

جلسه سوم گنو/ لینوکس

- بخش اول (مقالات گنو) :

Patent ها یک چرند اقتصادی هستند

- بخش دوم (گنو/ لینوکس) :

آشنایی با سیستم بسته بندی APT

Patent چیست؟

- پتنت حق انحصاری است که در قبال اختراع ثبت شده به مخترع یا نماینده قانونی او اعطا (Grant) می شود. اختراع (Invention) محصول یا فرآیندی است که راه جدید انجام کاری را ارائه می دهد یا راه حل فنی جدیدی را برای مشکل خاصی پیشنهاد می کند.
- پتنت حمایت قانونی از اختراع را برای دارنده آن تامین می کند. این حمایت برای مدت محدودی (حداکثر ۲۰ سال) در ازای افشای جزئیات اختراع برای عموم، اعطا می گردد.

حمایت Patent چگونه است؟

- مورد اختراع را نمی توان بدون اجازه دارنده پتنت به صورت تجاری تولید کرد، استفاده یا توزیع نمود و به فروش رساند.
- پتنت حقی برای تولید محصول به مخترع نمی دهد بلکه فقط دیگران را از تولید آن باز می دارد (!).
- دادگاه در بیشتر نظام ها، شخص یا اشخاص نقض کننده حقوق پتنت را از این کار بازداشته و وی را مکلف به جبران خسارتهای وارده به صاحب حق اختراع می نماید.

اختیارات دارنده‌ی Patent

- دارنده پتنت حق دارد تصمیم بگیرد چه کسی می‌تواند در طول مدت حمایت، از اختراع وی بهره‌برداری نماید.
- وی میتواند با عقد قرارداد اجازه یا لیسانس استفاده از اختراع را به فرد دیگر واگذار کند.
- دارنده پتنت همچنین می‌تواند حق اختراع خود را بطور کامل به دیگری بفروشد که خریدار دارنده جدید آن اختراع شناخته خواهد شد.
- پس از پایان دوره حمایت (حداکثر ۲۰ سال) مجوز آن در اختیار همه‌ی افراد خواهد بود.

چه چیزهایی قابلیت پتنت شدن را دارند؟

- اختراعات پتنت شده در واقع در تمام شئونات زیست بشر سایه افکنده است، از روشنایی برق (پتنت شده توسط ادیسون) و پلاستیک (پتنت شده توسط Baekeland)، تا قلم دارای نوک تویی (پتنت شده توسط Biro) و ریز پردازنده ها (پتنت شده توسط شرکت اینتل).
- نکته‌ی بسیار مهم برای برنامه‌نویسان این است که الگوریتم‌ها نیز قابلیت پتنت شدن را دارند.
- برای مثال فرمت‌های صوتی و تصویری MPEG پتنت شده هستند.

چه نوع اختراعی را می‌توان پتنت کرد؟

- اختراعی دارای قابلیت پتنت شدن است که دارای ویژگی‌های زیر باشد:
- نو بودن (Novelty) : مشخصه‌ی جدیدی که در آن رشته پیش‌بینی نشده باشد
- گام ابتکاری (Inventive Step) : ارتقای قابل توجه را در حیطه‌ی خود موجب شود
- استفاده عملی (Practical Use) : اختراع نباید صرفاً تئوریک باشد
- دارا بودن قابلیت ثبت (Patent Ability) : بعضی از اختراعات بسته به قانون آن کشور قابلیت پتنت کردن ندارند.

Patent ها یک چرند اقتصادی هستند

- پتنت‌ها یک امتیاز ویژه هستند نه دارایی. درواقع، حتی توسعه یافته ترین قوانین پتنت تایید میکنند که آزادی باید عاقبت (پس از چند سال) برای استفاده‌ی آزادانه از فناوری پتنت شده به عموم بازگردانده شود.
- دارایی قانونی برای همیشه پایدار خواهد بود، درحالی‌که پتنت‌ها برای چند سال اختیاری که برحسب کشور، تاریخ، و هوس قانونگذار (و نفوذ درحال رشد لابی‌های پتنت بر حکومتها) تغییر میکنند دوام می‌آورند.
- پتنت‌ها یک امتیاز هستند، هزینه‌ی آن بوسیله‌ی عموم پرداخته میشود.

