



Software de Código Abierto y Libre: Mitos y Realidades.

Jesús M. Castagnetto, Ph.D.
<jmcastagnetto@php.net>

25 de Marzo de 2006
FLISOL 2006
Huancayo, Perú



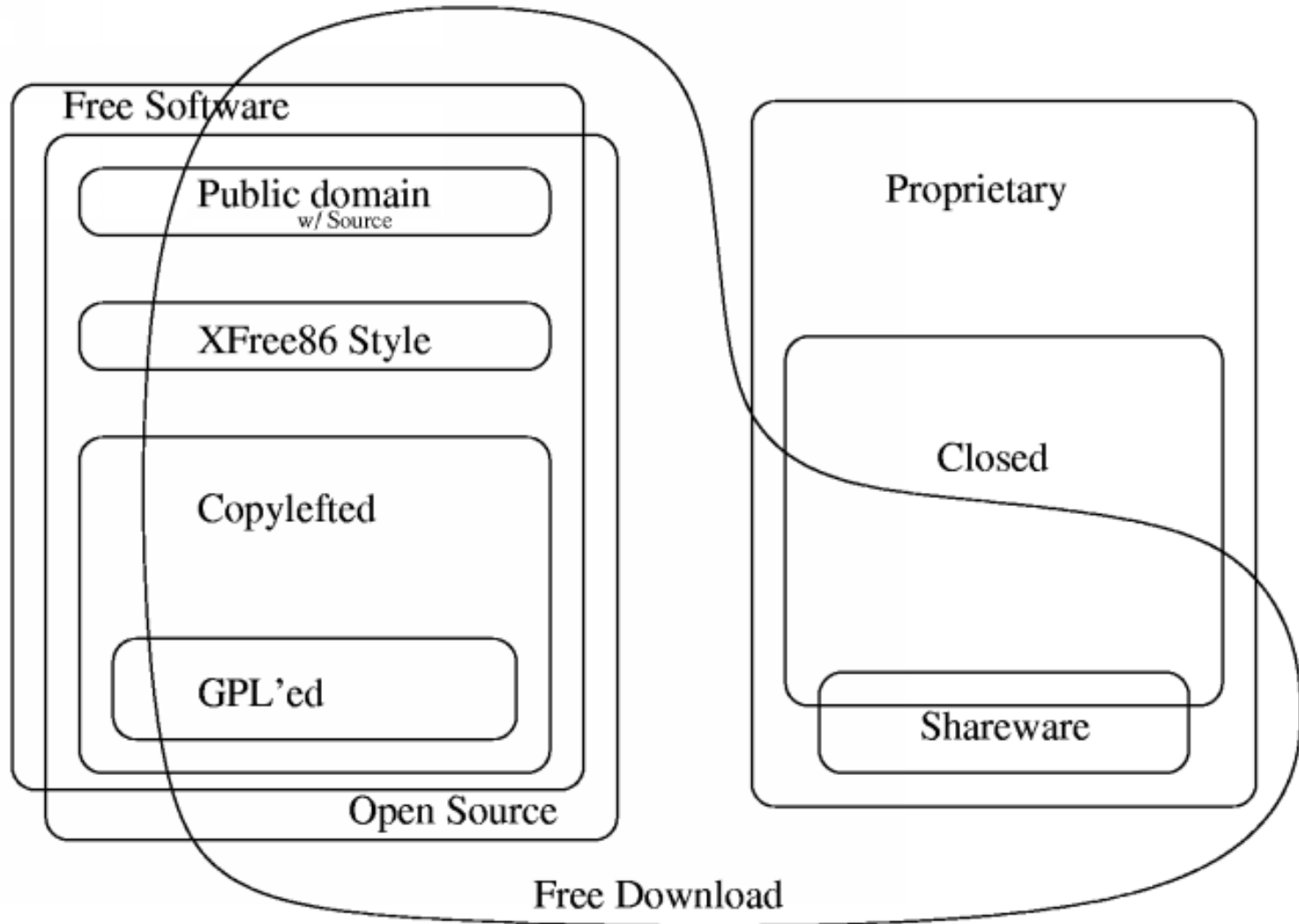
Agenda

- ¿Qué es Software de Código Abierto o Libre? (**Free/Open Source Software: FOSS**)
- Mitos acerca de FOSS
- Las comunidades que desarrollan FOSS
- Los motivos por los cuales FOSS se crea y desarrolla.
- Ejemplos de FOSS
- Donde conseguir más información.



Muchos tipos de software

<http://www.fsf.org/licensing/essays/categories.html>



FLISOL 2006



Categorías de software

Desde el punto de vista de los derechos y las opciones de uso que brinda la licencia de los aplicativos informáticos, estos se pueden clasificar en:

- **Privativo** (software cerrados, shareware, freeware, postcardware, etc.)
- **Dominio Público** (también llamado abandonware)
- **Free/Open Source Software (FOSS)**



Software privativo

- Puede o no tener un costo asociado a su uso (ej. freeware).
- Impide y desalienta la modificación y redistribución del programa.
- Limita donde y como puede ser instalado.
- Código no disponible o bajo una NDA, etc.
- Licencias restrictivas (y patentes).
- El usuario no tiene derecho a reclamo o arreglo si el programa no funciona.



