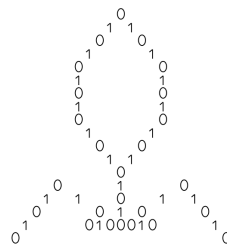




UNIVERSIDAD NACIONAL DE RÍO CUARTO  
FAC. DE CS. EXACTAS, FCO-QCAS Y NATURALES  
DEPARTAMENTO DE COMPUTACIÓN



FUDEPAN  
FUNDACIÓN PARA EL DESARROLLO DE LA  
PROGRAMACIÓN EN ÁCIDOS NUCLEICOS

TRABAJO FINAL  
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

---

## Manual de compilación e instalación de FuD-BOINC

---

*Autores:*  
Lucas BESSO  
Raúl STRIGLIO

*Director:*  
Lic. Laura TARDIVO

*Co-Director:*  
Daniel GUTSON

*Última actualización:*  
20 de diciembre de 2011

# Índice general

<b>I</b>	<b>Manual de instalación y compilación de FuD-BOINC</b>	<b>1</b>
<b>1.</b>	<b>Introducción</b>	<b>2</b>
<b>2.</b>	<b>BOINC</b>	<b>3</b>
2.1.	Dependencias requeridas . . . . .	3
2.1.1.	Paquetes requeridos por cliente y servidor . . . . .	3
2.1.2.	Paquetes requeridos por el servidor . . . . .	4
	Instalación de Apache . . . . .	4
2.1.3.	Paquetes requeridos por el cliente . . . . .	5
2.2.	Configuración de MySQL Server . . . . .	6
2.3.	Descarga del código fuente . . . . .	6
2.4.	Compilación . . . . .	6
2.4.1.	Crear un proyecto . . . . .	7
2.4.2.	Compilar el cliente BOINC . . . . .	7
2.4.3.	Desarrollar aplicaciones BOINC . . . . .	7
2.5.	Proyecto BOINC . . . . .	7
2.5.1.	Dependencias requeridas . . . . .	7
2.5.2.	Crear un proyecto . . . . .	7
2.5.3.	Configuración del <code>crontab</code> . . . . .	8
2.5.4.	Configuración de un proyecto . . . . .	8
2.5.5.	Agregar aplicaciones al proyecto . . . . .	8
	Agregar nuevas versiones de una aplicación . . . . .	9
<b>3.</b>	<b>FuD-BOINC</b>	<b>10</b>
3.1.	Dependencias requeridas . . . . .	10
3.1.1.	Dependencias de FuD . . . . .	10
	Mili . . . . .	10
	Boost 1.42 o superior . . . . .	10
3.1.2.	Dependencias de FuD-BOINC . . . . .	10
3.2.	Descarga del código fuente . . . . .	11
3.3.	Compilación . . . . .	11
3.4.	Instalación . . . . .	12

---

<b>4. Aplicación de prueba Parallel-Clusterer</b>	<b>13</b>
4.1. Dependencias requeridas . . . . .	13
4.1.1. Biopp . . . . .	13
4.1.2. Prot-filer . . . . .	13
4.1.3. Getopt_pp . . . . .	14
4.1.4. xdrfile . . . . .	14
4.1.5. Feca . . . . .	14
4.2. Descarga del código fuente . . . . .	14
4.3. Compilación . . . . .	14
4.4. Ejecución . . . . .	15

# Parte I

## Manual de instalación y compilación de FuD-BOINC

# Capítulo 1

## Introducción

Este documento describe los pasos que se deben seguir para poder compilar la librería FuD con la capa de distribución FuD-BOINC. Para lograr ésto, se explica cómo descargar BOINC, cómo compilarlo y cómo crear un proyecto de computación voluntaria en donde se puedan correr las aplicaciones desarrolladas con FuD-BOINC. Por último, se explican los pasos a seguir para descargar y compilar FuD con las librerías BOINC ya compiladas.

El manual pretende ser un medio simple en donde se integren los pasos necesarios para comenzar a utilizar FuD-BOINC por lo que si se desea extender algunos conceptos y/o instrucciones aquí detalladas recomendamos consultar la información oficial que será incluida con cada sección.

# Capítulo 2

## BOINC

### 2.1. Dependencias requeridas

Para poder compilar las librerías de BOINC es necesario resolver los requisitos previos del framework.

La información de esta sección está basada de la wiki oficial de BOINC<sup>1</sup>.

A continuación se especifican los paquetes necesarios para compilar BOINC basado en sistemas Unix/Linux.