اثرات مرتبه اول پتنت‌ها : بازدهی کم انحصار

- این واقعیت را باید بپذیریم که اثر انحصار بر روی اقتصاد منفی است.
- انحصار تنها یک فایده آن هم برای انحصارگر دارد : دیده شدن . اما این فایده با فشار بر روی مشتریان و رقیبان بدست آمده است.
- تنها راهی که یک انحصار میتواند کسی را ثروتمند کند بوسیله‌ی بازداری کس دیگری از در اختیار گذاشتن یک سرویس رقابتی است.
- تنها دلیلی که یک مشتری به یک رقیب مراجعه می‌کند قیمت پایین‌تر محصول با کیفیت یکسان (و گاهی بهتر) است. ولی پتنت این رقابت را از جامعه‌ی ما می‌گیرد.

اثرات مرتبه دوم پتنت‌ها : بازدارنده و بدانگیزنده

- هر انگیزه‌ی موفقیت‌آمیز برای ابداع کردن که پتنت‌ها می‌توانند ایجاد کنند منجر به یک انحصار جدید می‌شوند که در آن هرکس برای ابداع کردن چیزی بیشتر بی‌انگیزه می‌شود، چراکه رقابت تخفیف یافته است.
- انگیزه‌ی پتنت‌ها انگیزه‌ای برای پیدا کردن راه‌هایی برای بدست آوردن پول با تخریب کردن اقتصاد عمومی است.
- بدترین ابداعات هستند که بوسیله‌ی سیستم‌های پتنت بیش از همه انگیزه‌دهی میشوند: آنهایی که پتنت کردن آنها بیشترین زیان را به عموم وارد میکند.

اثرات مرتبه سوم پتنت‌ها : بهم ریختگی

- پولی که از پتنت‌کردن بدست می‌آید به احتمال قریب به یقین هیچ کمکی به توسعه‌ی آن محصول نخواهد کرد (چرا که در راه شخصی و گذران زندگی فرد مخترع ثبت خواهد شد)
- همه‌ی دانش، ریسک‌ها، و بقیه‌ی عوامل درگیر در خلاق محصولات مشتق اکنون عمدتاً بر شانه‌های دارنده‌ی انحصار قرار گرفته‌اند، درحالی‌که مردم دیگر در تناسب با هزینه‌ی مجوزی که دارنده می‌تواند درخواست کند دلسرد شده‌اند.
- همچنین لابی‌های قدرت همواره در تلاش برای تهدید رقیبان جهت عدم دسترسی به فن اختراع هستند چرا که پتنت را نقض می‌کند.

اثرات ویژه ی پتنت های نرم افزاری : آشغال !

- نرم افزار مهم ترین نقش را برای رایانه های امروزی که اکثرا به هم متصل هستند بازی می کند. (مانند زبان برای انسان)
- یک برنامه نویس خوب ، خوب کد می زند ولی یک برنامه نویس حرفه ای از کدهای نوشته شده ی دیگران به خوبی استفاده می کند.
- بنابراین اصل ما برنامه نویسان حرفه ای خود را از دست خواهیم داد.
- مهندسان نرم افزار بطور مداوم روشی را که آنها باید با پروتکل های انحصاری که بخوبی مستند نشده اند (اگر اصلا شده باشند) مطابقت کنند نفرین میکنند.