Ej: EULA de MS Office

END USER LICENSE AGREEMENT FOR MICROSOFT SOFTWARE.

IMPORTANT-READ CAREFULLY: This Microsoft End-User License Agreement ("EULA") is a legal agreement between you (either an individual person or a single legal entity, who will be referred to in this EULA as "You") and Microsoft Corporation for the Microsoft software product that accompanies this EULA,

....

- o You are not licensed to do any of the following:

...

- o ***You may not create obscene or scandalous works, as defined by federal law at the time the work is created, using the Media Elements.***

...

Limitations on Reverse Engineering, Decompilation, and Disassembly. ***You may not reverse engineer, decompile, or disassemble the Software Product***

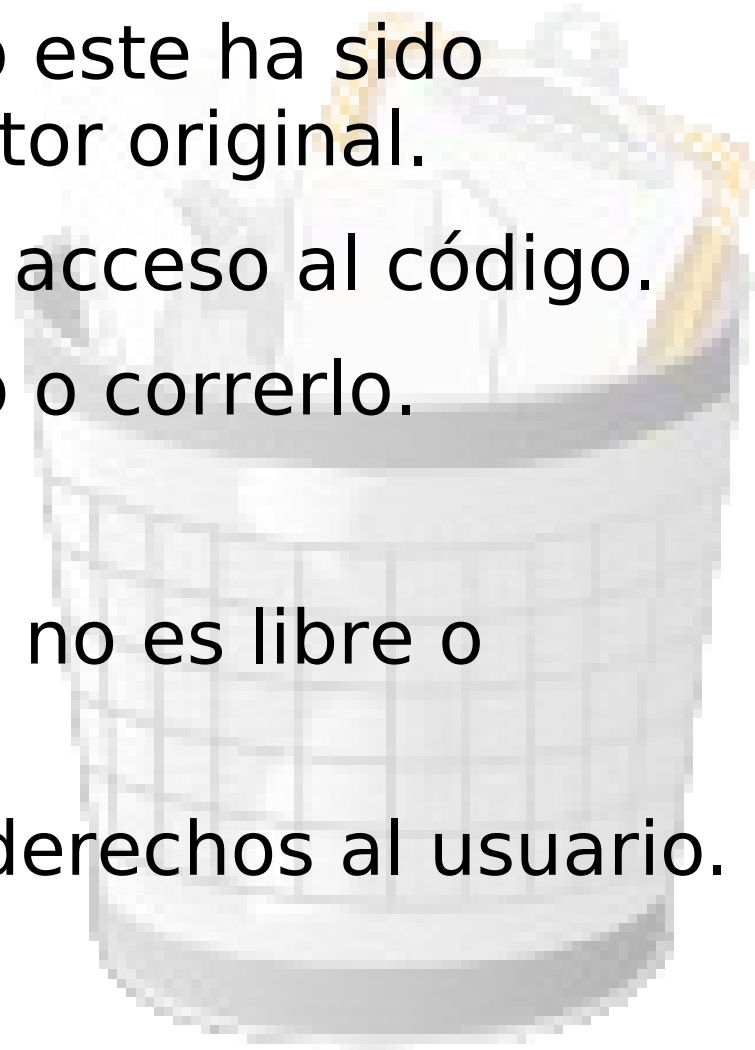
...

LIMITATION ON REMEDIES; NO CONSEQUENTIAL OR OTHER DAMAGES. Your exclusive remedy for any breach of this Limited Warranty is as set forth below. Except for any refund elected by Microsoft, ***YOU ARE NOT ENTITLED TO ANY DAMAGES, INCLUDING BUT NOT LIMITED TO CONSEQUENTIAL DAMAGES,*** if the Software Product does not meet Microsoft's Limited Warranty, and, to the maximum extent allowed by applicable law, even if any remedy fails of its essential purpose.



Software de dominio público

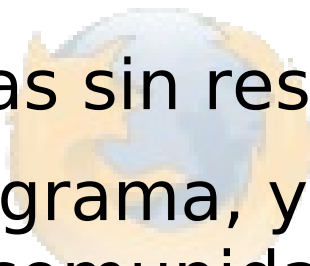
- No tienen copyright, o este ha sido abandonado por el autor original.
- Generalmente no hay acceso al código.
- No cuesta el instalarlo o correrlo.
- No existen licencias.
- Puede ser gratis, pero no es libre o abierto.
- No protege ni brinda derechos al usuario.





Características de FOSS

- El código fuente está siempre disponible.
- El usuario es libre de:
 - usar el programa para cualquier propósito
 - estudiar como el programa funciona, y adaptarlo a sus necesidades
 - distribuir copias sin restricciones
 - mejorar el programa, y de dar sus mejoras al resto de la comunidad
- Tiene copyright, licencias y atribuciones.
- Puede ser gratis o pagado.