#### 2.1.1. Paquetes requeridos por cliente y servidor

- m4
- make
- autoconf
- automake1.9
- gcc-4.1
- gcc
- g++-4.1
- pkg-config
- libtool
- subversion
- vim

---

<sup>1</sup><http://boinc.berkeley.edu/trac/wiki/ServerIntro#cookbook-debian40>

### 2.1.2. Paquetes requeridos por el servidor

- apache2-mpm-prefork
- libapache2-mod-php5
- mysql-query-browser
- mysql-client-5.0
- mysql-server-5.0
- php5-mysql
- php5-cli
- php5-gd
- phpmyadmin
- python-mysqldb
- libmysql++-dev
- libssl-dev
- openssl

### Instalación de Apache

Para instalar apache y las dependencias para con BOINC<sup>2</sup> instalar los siguientes paquetes:

- apache2
- php5
- libapache2-mod-auth-mysql
- php5-mysql
- phpmyadmin
- libfcgi-dev
- libapache2-mod-fcgid

---

<sup>2</sup>Para más información consultar: [http://www.guia-ubuntu.org/index.php?title=Servidor\\_web](http://www.guia-ubuntu.org/index.php?title=Servidor_web)

- php5-cgi
- libcurl3-openssl-dev
- glutg3-dev
- libXmu-dev
- libXi-dev
- libjpeg8-dev

### 2.1.3. Paquetes requeridos por el cliente

- libssl-dev
- libglut3-dev
- glutg3-dev
- libglui-dev
- libglitz-glx1-dev
- libsdl1.2-dev
- libcurl3-dev
- freeglut3
- freeglut3-dev
- libsm-dev
- libice-dev
- libxmu-dev
- libxi-dev
- libx11-dev
- libjpeg62-dev
- libgtk2.0-0
- libgtk2.0-0-dev



## 2.2. Configuración de MySQL Server

Para definir una nueva contraseña del usuario root hacer lo siguiente:

```
mysqladmin -h localhost -u root password mysqlrootpw {or own}
```

Para crear un nuevo usuario en la base de datos hacer lo siguiente:

```
mysql -h localhost -u root -p  
> GRANT ALL ON *.* TO 'boincadm'@'localhost';  
> SET PASSWORD FOR 'boincadm'@'localhost' = '';
```

Los permisos deberían ser limitados a la base de datos del proyecto después. Aquí, la definición de una contraseña vacía simplifica el proceso de instalación la cual luego puede ser modificada.

## 2.3. Descarga del código fuente

El código fuente de BOINC se encuentra almacenado en un repositorio de Subversion (SVN). Correr el siguiente comando para obtener la última versión estable:

```
svn co http://boinc.berkeley.edu/svn/trunk/boinc
```

Para más información consultar la wiki oficial de BOINC<sup>3</sup> que menciona este tema.

## 2.4. Compilación

Para más información sobre cómo compilar BOINC puede consultar la wiki oficial de BOINC<sup>4</sup>.

Una vez descargado el código fuente, se puede compilar el software BOINC escribiendo lo siguiente:

```
./_autosetup  
./configure  
make
```

en el directorio principal, `boinc/`.

Puede consultar todas las opciones del script `configure` en la wiki oficial<sup>5</sup>.

<sup>3</sup><http://boinc.berkeley.edu/trac/wiki/SourceCode>

<sup>4</sup><http://boinc.berkeley.edu/trac/wiki/BuildSystem>

<sup>5</sup><http://boinc.berkeley.edu/trac/wiki/BuildSystem#Configuration>

### 2.4.1. Crear un proyecto

Si lo que se desea es únicamente crear un proyecto, deberá compilar solamente el servidor:

```
./configure --disable-client --disable-manager
```

### 2.4.2. Compilar el cliente BOINC

Si lo que se desea es compilar el cliente BOINC para una plataforma específica ejecutar lo siguiente:

```
./configure --disable-server
```

### 2.4.3. Desarrollar aplicaciones BOINC

Para poder desarrollar aplicaciones BOINC, como en el caso de FuD-BOINC, se debe compilar de la siguiente manera:

```
./configure --disable-server --disable-client --disable-manager
```

## 2.5. Proyecto BOINC

A continuación se resumen los pasos para la creación de un proyecto BOINC.

### 2.5.1. Dependencias requeridas

Para poder crear un proyecto es necesario haber compilado el código fuente de BOINC tal como se indica en la sección 2.4.1. Para poder efectuar ese paso, es necesario considerar las dependencias mencionadas en la sección 2.1.

### 2.5.2. Crear un proyecto

Toda la información aquí detallada fue tomada de la wiki oficial<sup>6</sup>. Consultar allí para más detalles.

Para crear un proyecto BOINC es necesario utilizar la herramienta `make_project` proporcionada por BOINC. Para más detalles consultar la wiki oficial que habla sobre esta herramienta<sup>7</sup>.

