Software como Producto
vs.
Software como Servicio

Ciclo de conferencias del CeCom

Mg. A. G. Stankevicius

Primer Cuatrimestre

2006

Copyright

Objetivos de la charla

- Familiarizarnos con los distintos modelos de construcción de software:
  - Software como producto.
  - Software como servicio.
  - Software como producto y servicio a la vez.
- Analizar desde nuestra perspectiva tanto sus ventajas como desventajas.
- El determinar qué modelo resulta más conveniente o redituable quedará en manos de los asistentes.

Dos modelos... ¿antagónicos?

- ¿Qué producto vende la compañía de cable, o mejor aún Wal-Mart?
- ¿Qué servicio vende un fabricante de tuercas, o un productor de soja?
- En principio, parecen no tener nada en común.
- Pero, ¿el software, en qué categoría debería estar?
  - ¡puede ser clasificado en ambas!

Características de los productos

- La fabricación de productos tiene ciertas características en común:
  - Se requiere contar con un capital previo.
  - Primero se diseña el producto.
  - Luego se replica infinidad de veces.
  - Más tarde se reparte a través de alguna cadena de distribución.
  - Finalmente, el cliente paga por recibir una copia del producto.
  - El producto suele ser una “cosa” tangible.

Características de los servicios

- El proveer servicios también presenta ciertas características en común:
  - No se requiere contar con un capital previo.
  - Se debe encontrar una necesidad insatisfecha que tengan los clientes.
  - No se requiere una cadena de distribución.
  - El cliente paga por tener su necesidad debidamente atendida.
  - El servicio no suele ser una “cosa” tangible.

El software como producto

- Muchas compañias de software exitosas adoptan este modelo:
  - Microsoft.
  - Adobe.
  - Muchas otras!
- Recordemos que el software en sus inicios venía gratis con el hardware.
- Recién a fines de los 70' principios de los 80' se “inventó” esta nueva forma de concebir al software.

El software como producto

- El software atraviesa las etapas propias de la fabricación de productos:
  - Es diseñado (aplicando técnica de ingeniería de software).
  - Luego es replicado.
    - Ej: Plantas de estampado de CDs o DVDs.
  - Es eventualmente distribuído.
  - Y finalmente vendido (Wal-Marts, Electronic Boutiques, CompuWorlds, etc.).
- Se genera una “cosa” tangible: la cajita.

Ventajas del software como producto

- Esta concepción brinda varias ventajas:
  - El costo de diseñar el software se paga una vez y se cobra miles o millones de veces.
  - Podemos elegir la licencia que nos venga en gana, o bien diseñar nuestra propia licencia.
    - Ej: Prohibir que se pueda hablar mal del producto.
  - Se puede negociar con los vendedores de otros productos para que incorporen el nuestro a cambio de alguna prestación.
    - Ej: Firefox integrando a Google en el search box.
    - Ej: Dell instalando Windows en sus computadoras.

Ventajas del software como producto

- Esta concepción brinda varias ventajas:
  - Me permite agregar "planned obsolescence" para mantener el mercado cautivo.
    - Ej: Office 2003 no soporta OpenXML.
  - Si logramos una posición monopólica en el mercado podremos reducir los costos de desarrollo.
  - Más aun, podemos proteger nuestro monopolio mediante patentes de software que impidan o desalienten la competencia.
    - Incluso podemos hacer plata litigando.

Desventajas del software como producto

- No obstante, también existen ciertos inconvenientes:
  - Es difícil contemplar las necesidades de miles o millones de usuarios a la vez.
  - La competencia nos puede arruinar el negocio brindando un producto similar a menor costo.
  - Aun sin competencia, seguimos compitiendo con nostros mismos.
    - El software no se estropea.

Desventajas del software como producto

- No obstante, también existen ciertos inconvenientes:
  - El soporte técnico del producto vendido genera pérdidas.
    - Solución: dejar de soportar productos “legacy”.
  - Nos transformamos en los responsables legales del funcionamiento del producto.
  - La piratería nos afecta seriamente.
    - Solución: incorporar algún esquema de protección del software.

Aspectos no tan éticos del software como producto

- Se supone que la ley de oferta y demanda asegura a los clientes obtener el mejor producto al menor precio.
- Si bien la competencia es buena para el cliente, no lo es para el productor.
- Solución: si logramos monopolizar el mercado, podemos evitar la competencia.
  - Ej: El juicio anti-trust a Microsoft.

Patentes de software

- ¿Qué sucedería si alguien registra que tuvo la idea original de que al final del cuento el asesino fue el mayordomo?
- ¿Qué son las patentes de software?
  - Las patentes de software son similares a las patentes de invención comunes, con la diferencia que en vez de patentar una cosa, estamos patentando una idea.
- Lamentablemente, esto pasa hoy en día...