Licencias de FOSS (1)

- Todas tienen como pre-condición la disponibilidad libre del código fuente.
- No restringen quién, donde o cómo se usa el software siendo licenciado.
- Difieren en la responsabilidad puesta en el usuario que modifica el código, en cuanto a su deber de re-distribución.
- Existen versiones para software y para otras obras (documentos, arte, etc.)



Licencias de FOSS (2)

- Para código
 - Free Software (Libre): GPL, LGPL.
 - Open Source Software (Código Abierto): Apache License, MPL, MIT License, PHP License, IBM Public License, BSD, etc.
- Para documentos: Open Publication License, Gnu Free Documentation License, Open Content License, etc.
- Otros: Design Science License, Creative Commons Licenses (varias).



Definición de Open Source (según la OSI)

- 1. Libre redistribución:** La licencia no debe restringir el que el software se de libremente o se venda. No debe requerir un pago de honorarios o regalía en caso de venta.
- 2. Disponibilidad de código fuente:** El programa debe de incluir el código fuente, y debe de permitir la distribución en forma compilada y de código fuente. Si el código no esta incluido, debe de existir una forma de obtenerlo libremente.



Definición de Open Source (según la OSI) [cont.]

- 3. Permitir trabajos derivados:** La licencia debe permitir modificaciones y trabajos derivados, los cuales puedan ser distribuidos bajo los mismos términos que el programa original.
- 4. Integridad del código fuente del autor:** La licencia puede requerir que las modificaciones se pongan como “parches” o que el trabajo derivado tenga nombre diferente, pero debe permitir que trabajos creados usando código modificado se distribuya libremente.



Definición de Open Source (según la OSI) [cont.]

5. **No puede discriminar su uso por ciertas personas o grupos.**
6. **No puede discriminar campos de estudio:** La licencia no puede restringir el uso sólo a ciertas áreas de estudio y desarrollo (ciencia, negocios, etc.)
7. **Distribución libre de la licencia:** Los derechos asociados al programa deben de aplicarse a todos aquellos a quienes el programa es redistribuido, sin necesitarse licencias adicionales.



Definición de Open Source (según la OSI) [cont.]

- 8. La licencia no debe ser específica a un producto:** La licencia no debe depender de si el programa es parte de una distribución en particular. Debe de poder aplicarse aún después de extraer el programa para uso separado.
- 9. La licencia no debe restringir a otros programas:** La licencia no debe poner restricciones en cuanto a otros programas incluidos en la distribución. No debe de insistir que todos usen el mismo tipo de licenciamiento.



Definición de Open Source (según la OSI) [cont.]

10. **La licencia debe ser neutra en cuanto a la tecnología:** No puede limitar el uso de una tecnología en particular o un estilo dado de interfaz programática o de usuario.

Tomado de “The Open Source Definition”
de la Open Source Initiative:

http://opensource.org/docs/definition_plain.php



Motivaciones (principios) para hacer Open Source

- Todo programa bueno comienza por que un desarrollador tenia algo que “*le picaba*” (un problema no resuelto).
- Los buenos programadores saben que escribir, los mejores saben que re-escribir (y lo mas importante, que reusar).
- “Planea el botar la primera versión; pues lo harás de todas maneras” (de Fred Brooks, *The Mythical Man Month*): Es preferible el rediseñar a tratar de parchar una solución que esta mal hecha o conceptualizada.



Motivaciones (principios) para hacer Open Source [cont.]

- Si tienes la actitud apropiada, problemas interesantes aparecerán en tu camino.
- Si pierdes el interés en un programa, tu último deber es el de buscar alguien competente para que tome las riendas.
- El tratar a tus usuarios como co-desarrolladores, es la ruta menos problemática para una mejora rápida del código y una búsqueda eficiente de defectos.



Motivaciones (principios) de Open Source [cont.]

- Saca versiones de tu programa lo mas pronto posible, con gran frecuencia, y escucha a tus usuarios.
- Si tienes una base de usuarios haciendo pruebas y de co-desarrolladores lo suficientemente grande, casi todos los defectos se podrán caracterizar rápidamente y su solución será obvia para alguien.
- Estructuras de datos inteligentes y código tonto funciona mejor que la situación inversa.



Motivaciones (principios) de Open Source [cont.]

- Si tratas a quienes prueban tu versión beta como tu recurso humano mas valioso, ellos responderán convirtiéndose en tu recurso humano mas valioso.
- La segunda cosa mejor a tener buenas ideas, es el saber reconocer buenas ideas de tus usuarios. Muchas veces eso es mejor.
- A menudo, las soluciones mas elegante e innovativas surgen de darte cuenta que tu concepto original del problema estaba equivocado.



Motivaciones (principios) de Open Source [cont.]

- “La perfección (en el diseño) se alcanza no cuando ya no hay mas que añadir, sino cuando ya no hay mas que quitar”
- Cualquier herramienta debe ser útil para la tarea que fue creada. Una herramienta verdaderamente útil puede ser usada en formas que no previstas.
- Cuando escribas una herramienta que conecte sistemas, trata de no cambiar el flujo de datos, y nunca botes información sin una razón muy importante.



