

# Mantener Paquetes **Debian** en equipo

Alejandro Ríos Peña

21/03/10

<http://www.debiancolombia.org>

[aleros@debian.org](mailto:aleros@debian.org)



# Contenido

1. Introducción
2. ¿Por qué mantener paquetes en equipo?
3. ¿Cómo entrar a un grupo?
4. El caso pkg-voip





¿Desea recordar conceptos de empaquetamiento?

Aceptar

Cancelar

# Conceptos de Empaquetamiento

- Resumen:
  - 1. Instalar/comprobar dependencias
  - 2. Bajar y descomprimir el programa a empaquetar
  - 3. Debianización inicial y edición de scripts.
  - 4. Construcción y prueba del paquete.

# Empaquetamiento (1)

- Paquetes necesarios:

*apt-get install build-essential lintian fakeroot  
debhelper dh-make dpkg-dev devscripts*



# Empaquetamiento (2)

- Descargar, descomprimir y explorar:

```
mkdir hello
```

```
cd hello
```

```
wget http://ftp.gnu.org/gnu/hello/hello-2.3.tar.gz
```

```
tar -xzvf hello-2.3.tar.gz
```

```
cd hello-2.3
```

# Empaquetamiento (3)

- Debianización inicial:

```
dh_make -s -c gpl -f ../hello-2.3.tar.gz
```

- Edición de scripts en `./debian/`:
  - **copyright**: Derechos de autor y licencia.
  - **control**: Dependencias y descripción.
  - **compat**: Versión del sistema de empaquetado.
  - **changelog**: Información sobre cada versión.
  - **rules**: Reglas para compilación e instalación.
  - **README.Debian**: Doc. específica de Debian.

# Empaquetamiento (4)

- Construcción:

```
dpkg-buildpackage -rfakeroot
```

- Verificación:

```
cd ..
```

```
lintian -i *.changes
```

```
lesspipe hello_2.3-1_i386.deb | less
```

# Empaquetamiento (5)

- Instalación y pruebas:

```
dpkg -i hello_2.3-1_i386.deb
```

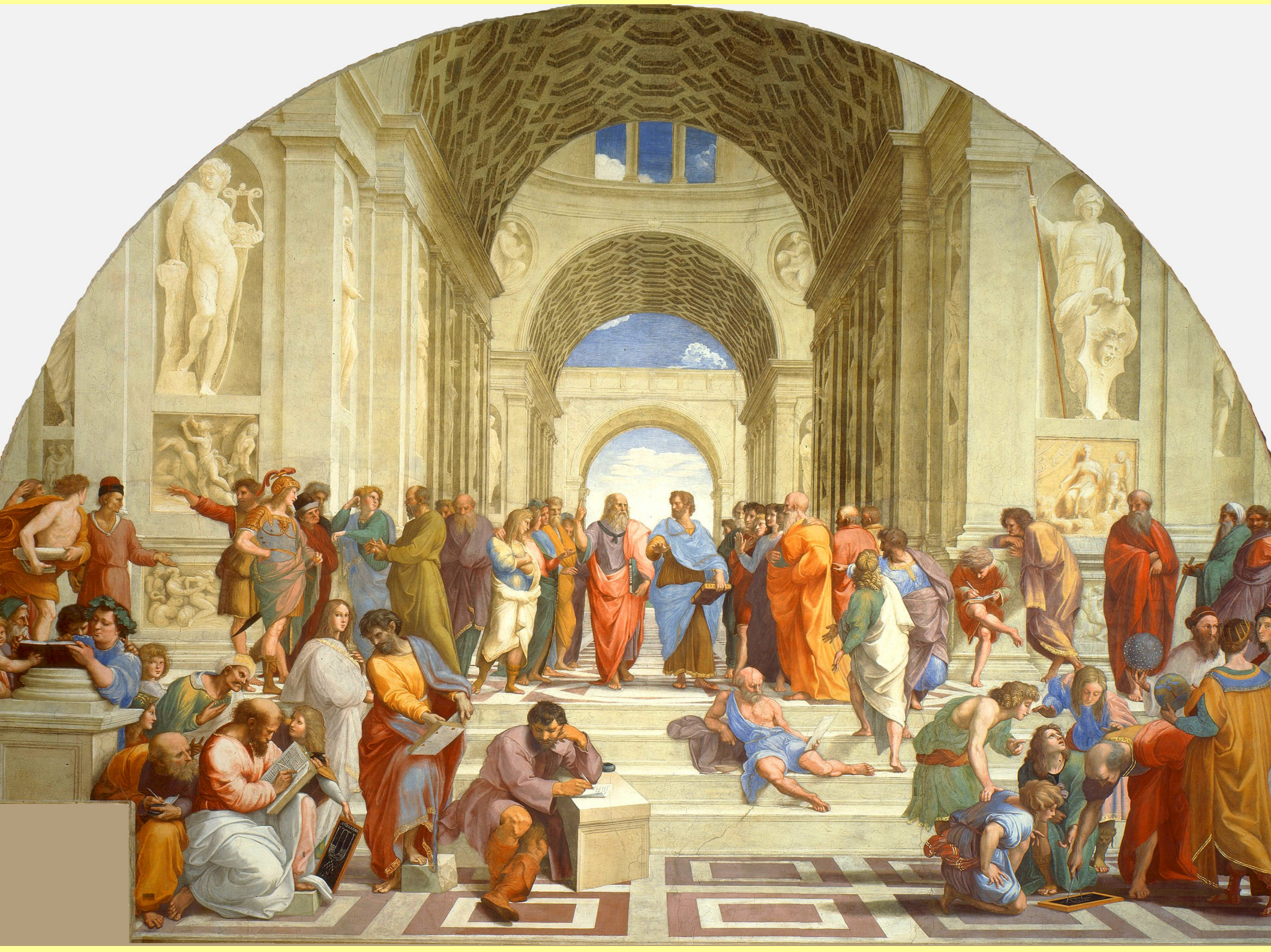
```
hello
```

# ¿Por qué mantener paquetes en equipo?

- Equipo != Grupo.
- Se unen fuerzas para empaquetar programas complejos y/o de interés común.
- Se pueden resolver fallos más rápidamente.