A partir de la documentación oficial, durante el desarrollo de la tesis FuD-BOINC se ha escrito un simple script que resume los pasos necesarios para que un proyecto

---

<sup>6</sup><http://boinc.berkeley.edu/trac/wiki/CreateProjectCookbook>

<sup>7</sup><http://boinc.berkeley.edu/trac/wiki/MakeProject>

quede funcionando. Estos pasos incluyen desde la creación de un proyecto hasta la configuración de permisos de algunos directorios. Dentro del directorio `Tools/` puede encontrar el script `createProject` el cual, al ejecutarlo con los parámetros correctos, permite la creación de un proyecto. El script debe ser ejecutado de la siguiente manera:

```
sudo ./createProject NOMBRE_PROYECTO 'DESCRIPCION_PROYECTO'
```

y su función es crear un proyecto con el nombre y descripción pasados como parámetros, y de cambiar los permisos de las carpetas para que apache pueda escribir en esos directorios<sup>8</sup>.

**Nota:** `NOMBRE_PROYECTO` indica el nombre del proyecto a crear, y `DESCRIPCION_PROYECTO` la descripción de dicho proyecto. El proyecto se creará en la ruta: `/projects/NOMBRE_PROYECTO`.

### 2.5.3. Configuración del crontab

Luego de crear el proyecto, es importante configurar el crontab. Para ello, agregar la siguiente línea con el comando `crontab -e`:

```
00,05,10,15,20,25,30,35,40,45,50,55 * * * * ~/projects/NOMBRE_PROYECTO/bin/start --c  
>/dev/null
```

### 2.5.4. Configuración de un proyecto

La configuración de tu proyecto BOINC es controlada por un archivo llamado `/projects/NOMBRE_PROYECTO/config.xml` que es creado por el script `make_project`. Es necesario editar este archivo para iniciar un proyecto.

Se recomienda consultar información detallada sobre cómo configurar este archivo en la wiki oficial de BOINC<sup>9</sup>.

### 2.5.5. Agregar aplicaciones al proyecto

Para agregar las aplicaciones se deben efectuar los siguientes pasos:

- Editar el archivo de configuración `config.xml` y eliminar los items relacionados a la aplicación de prueba `uppercase`.
- Agregar la aplicación al archivo `project.xml`.
- Asegurarse que la base de datos del proyecto esté disponible.

<sup>8</sup><http://boinc.berkeley.edu/trac/wiki/ServerIntro/>

<sup>9</sup><http://boinc.berkeley.edu/trac/wiki/ProjectConfigFile/>

- Ejecutar los siguientes comandos:

```
cd ~/projects/NOMBRE\_PROYECTO/  
bin/xadd  
bin/update_versions
```

y seleccionar que si (y) en todos los pasos que se pregunten.

### Agregar nuevas versiones de una aplicación

Para agregar una nueva versión de una aplicación se deben seguir los siguientes pasos:

- Agregar la aplicación al archivo `project.xml` y ejecutar `bin/xadd`.
- Escribir una aplicación BOINC y compilarla para la plataforma que desees.
- Copiar el ejecutable a `/projects/NOMBRE_PROYECTO/apps/APPNAME`
- Instalar la nueva versión ejecutando los mismos comandos mencionados en la sección 2.5.5.
- Reiniciar el proyecto.

# Capítulo 3

## FuD-BOINC

### 3.1. Dependencias requeridas

Para poder compilar la librería FuD-BOINC es necesario resolver los requisitos de la misma. A continuación se especifican los paquetes necesarios para compilar la librería sobre sistemas Unix/Linux.

#### 3.1.1. Dependencias de FuD

##### Mili

El código fuente de Mili se encuentra alojado en un repositorio de Subversion en googlecode<sup>1</sup>. Para descargar la última versión del trunk correr el siguiente comando:

```
svn checkout http://mili.googlecode.com/svn/trunk/ mili-read-only
```

Para conocer los pasos de instalación, consultar el archivo README.

##### Boost 1.42 o superior

Descargar e instalar los paquetes:

- libboost-system
- libboost-thread

#### 3.1.2. Dependencias de FuD-BOINC

Descargar e instalar los paquetes:

---

<sup>1</sup><http://code.google.com/p/mili/>

- glibc
- libssl-dev
- libmysqlclient-dev

Se deben tener correctamente instaladas las siguientes librerías de BOINC:

- librerías de BOINC
- libboinc
- libboinc\_api
- libsched
- libboinc\_crypt

## 3.2. Descarga del código fuente

El código fuente de FuD-BOINC se encuentra almacenado en un repositorio de Subversion (SVN). Correr el siguiente comando para obtener la última versión estable:

```
svn checkout https://fud.googlecode.com/svn/branches/boinc FuD-BOINC
```

Si bien el código fuente de FuD-BOINC es descargado desde un branch, en un futuro estará disponible en la versión trunk de FuD.