Motivaciones (principios) de Open Source [cont.]

- Si creas un lenguaje que no es completo según Turing, hacerlo sintácticamente sencillo ayuda mucho al usuario.
- Un sistema seguro es sólo tan seguro como su secreto. Ten cuidado con pseudo-secretos.
- Para resolver un problema interesante, empieza encontrando un problema que a ti te parezca interesante.
- Si el coordinador tiene un buen medio de comunicación, y sabe guiar sin coerción, muchas cabezas son mejor que una.



Mitos acerca del Código Libre/Abierto

(Mito 1) “El único atractivo es el costo”

(Mito 2) “Open Source no es confiable o no existe soporte para este”

(Mito 3) “Ninguna compañía grande usa Open Source”

(Mito 4) “Open Source es hostil a la propiedad intelectual”

(Mito 5) “No se puede hacer dinero con Open Source”





Mitos acerca del Código Libre/Abierto (cont.)

- (Mito 6) “Open Source es sólo la licencia”
- (Mito 7) “Si pongo mi código en Open Source, magicamente habrá miles de programadores que trabajaran por nada”
- (Mito 8) “Open Source es sólo de interés para los programadores”
- (Mito 9) “Open Source no es sostenible como modelo de desarrollo”
- (Mito 10) “Open Source esta tratando de alcanzar a las compañías comerciales”



Mito 1: Costo

- En general, el costo del software es mínimo o no existente.
- Pero, el poder hacer lo que se necesite eficientemente es más importante.
- Para muchas compañías, universidades, gobiernos, etc., el costo no es uno de los determinantes, pero si la confiabilidad y robustez del sistema.
- Siempre hay un costo directo o indirecto: ej. entrenamiento del personal.



Mito 2: Confiabilidad

- La columna vertebral de la Internet corre en Free/Open Source Software: BIND (DNS), Sendmail (SMTP), TCP/IP, etc.
- Compañías e instituciones que dependen de la performance usan Open Source: Yahoo!, Google, Akamai, NASA, etc.
- Estadísticas de seguridad de Open Source son mejores que productos privativos, ej. OpenBSD: 1 “agujero” remoto en más de 8 años (comparar con MS Windows :-)



Mito 3: No es usado por “compañías grandes”

- Google
 - Yahoo!
 - Amazon.com
 - British Petroleum
 - Chicago Mercantile Exchange
 - UPS
 - Walt Disney
 - IBM
 - Sony
 - E-Trade
 - Sabre Holdings
 - Lockheed Martin
 - Pfizer
 - Merrill Lynch
 - Motorola
- y otras compañías tan o más “pequeñitas”...



Mito 4: Hostil/destructor de la propiedad intelectual (IP)

- FUD^[1] perpetrado por Jim Allchin de Microsoft (otros lo han repetido).
- Las licencias de FOSS existen para proteger los derechos de todos, no para mermarlos o destruirlos.
- IBM y Apple usan FOSS, y tienen muchas patentes y IP en su portafolio.
- Algunas compañías usan FOSS y no lo dicen (ej. librerías de conexión a internet, BSD TCP/IP stack, en MS Windows)

[1] FUD: Fear, Uncertainty, and Doubt – Miedo, Incertidumbre y Duda.



Mito 5: No se puede ganar dinero con Open Source

- Entonces las ganancias de Google, Amazon.com, y Yahoo! deben de ser imaginarias y pura fantasía. ; -)
- IBM también esta ganando mucho dinero y no debería, según este mito.
- El modelo ha cambiado: lo esencial son servicios, desde instalación, personalización y configuración, hasta soporte, monitoreo y entrenamiento.
- El antiguo modelo de negocios es caduco.



Mito 6: FOSS = licencia

- La licencia es una parte, y no la más crítica en la vida de un proyecto FOSS.
- Lo importante es el acceso comunitario al código, y la evolución del mismo en respuesta a las necesidades del público.
- La dinámica de interacción y desarrollo es otro factor de importancia. Las comunidades son **meritocracias**.
- Desarrollo distribuido es la norma, no la excepción (al revés del modelo privativo).



Mito 7: Trabajadores gratis

- Si realmente crees eso, tengo un puente baratito que venderte en New York...
- Netscape aprendió esto de la forma más dura con Mozilla. No existe una mágica comunidad de “desarrolladores de Open Source” listos al instante.
- Si quieres desarrolladores, cultiva una comunidad de gente interesada. Esto requiere esfuerzo y recursos (tiempo, dinero y ganas): TANSTAAFL





Mito 8: Open Source es sólo para programadores

- En un mercado libre, los consumidores se benefician por la mejora continua de los productos: Open Source proporciona uno de los mercados de ideas y software más libres, donde todos pueden competir.
- Esto trae: mejores precios, crecimiento en la innovación, especialización, más variedad en las ofertas, etc.
- FOSS te da el poder de tener un efecto directo en la evolución de un sistema. El usuario es un ente activo, no pasivo.