اثرات ویژه‌ی پتنت‌های نرم‌افزاری : آشغال ! (ادامه)

- بنابراین حیظه‌ی توسعه‌ی رایانه با آشغال پر شده است، که با مرور زمان انباشته می‌شود، و کسی حق تمیز کردن ندارد، وگرنه او با دیگران ناسازگار می‌شود.
- هر پتنت روی یک نرم افزار یا فن موفق یک سد راه برای تمامی صنعت است که تا زمانی که به انقضا برسد باقی میماند.
- حتی دارنده، وقتی که می‌خواهد فن قبلی خود را پیشرفت دهد، خویش را با اینرسی یک صنعت کامل که با آشغال خود او تطابق یافته است مواجه می‌یابد.
- تغییر شکل یافتن برای ارتباط یافتن با آن، یا دور زدن آن.

پیامدهای فنی پتنت‌ها

- تاخیر در انتشار :

واقعا انتشار یک الگوریتم نوآورانه! در ۲۰ سال دیگر به چه درد ما خواهد خورد؟!!

- قفل‌شدگی و اجتناب از تحقیق

- توسعه‌های اضافی (وجود نسخه‌های پیاده‌سازی شده‌ی ناکارآمد)

- اینرسی فنی و فنون مرده (مخارج یک اختراع بیشتر از منافع آن)

- تحقیق منفی (پتنت کردن فنونی که توسط می‌توانند استفاده شوند)

- مغایرت میان پتنت‌ها و تحقیق عمومی (پتنت‌کردن پایان نامه!!!)

بخش دوم : سیستم بسته بندی APT

- برای نصب بسته های نرم افزاری در دبیان راه های مختلفی وجود دارد :
- نصب بسته دبیان (deb)
- کامپایل کردن سورس کد (معمولاً با پسوند tar.gz)
- تبدیل از سایر فرمت های بسته بندی به deb (مثل rpm به deb)
- استفاده از یک برنامه ی کامپایل شده ی عمومی (مثل Firefox)

APT (Advanced Packaging Tool)

سیستم‌های بسته‌بندی در توزیع‌های مختلف

- هرکدام از توزیع‌های مختلف برای خود یک نوع بسته‌بندی دارند. که هر کدام نقاط قوت و ضعفی دارند. اما توزیع دبیان که از APT استفاده می‌کند در حال حاضر قوی‌ترین سیستم بسته‌بندی را داراست.
- تعدادی از بسته‌بندی توزیع‌های مشهور:
- RPM برای ردهت و توزیع‌های مبتنی بر آن
- DEB برای دبیان و توزیع‌های مبتنی بر آن
- YMP برای OpenSuse

ابزارهای مدیریت بسته‌های دبیان

- سه ابزار متنی بسیار قوی برای کار با بسته‌های دبیان در ابونتو وجود دارند که عبارتند از:
 - apt-get پیشرفته ترین ابزار مدیریت بسته‌ها
 - dpkg ابزاری برای مدیریت بسته‌ها نه به قدرت اولی
 - dpkg-deb ابزاری دیگر برای کار با بسته‌ها
- برای محیط گرافیکی مدیریت بسته‌بندی در KDE از Kpackagekit یا همان Software Management در System Settings استفاده می‌کنیم.

سیکل حیاتی یک بسته

- بر روی هر بسته‌ی نرم‌افزاری اعمالی انجام می‌شود که به صورت :
 - دانلود شده (Downloaded)
 - باز شده (Unpacked)
 - کامپایل شده (Compiled)
 - پیکربندی شده (Configured)
 - نصب شده (Installed)
 - برداشته شده (Deinstalled) و حذف شده (Purged)

اطلاعات درون بسته‌ها : محتویات بسته‌ها

- برنامه‌های نرم‌افزاری
- فایل conffiles (لیست فایل‌های پیکربندی)
- اسکریپت‌های نصب شامل :
 - preinst – postinst قبل و بعد از نصب بسته
 - permm – postrm قبل و بعد از حذف بسته
- فایل‌های man و info (راهنمای برنامه)
- فایل کنترل (شامل نسخه‌ی برنامه ، کتابخانه‌های مورد نیاز و ...)