Las dos caras de las patentes de software

- Son un capital para quien las posee:
  - Mediante las patentes de software puedo impedir que la competencia provea un producto similar al mío.
    - Ej: las ventas One-Click en Amazon vs. B-N.
  - No están al alzance de todos, patentar una idea cuesta arriba de 50.000 USD.
- Son un riesgo para el resto del mundo:
  - ¿Qué pasa si mi programa incluye sin saberlo un algoritmo patentado?

Ejemplos de patentes de software

- Sorprendentemente, algunas patentes de software son bastante ridículas:
  - Microsoft patentó el operador que permite determinar si dos variables son diferentes.
  - Amazon patentó las ventas “one-click”.
  - Adobe patentó mostrar una paleta de colores usando pestañas.
  - Sun patentó como transformar los nombres de los archivos de W95 a NT.
  - IBM patentó reorgizar las ventanas cuando no se puede ver todo su contendo.

Sistemas de protección del software

- Existen dos modelos para los sistemas de protección del software:
  - Basados en software:
    - Siempre pueden ser crackeados.
    - Puede traer problemas de confiabilidad.
  - Basados en hardware:
    - Son apenitas más seguros.
    - Molestan al usuario.
    - Pueden traer problemas de compatibilidad.
- ¡La mejor solución consiste en evitarlos!

Software como producto

- En síntesis:
  - Este enfoque permite recaudar cuantiosas cantidades de dinero a las empresas ya establecidas (especialmente, las que controlan monopólicamente al mercado).
  - Es complicado que una empresa recien formada pueda tener éxito siguiendo este enfoque por diversas razones:
    - Alta inversión inicial.
    - Peligro de litigación.
    - Riesgo de dumping.

El software como servicio

- El software también presenta ciertas características que lo hacen un servicio:
  - El software no se fabrica, se desarrolla.
- En vez de llevar el producto a los clientes, éstos vienen a pedirlo.
- Generan una “cosa” no tangible: el servicio que es brindado por el software.
  - Ej: Gmail, Amazon, eBay, etc.
  - Ej: Salesforce.

Ventajas del software como servicio

- Esta concepción también brinda sus ventajas:
  - La mayoría del software sigue siendo a medida.
  - El mantenimiento y el soporte del software insume muchos recursos:
    - El 60% de todo el dinero gastado en el mundo cada año en software se va en mantenimiento.
    - El matenimiento insume el 70-80% del costo total del software.

Ventajas del software como servicio

- Esta concepción también brinda varias ventajas:
  - Podemos generar un gran número de puestos de trabajo.
    - La demanda es muy, muy alta.
  - Los usuarios en general no están conformes con el software que usan, siempre desean adaptarlo a sus necesidades o comonidades.
    - Gran oportunidad de negocio.
    - Dificil de lograr sin acceso al código fuente.
    - Los plug-ins son una solución de compromiso.

Ventajas del software como servicio

- Esta concepción también brinda varias ventajas:
  - El desarrollo que ha tenido internet brinda un nuevo espacio para el software como servicio: los servicios web.
    - El servicio que brindemos puede depender de servicios brindados a su vez por terceros.
  - El advenimiento de la semantic web mejora aun más las perspectivas de este modelo.
  - Gran oportunidad para aprovechar:
    - Reconversión de los sistemas existentes.

Ventajas del software como servicio

- Esta concepción también brinda varias ventajas:
  - Grandes empresas que apuestan por esta nueva visión ya nos proveen de las herramientas necesarias, por caso:
    - La arquitectura .NET de Microsoft.
    - Los applets y los servlets de Sun.
    - La tecnología AJAX (Gmail).
  - Se abaratan los costos de “deployment”:
    - Actualizaciones continuas y transparentes para el usuario final.

Ventajas del software como servicio

- Esta concepción también brinda varias ventajas:
  - Podemos usar los repositorios de software libre como punto de partida de nuestros desarrollos.
  - O bien, podemos usar software libre como la infraestructura de nuestro servicio.
    - Ej: La arquitectura LAMP.
  - Por último, también podemos centrar nuestro servicio en torno al soporte.
    - ¡Ni siquiera necesitamos programadores!

Clasificación del software

- El software puede clasificarse de acuerdo a diferentes criterios.
  - De acuerdo a su costo:
    - Software gratis o gratuito (freeware).
    - Software comercial o pago.
  - De acuerdo a la disponibilidad de su código fuente:
    - Software de código abierto (open source).
    - Software de código cerrado.

Clasificación del software

- El software puede clasificarse de acuerdo a diferentes criterios.
  - De acuerdo a su protección:
    - Software de dominio público (public domain).
    - Software abandonado (abandonware).
    - Software protegido por licencias:
      - Protegido por copyright.
      - Protegido por copyleft.
  - De acuerdo a su legalidad:
    - Software legal.
    - Software pirata.

Clasificación del software

- El software puede clasificarse de acuerdo a diferentes criterios.
  - De acuerdo a su filosofía:
    - Software propietario o privativo.
    - Software libre.
- Las compañías de software no suelen vender software.
  - Venden “permisos” para usar software.
  - Al comprar software enlatado estamos comprando una licencia de uso.