## 3.3. Compilación

Una vez descargado el código fuente se debe proceder con los siguientes pasos para la compilación e instalación de la librería FuD-BOINC:

1. Crear un directorio donde se compilará el código fuente.
2. Ingresar al nuevo directorio.
3. Ejecutar el comando:

```
cmake -Dmiddleware=boinc [options] PATH_TO_FuD-BOINC_Source
```

Debemos destacar que la opción **-Dmiddleware=boinc** debe ser utilizada para compilar con la capa de distribución implementada con BOINC. Si ésta opción no se la especifica, se compilará con la implementación por defecto(ASIO).

A continuación se destacan las opciones (**options**) de compilación de FuD-BOINC:

- **-Dboinc\_source=PATH**: utilizar ésta opción para especificar el PATH del directorio de código fuente de BOINC. Por defecto PATH = “ /boinc”
- **-DCMAKE\_BUILD\_TYPE=Debug**: especificar ésta opción si se desea compilar con el flag para dar soporte a la depuración.
- **-DCMAKE\_COVER\_ON=on**: especificar ésta opción si se desea compilar con los flags para dar soporte a la cobertura de código.

## 3.4. Instalación

Una vez compilado, solo se debe ejecutar lo siguiente:

```
sudo make install
```

## Capítulo 4

# Aplicación de prueba Parallel-Clusterer

Para poder compilar la aplicación Parallel-Clusterer es necesario resolver los requisitos de la misma. A continuación se especifican los paquetes necesarios para compilar la aplicación sobre sistemas Unix/Linux.

### 4.1. Dependencias requeridas

#### 4.1.1. Biopp

El código fuente de Biopp se encuentra almacenado en un repositorio de Mercurial (hg). Correr el siguiente comando para obtener la última versión estable:

```
hg clone https://code.google.com/p/biopp/
```

Luego reemplazar el archivo Makefile del directorio descargado por el que se provee en el directorio `Makefiles/Biopp` del CD.

Para su instalación: ejecutar el comando “make” y luego “make install”.

#### 4.1.2. Prot-filer

El código fuente de Prot-filer se encuentra almacenado en un repositorio de Mercurial (hg). Correr el siguiente comando para obtener la última versión estable:

```
hg clone https://code.google.com/p/prot-filer/
```

Luego reemplazar el archivo Makefile del directorio descargado por el que se provee en el directorio `Makefiles/Prot-filer`.

Para su instalación: ejecutar el comando “make” y luego “make install”.



### 4.1.3. Getopt\_pp

El código fuente de Getopt\_pp se encuentra almacenado en un repositorio de Mercurial (hg). Correr el siguiente comando para obtener la última versión estable:

```
hg clone https://code.google.com/p/getoptpp/
```

Para su instalación: seguir los pasos indicados en el archivo README. Debemos destacar que se deben compilar librerías estáticas.

### 4.1.4. xdrfile

Descargar la librería desde la siguiente dirección: <http://download.fedora.redhat.com/pub/fedora/linux/releases/15/Everything/source/SRPMS/xdrfile-1.1-4.fc15.src.rpm>.

Para su instalación: seguir los pasos indicados en el archivo INSTALL.

### 4.1.5. Feca

El código fuente de feca se encuentra almacenado en un repositorio de Mercurial (hg). Correr el siguiente comando para obtener la última versión estable:

```
hg clone https://code.google.com/p/feca/
```

Luego reemplazar el archivo Makefile del directorio descargado por el que se provee en el directorio Makefiles/feca.

Para su instalación: ejecutar el comando “make” y luego “make install”.

## 4.2. Descarga del código fuente

El código fuente de la aplicación Parallel-clusterer se encuentra almacenado en un repositorio de Subversion (SVN). Correr el siguiente comando para obtener la última versión estable:

```
svn checkout http://parallel-clusterer.googlecode.com/svn/branches/  
biopp parallel-clusterer
```

Notar que la descarga se efectúa desde el branch biopp ya que aún no se ha unificado esta versión al trunk.

## 4.3. Compilación

Para compilar la aplicación, primero se debe reemplazar el archivo “Makefile” original por el que se provee en el directorio Makefiles/Parallel-Clusterer

perteneciente al CD de éste proyecto. Luego se debe ejecutar el comando “make” para compilar la aplicación.

## 4.4. Ejecución

Para ejecutar la aplicación compilada con FuD-BOINC, primero se debe agregar la aplicación al proyecto BOINC siguiendo los pasos especificados en la sección 2.5.5. Luego, desde el directorio del proyecto, ejecutar el siguiente comando:

```
bin/clusterer -i [input\_file] -f compressed -s [output\_file] -a full\_cache.
```

Donde:

- **input\_file**, es el archivo que se provee en el directorio CD-Tesis/Parallel-clusterer/input\_files/.
- **output\_file**, es el nombre del archivo de salida donde la aplicación escribirá las estadísticas resultantes.