

## **CAPÍTULO 2. COMPARACIÓN DE LAS OPCIONES COMERCIALES DE SOFTWARE DE CÓDIGO CERRADO Y DEL SOFTWARE LIBRE EN EL MERCADO NACIONAL**

En este apartado se hará referencia a aquellos aspectos que caracterizan cada dos de las diferentes modalidades de software (y sin duda las principales) basándonos fundamentalmente en aquellos beneficios que presentan y que nos ofrecerán una visión comparativa acerca de los mismos.

Se han dejado de lado los criterios que tachan a algún tipo de software como obsoleto o ineficiente, los beneficios o desventajas que ofrece un software en particular o las características cuestionables de algún tipo de software o modelo de programación. Se presentan panoramas generales que si bien algunos no son evidentes a simple vista, todos pueden ser comprobados con facilidad.

### **2.1. Software de Código Cerrado**

Este modelo de elaboración de Software es es más difundido actualmente, esto se debe principalmente a que el sistema operativo MS-Windows es el más utilizado alrededor del mundo. En México la gran mayoría de las computadoras (hasta hace unos años su totalidad) tienen pre-instaladas el sistema MS- Windows y es común que el usuario mencione a MS-Windows como una parte de la computadora.

El software de código cerrado en México no se limita a programas elaborados por Microsoft, sin embargo, la presencia del software para la producción orientada a la demanda estandarizada se ve dominada por productos de otros países del mundo. Además las herramientas de desarrollo son también elaboradas por empresas extranjeras, lo cual influye en el software de código cerrado que es elaborado para adaptación a requisitos locales.

Por la forma de desarrollo del software de código cerrado, una empresa obtiene un contrato exclusivo y en contratante obtiene principalmente:

- La integración de las diversas herramientas disponibles para la plataforma se facilita

enormemente (obviamente con las limitaciones que presenta el uso de ésta plataforma).

- Hay grandes facilidades para el desarrollo con las herramientas que se proporcionan, aunque éstas están enfocadas únicamente a la programación de clientes ligeros e interfaces gráficas de usuario.
- Una sola compañía se podría encargar de todo el desarrollo. La cual ser la única responsable del buen funcionamiento de todo el proyecto.
- Se podría obtener grandes descuentos por compra de licencias.
- Una sola compañía podría encargarse del soporte total.

Suponiendo que la misma empresa proporciona la plataforma tecnológica y otras compañías los desarrollos:

- Gran cantidad de desarrolladores en varias consultoras proporcionan una excelente competencia para los desarrollos, que se refleja en menores costos del mismo.
- Gran cantidad de opciones para tener la plataforma tecnológica.
- Se podrían escoger unas cuantas compañías que se encarguen de todos los desarrollos bajando los costos por mantenimiento.

La tabla 1 de la página 34 nos da una visión general de los beneficios del modelo de Software de Código Cerrado.

*Tabla 1: Ventajas Software de Código Cerrado<sup>31</sup>*

| Ventaja.            | Descripción adicional.                   |
|---------------------|--|
| Control de calidad. | Las empresas tienen por regla general un |

---

31 CULEBRO Juárez, Montserrat. GÓMEZ Herrera, Wendy Guadalupe. TORRES Sánchez, Susana. Software libre vs software propietario: ventajas y desventajas [CD-ROM]. México: 2003.

|  |   |
|--|---|
|  | departamento de control de calidad para probar sus productos.   |
| Recursos destinados a la investigación.                          | La investigación es sobre usos del producto, no forzosamente a nuevas implementaciones.                             |
| Personal altamente calificado.                                   | Es común que los programadores con más experiencia trabajen en las compañías que desarrollan este software.         |
| Uso común para los usuarios.                                     | Debido principalmente a la popularidad de los programas.  |
| Software para aplicaciones muy específicas.                      | Este tipo de software realiza tareas particularmente útiles pero solo en ciertos S.O.                               |
| Amplio campo de expansión de uso en las universidades.           | En el caso de México, la mayor difusión la tiene este software. Esto se debe a acuerdos empresa-universidad.        |
| Difusión de publicaciones acerca del uso y aplicación.           | Existen numerosas publicaciones, ampliamente difundidas, que documentan y facilitan el uso de diferentes programas. |
| Curva de aprendizaje menor.                                      | Interfaces intuitivas y estandarizadas.   |
| Facilita la integración plataforma-aplicaciones.                 | La integración de diversas herramientas se facilita enormemente cuando se usa software de una sola compañía.        |
| Una sola compañía puede dar soporte a la totalidad del software. |   |
| Mejoras para el desempeño en la red.                             | Aunque su seguridad es dudosa, la empresas han  |