¿Qué es el software libre?

- Las cuatro libertades del software libre:
  - Libertad 00: la libertad para ejecutar el programa con cualquier fin.
  - Libertad 01: la libertad para estudiar y modificar el programa.
  - Libertad 10: la libertad de copiar el programa de manera que puedas ayudar a tus pares.
  - Libertad 11: la libertad de mejorar el programa y de hacer públicas esas mejoras, para beneficio de toda la comunidad.

GNU Public License

- El software libre también se licencia:
  - La GNU Public License (GPL) es la licencia de software libre más utilizada.
  - Esta licencia asegura las cuatro libertades.
  - Se comporta como una licencia de copyleft.
    - Es decir, una vez GPL siempre GPL.
  - Steve Ballmer, vice presidente de Microsoft, comparó a la GPL con un cancer...
    - ...y tiene razón!
    - La GPL resiste el Embrace, Extend & Extingish.

Otras licencias libres

- La Berkeley Software Distribution (BSD) es otra licencia libre muy popular.
- Esta licencia permite tomar software libre y transformalo en no libre.
  - Ej: Partes del TCP/IP stack del Windows fue tomado del sistema operativo NetBSD.
  - Esto la hace incompatible con la GPL
- Esta licencia no resiste al EEE.
- La licencia del Firefox es otro ejemplo de licencia libre: la Mozilla Public Licence.

¿En qué categoría cae el software libre?

- El software libre puede ser gratis o pago.
- Siempre es open source.
  - Libre implica open source.
  - Open source no implica libre.
- Se encuentra protegido por licencias del tipo copyleft.
- Siempre es legal, de hecho sería imposible piratear software libre.

Desventajas del software como servicio

- Como era de esperarse, aquí también se pueden observar ciertos inconvenientes:
  - ¿No será sanata?
    - Algunos sostienen que se trata de la programación orientada a componentes bajo un nuevo disfraz.
  - El software como servicio implica cambio.
    - El cambio siempre es resistido.
  - Ojo, no nos vamos a hacer ricos de la noche a la mañana.
    - La historia del creador del PC-Write no se va a repetir.

Desventajas del software como servicio

- Como era de esperarse, aquí también se pueden observar ciertos inconvenientes:
  - No ganamos nada con mantener una posición monopólica en el mercado.
  - Las patentes de software constituyen un inconveniente, especialmente si basamos nuestro servicio en software libre o de código abierto.
  - De usar software libre, está claro que no es posible ganar dinero regalando software...
    - ¿O sí?

¿Cómo hacer dinero con el software libre?

- Integrando software libre al hardware.
  - Ej: Fravega, etc.
- Proveyendo soporte.
  - Ej: help-desk, soporte las 24 horas.
- Utilizando software libre para crear soluciones integradas.
  - Ej: instalación de un ciber con software legal.
- Comerzializando software libre.
  - Ej: Red-hat cobra 1.500 USD por una copia de GNU/Linux (que se puede bajar gratis).

¿Cómo hacer dinero con el software libre?

- Desarrollando software a medida a partir de software libre.
  - Recordemos que la mayoría del software sigue siendo a medida.
  - El tomar software existente y en funcionamiento como punto de partida puede acortar los ciclos de desarrollo.
  - Si contribuyo mis modificaciones de vuelta a la comunidad de software libre, puedo capitalizar los aportes de los demás.
    - Sufiente cantidad de ojos atrapan todos los bugs.

Software como servicio

- En síntesis:
  - Parece brindar un cúmulo de altenativas que aun no han sido exploradas:
    - Software libre como punto de partida.
    - Software libre como infraestructura.
    - Servicios webs y la web semántica.
  - Situación ideal para empresas recién formadas o en proceso de formación.
  - La principal desventaja es que es imposible mantener cautivo al mercado.
    - Se debe competir de forma honesta.

Un modelo híbrido

- Finalmente, también es posible ensayar un modelo híbrido.
  - Ej: el programa “Software Assurance” de Microsoft.
- Por caso, podemos brindar el servicio de obtener de forma gratuita los productos que desarrollemos a lo largo de un cierto período.
- Goza y sufre de algunas de las ventajas y desventajas de ambos acercamiento.

Conclusiones

- El software presenta características como producto y como servicio.
- Ambas concepciones tiene sus ventajas y desventajas.
- El software como servicio ha encontrado dos nuevos aliados:
  - La existencia de gran cantidad de software libre de calidad.
  - El interés en torno a los servicios web.

Conclusiones

- La decisión final acerca de qué modelo nos conviene más a cada uno de nosotros, es una pregunta sin una respuesta universal.
- No obstante, es una pregunta relevante, dada la situación coyuntaral del pais.
  - Se sancionó una ley de promoción de la industria del software (Ley Nro. 25.922).
  - La demanda de licenciados e ingenieros supera casi 5 a 1 a la oferta de graduados.

¿Preguntas?

¡Gracias por su atención!