing of <filename>	ошибочный формат файла конфигурации	завершение работы
3	filename necessary is NULL	отсутствует аргумент опции "--config" либо "--userlist"	завершение работы
4	file to read <filename> does not exist	неправильно задано полное имя файла конфигурации либо файла-списка пользователей	завершение работы
5	file to read <filename> does not exist in base directory <directory_name>	файл конфигурации либо файл-список пользователей не найден в базовом каталоге	завершение работы
6	CORBA System Exception	ошибочные опции ORB	завершение работы
7	error while orb destroing	ошибка разрушения ORB, возможна при завершении работы программы	никаких
8	Server has been activated	нормальный старт	запуск сервера

#### 4.3 Компиляция клиентских приложений

Клиентские приложения, использующие HostProcControl, должны быть скомпилированы с библиотекой HostControlClient.lib (для Windows NT) либо libHostControlClient.a (для UNIX) в дополнение к обычным библиотекам ORB, включая NamingService.

#### 4.4 Запуск клиентских приложений

Помни об опциях, с которыми был запущен сервер!

## 4.5 Риски безопасности

*Внимание:* клиенты HostProcControl могут получить полный контроль над операционной системой сервера. Поэтому

1. думайте о рисках безопасности;
2. следуйте правилам:
  - Используйте этот сервис в защищенных сетях;
  - При необходимости использовать комплекс в незащищенных системах прибегайте к технологии шифрования трафика (к примеру VPN или CORBA Security Service).

## 5 Удаление

1. Только для Windows NT:
  - (a) запустить NtInstaller и удалить NT сервис HostControl, если он установлен;
  - (b) удалить файл hpcservice.ini в каталоге Windows (обычно C:\WinNT), если такой существует.
2. Удалить конфигурационный файл и список пользователей, если они создавались;
3. Удалить системную переменную HPC\_CONFIG, если она создавалась;
4. Удалить каталог инсталляции либо файлы типа HostControl<smth> из данного каталога; список файлов, которые должны быть уничтожены, смотри в разделе **Компоненты** 1.2.2.
5. Удалить каталог, в котором вы развернули исходный архив проекта.

## 6 История документа

09.04.2001 - текущий релиз; основные отличия от документа, вошедшего в состав предыдущего релиза **HostProcControl-1.0b2**:

1. Расширен раздел **Общие сведения**;
2. Добавлен раздел **Удаление**.

24.02.2001 - первая редакция.