نام بسته‌های نرم‌افزاری

- `program_version-revision_processor.deb`

- نام بسته

- شماره نسخه

- شماره اصلاحیه

- نوع پردازنده (i386 برای اینتل و ppc برای Ibm ppc)

- پسوند deb که نشانگر یک بسته‌ی دبیان است.

اولویت بسته‌های نرم‌افزاری

- مورد نیاز (Requires) : بسته یکی از اجزای پایه OS است.
- مهم (Important) : نیاز حیاتی وجود ندارد ولی برای تکمیل نیاز است
- استاندارد (Standard) : وجود آن سودمند است.
- انتخابی (Optional) : بیشتر بسته‌ها از این نوع هستند
- اضافی (Extra) : نیازی به وجود این بسته‌ها نیست.

در صورت عدم وجود بسته‌های Requires سیستم با مشکل مواجه خواهد شد

وابستگی‌های بسته‌ها

- Depends بسته به بسته یا بسته‌های دیگر برای عملکرد صحیح نیاز دارد
- Recommends بسته با یک بسته‌ی دیگر بهتر کار خواهد کرد
- Suggests در بیشتر موارد با این بسته‌ها بهتر کار خواهد کرد
- Conflicts بسته با یک بسته‌ی دیگر کار نخواهد کرد
- Replaces بسته هنگام نصب بسته‌هایی را جایگزین خواهد کرد
- Provides حاوی بسته‌ی دیگری است که همان کارکرد را دارد
- Predepends بسته مورد نیاز باید قبل از این بسته نصب شود

وضعیت بسته‌ها (Package Statuses)

- Config-Files تنها فایل‌های پیکربندی نصب شده‌اند
- Half-Configured پیکربندی ناقص مانده است
- Half-Installed نصب بسته ناقص مانده است
- Installed بسته با موفقیت باز شده و پیکربندی و نصب شده است
- Not-Installed هیچ بخشی از بسته نصب نشده است
- Unpacked بسته باز شده ولی پیکربندی نشده است

حالت‌های انتخاب (Selection States)

- Install بسته برای نصب و پیکربندی انتخاب شده
- Deinstall بجز فایل‌های پیکربندی همه‌ی بسته برای حذف شدن
- Purge تمامی فایل‌های یک بسته
- Hold بر روی نسخه جاری بسته کاری انجام نمی‌گیرد
- Unknown بسته هنوز باز نشده است

پرچم‌های بسته‌ها

- برای هر بسته APT دو پرچم را در صورت نیاز ست می‌کند که عبارتند از:
- Hold هیچ کاری بسته انجام نخواهد شد. در ارتقای بسته‌ها، بسته‌های Hold ارتقا داده نمی‌شوند
- Reinst-required بسته شکسته شده و نیاز به نصب مجدد دارد. امکان حذف بسته تنها با گزینه‌ی `force-reinstreq` وجود دارد

اطلاعات بسته‌ها را از طریق `dpkg -s package_name` بگیرید

فایل‌های مدیریت بسته‌های نرم‌افزاری

- `/etc/apt/sources.list`
- `/etc/apt/apt.conf`
- `/var/lib/dpkg/available`
- `/var/lib/dpkg/status`

نصب و یا حذف بسته‌ها با استفاده از apt-get

- برای نصب بسته‌ها ابتدا مخزن (Repository) آن را به `sources.list` اضافه می‌کنیم ، سپس دستور `sudo apt-get update` را می‌زنیم و با دستور زیر بسته‌ی مورد نظر را نصب می‌کنیم :
- `sudo apt-get install package_name`
- برای حذف بسته‌ها
- `sudo apt-get remove package_name`

برای دریافت این اسلاید و دیگر منابع

- به صفحه‌ی اختصاصی این کلاس در وبلاگ بنده مراجعه کنید :

<http://blog.itnotes.info/linux>