Mito 9: El Modelo de desarrollo no es sostenible

- Los proyectos evolucionan, y a la vez sufren competencia en el mercado.
- Sólo los más útiles y que tengan respaldo comunitario sobreviven.
- La gente contribuye en la medida que el producto hace su trabajo más fácil.
- Los contribuidores son una mezcla de gente en academia, en compañías, profesionales, y amateurs. Todos proveen algo a la economía del proyecto.



Mito 10: FOSS pretende alcanzar al software privativo

- Históricamente ha sido al revés, ej.: MS Windows empezó a ser multiusuario en los 90's, BSD Unix lo era desde los 70's.
- Los mejores sistemas hoy en día son FOSS: El servidor más usado en la Web (según Netcraft) es Apache (70%), mientras MS IIS es segundo (20%).
- Nuevos tipos de software: Wiki(pedia), Blogs, RSS, Semantic Web, etc.
- Seguridad, ej.: NSA tiene SELinux.





... y algunos otros mitos

- El desarrollo de Open Source es algo reciente. Ya en los 1960s se compartía código (ARPANet). Linux empezó en 1991.
- Open Source es escrito por gente inexperta y como pasatiempo. Entidades como IBM, Google, NASA, etc. hacen Open Source.
- El software de Open Source no tiene interfaces y formatos estables. En general los proyectos de FOSS siguen estándares internacionales en formatos, protocolos, etc. Lo contrario de soluciones privadas que hacen lo que quieren perjudicando al usuario.



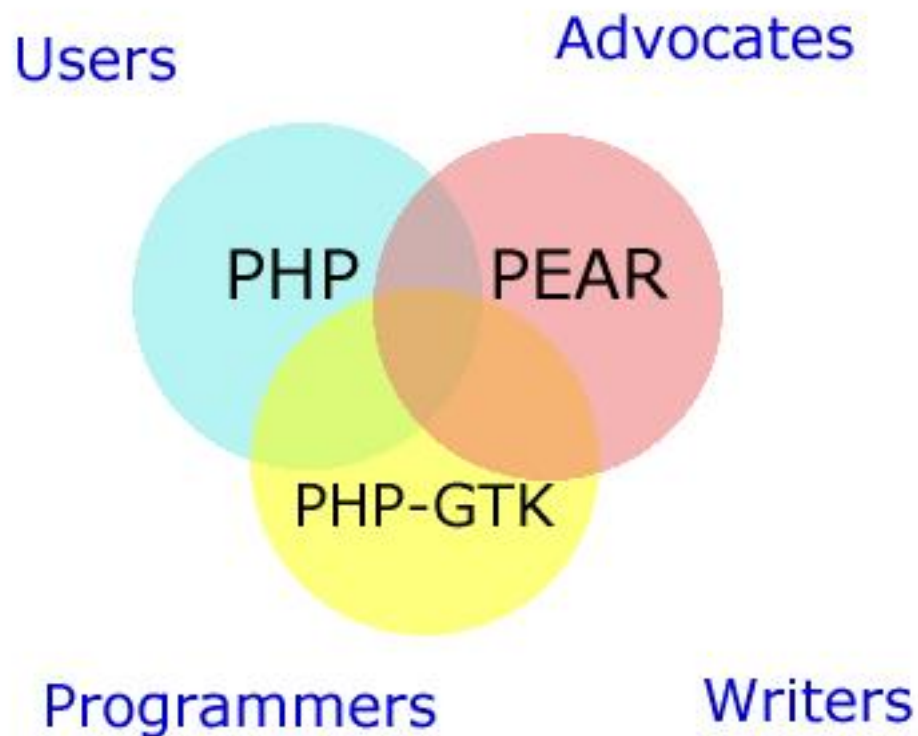
Ecosistemas en FOSS

- FOSS: Muchas comunidades cada una se beneficia de la producción de las otras.
- Tipos de comunidades:
 - Desarrolladores
 - Divulgadores
 - Usuarios