# ¿Por qué mantener paquetes en equipo?

- Los paquetes se apadrinan y patrocinan más fácilmente (sin mentores).
- Se pueden facilitar el ofrecer y tomar en adopción los paquetes
- Es mejor aprender de los equipos que solamente por sí mismo.



# ¿Cómo entrar a un equipo?

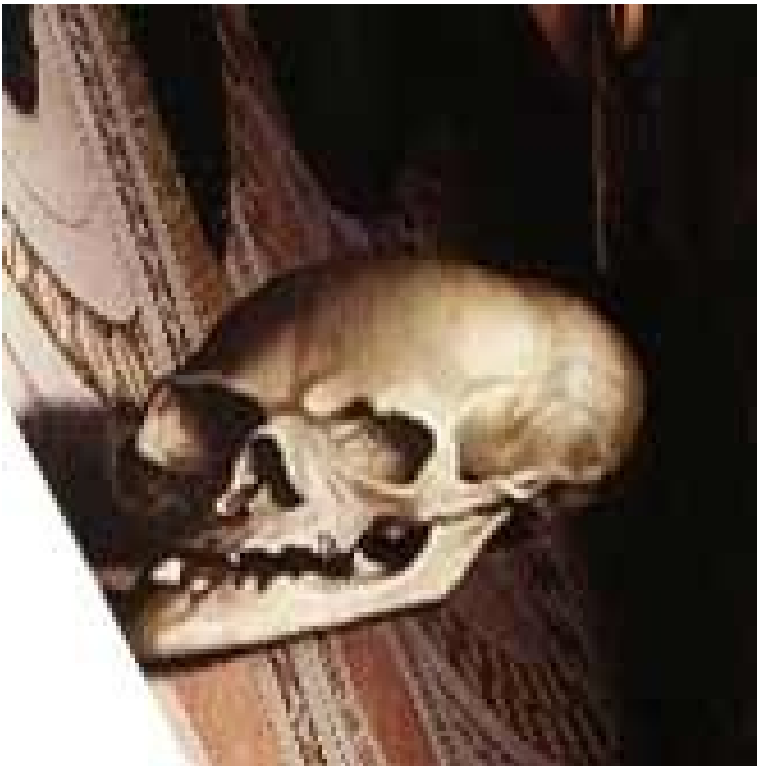
- Cada equipo tiene objetivos diferentes.
- Se debe buscar la página donde se indican los recursos que ofrece el equipo para empezar a trabajar (<http://wiki.debian.org/Teams>)
- Se empieza por explorar cómo funciona, luego hacer contribuciones indirectas, y por último solicitar el acceso de escritura.



# ¿Cómo entrar a un equipo?

- Líder
- Recursos comunes:
  - Lista de correo, canal de IRC.
  - Sistema de control de versiones (SVN, GIT)
  - Documentación de procedimientos y herramientas específicas.
  - Las mismas herramientas que se usan para mantenimiento individual





# pkg-voip

Dónde empezar:

- Web: <http://pkg-voip.alioth.debian.org/>
- SVN: <http://svn.debian.org/wsvn/pkg-voip/>
- Snapshots: <http://updates.xorcom.com/pkg-voip/>
- Doc: <http://svn.debian.org/wsvn/pkg-voip/README>
- Lista de correo:  
<http://lists.alioth.debian.org/mailman/listinfo/pkg-voip-maintainers>
- Pendientes:  
<http://pkg-voip.alioth.debian.org/cgi-bin/qareport.cgi>

# pkg-voip

- SVN:
  - nombre-paquete/
    - trunk/
    - tarballs/
    - build-area/
    - tags/
      - X.X-1
      - X.X-2
      - X.Y-1
    - Branches/
      - cambio\_drastico
      - bugXXXXX
      - mi\_prueba

# pkg-voip

- Modificaciones al paquete:
  - debian/control:
    - Maintainer: correo del equipo
    - Uploaders: correos de principales mantenedores
    - Vcs-Svn: url para usar el repositorio
    - Vcs-Browser: url para ver el repositorio
  - debian/rules:
    - Agregar método 'get-orig-source'

# pkg-voip

- Modificaciones al paquete:
  - debian/changelog:
    - Target: unreleased → unstable
    - [NOT RELEASED YET]
    - [Mantenedor 1]
      - \* Mis cambios
    - [Mantenedor 2]
      - \* Sus cambios

# pkg-voip

- svn-buildpackage:
  - Similar a dpkg-buildpackage, sobre svn.
  - Ej: svn-buildpackage -rfakeroot
  - Ofrece la opción de ignorar los errores de svn por cambios hechos o archivos nuevos
  - Ofrece la opción de publicación automatizada



# pkg-voip

- Publicando:
  - Cambiar changelog en trunk/  
(UNRELEASED → unstable)
  - Construir el paquete en trunk/
  - Verificar con lintian, pbuilder, elida, etc.
  - Copiar trunk/ a tags/versión
  - Hacer dch en trunk/
  - Poner UNRELEASED en trunk/
  - Subir todo actualizado al SVN.



# Gracias :)

Alejandro Ríos Peña

21/03/10

<http://www.debiancolombia.org>

[aleros@debian.org](mailto:aleros@debian.org)