Comunidades FOSS: PHP

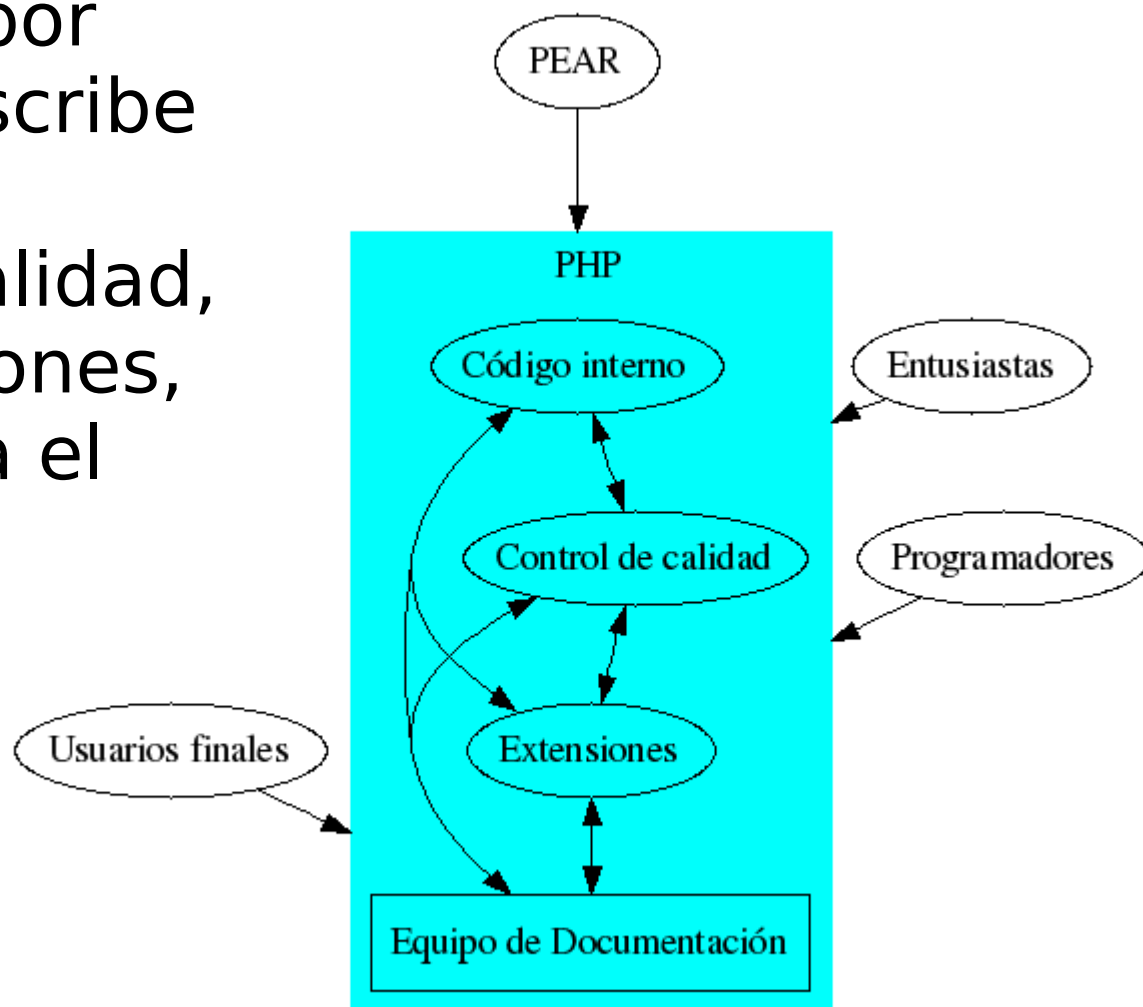


- Comunidad de comunidades
- Cada una compuesta de varios groups, interactuando entre si y superpuestos.



La comunidad de PHP

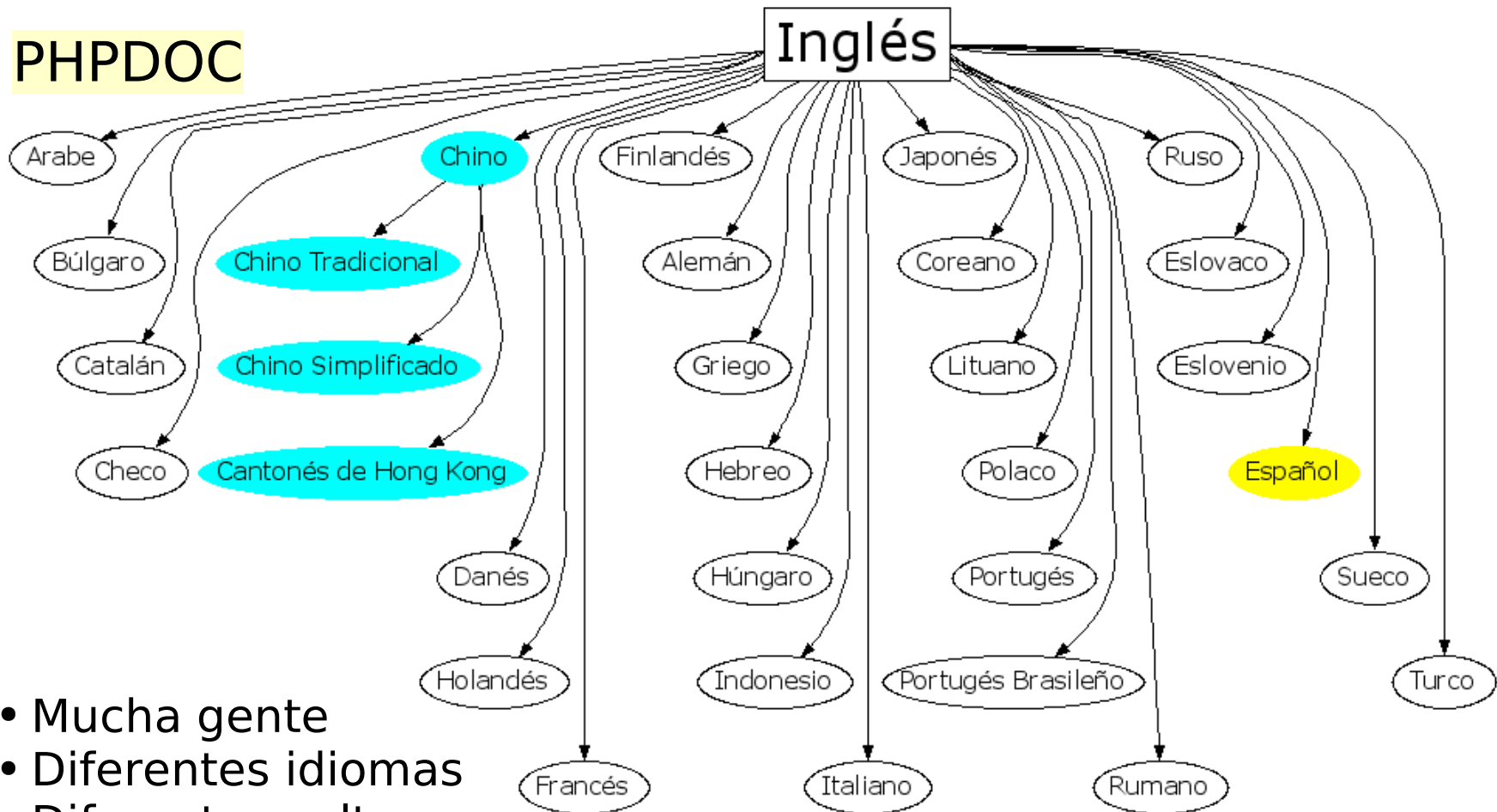
- Compuesta por gente que escribe código, hace control de calidad, crea extensiones, y documenta el lenguaje.





Grupo de documentación

PHPDOC



- Mucha gente
- Diferentes idiomas
- Diferentes culturas
- Diferentes costumbres
- Un objetivo común

2006/03/25

Código Libre - Jesús M. Castagnetto

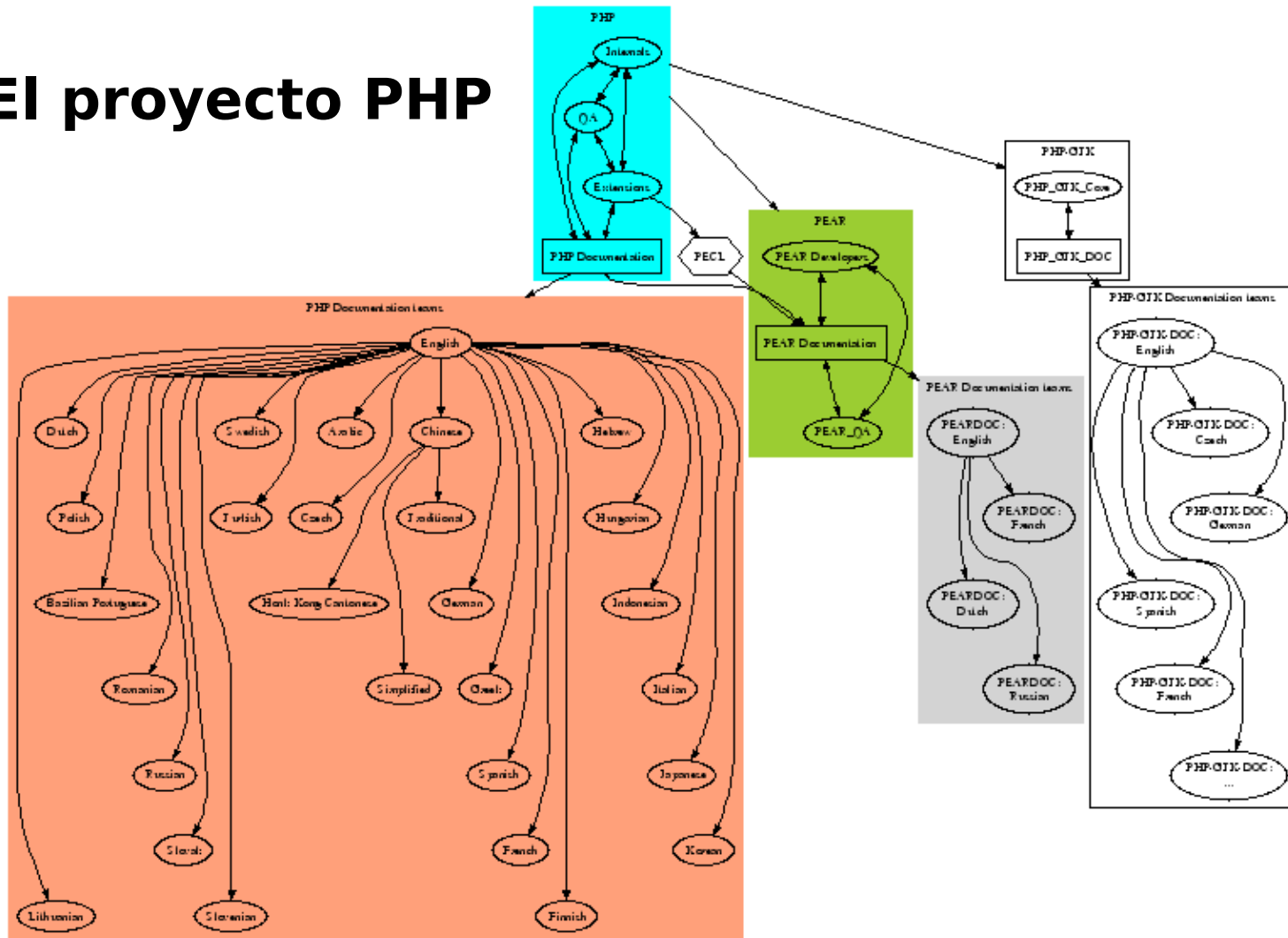
38



Una comunidad de comunidades

FLISOL 2006

El proyecto PHP



2006/03/25

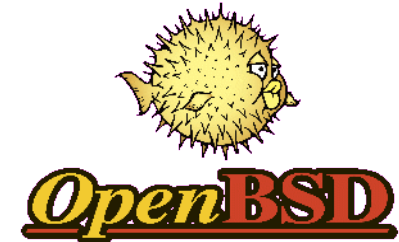
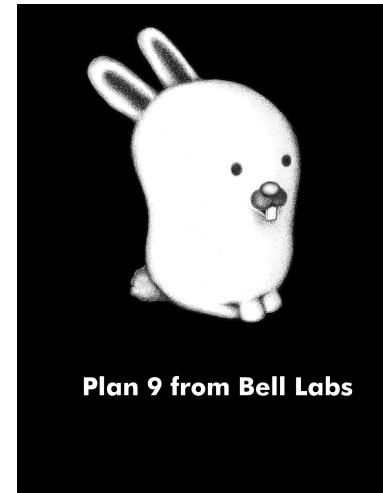
Código Libre - Jesús M. Castagnetto

39



Ejemplos de FOSS: Sistemas Operativos

FLISOL 2006





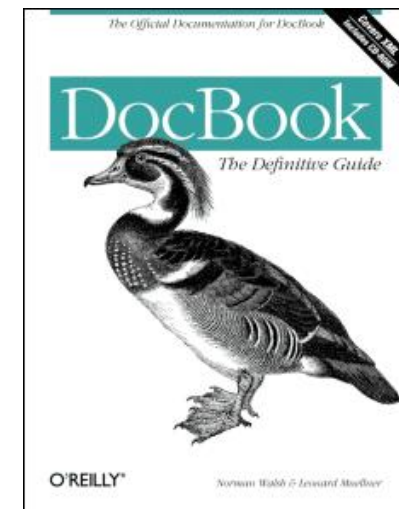
Ejemplos de FOSS: Aplicativos de Servidor

FLISOL 2006





Ejemplos de FOSS: Lenguajes y Librerías



FLISOL 2006



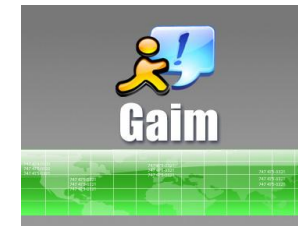
Ejemplos de FOSS: Aplicativos

FLISOL 2006



WIKIPEDIA

English <i>The Free Encyclopedia</i> 1 037 000+ articles	Deutsch <i>Die freie Enzyklopädie</i> 373 000+ Artikel
Français <i>L'encyclopédie libre</i> 257 000+ articles	Polski <i>Wolna Encyklopedia</i> 222 000+ haseł
日本語 フリー百科事典 194 000+ 記事	Nederlands <i>De vrije encyclopedie</i> 152 000+ artikelen
Svenska <i>Den fria encyklopedin</i> 145 000+ artiklar	Italiano <i>L'enciclopedia libera</i> 145 000+ articoli
Português <i>A enciclopédia livre</i> 122 000+ artigos	Español <i>La enciclopedia libre</i> 103 000+ artículos





Para mas información

- Free Software Foundation: <http://www.fsf.org/>
- Open Source Initiative:
<http://www.opensource.org/>
- Creative Commons: <http://creativecommons.org/>
- Algunos artículos por Eric S. Raymond:
 - The Cathedral and the Bazaar
<http://www.catb.org/~esr/writings/cathedral-bazaar/>
 - Open Minds, Open Sources
<http://www.catb.org/~esr/writings/analog.html>
- Freshmeat: <http://www.freshmeat.net/>
- SourceForge: <http://www.sourceforge.net/>



Gracias

Esta y otras presentaciones en:

<http://www.castagnetto.org/>

(Sección de “Talks/Charlas”)

E-mail: jmcastagnetto@php.net