|                                    |  |
|------------------------------------|--|
|                                    | mejorado sus herramientas para el trabajo en red                                     |
| Numerosas aplicaciones exclusivas. | Debido a patentes, existen programas que no puede desarrollar ninguna otra compañía. |

Como en todo modelo, las desventajas del software de código cerrado también deben ser listadas. La razón principal para la siguiente tabla es que las desventajas de este tipo de software no son siempre evidentes y en la mayoría de los casos los inconvenientes mencionados se aplican solo a un programa en específico.

Si bien los virus son un problema común, es una desventaja específica de un software y no del modelo de programación. El problema con la seguridad se basa en que es imposible saber que medidas se han implementado para proteger la información.

Es por ejemplos como el anterior que la tabla 2 esta basada en los problemas comunes del software que son frecuentemente mencionados por los clientes, usuarios y expertos de forma general.

*Tabla 2: Desventajas Software de Código Cerrado<sup>32</sup>*

| Desventaja                      | Descripción adicional.  |
|---------------------------------|---|
| Cursos de aprendizaje costosos. | Es difícil usar el software de forma eficiente sin haber asistido a un curso.   |
| Código fuente secreto.          | En muchos casos resulta riesgosa la utilización o de un componente que es como una caja negra, cuyo funcionamiento se desconoce y cuyos resultados son impredecibles. |
| Soporte técnico ineficiente.    | En la mayor de los casos el soporte técnico es insuficiente o tarda demasiado tiempo en ofrecer una respuesta satisfactoria.  |

<sup>32</sup> Ídem 25.

|   |  |
|---|--|
| <p>Ilegal o costosa la adaptación de un módulo del software a necesidades particulares.</p> | <p>En caso de que sea vitalmente necesaria una modificación, es necesario pagar una elevada suma de dinero a la compañía fabricante, para que sea ésta quien lleve a cabo la modificación a su propio ritmo de trabajo y sujeto a su calendario de proyectos.</p>  |
| <p>Derecho exclusivo de innovación.</p>   | <p>La innovación sobre el producto es derecho exclusivo de la compañía fabricante.</p>   |
| <p>Ilegalidad de copias.</p>  | <p>El software solo puede ser instalado en la mayoría de los casos en una computadora, si se necesita copiarlo a otra máquina es necesario adquirir licencias adicionales.</p>   |
| <p>Imposibilidad de compartir.</p>  |  |
| <p>Soporte técnico desaparece.</p>  | <p>Si la compañía fabricante desaparece, también lo hace el soporte técnico, la posibilidad de tener versiones mejoradas y la posibilidad de corregir los errores de dicho software también desaparecen. Los clientes que contrataron licencias para el uso de ese software quedan completamente abandonados a su propia suerte.</p> |
| <p>Líneas de software con fecha límite.</p>   | <p>Es probable que la línea de software quede descontinuada y nunca más vuelva a tener una modificación.</p>   |
| <p>Dependencias a proveedores.</p>  |  |
| <p>Anulación del desarrollo tecnológico de la industria local.</p>                          | <p>Esto es en referencia a la industria extranjera que elabora aplicaciones de consumo masivo como</p>   |

|                             |  |
|-----------------------------|--|
|                             | sistemas operativos o paquetes de ofimática. |
| Seguridad dudosa.           |  |
| Costo elevado de licencias. |  |

## 2.2. Software Libre

Es posible decir sin temor a equivocarnos que el éxito de este modelo se encuentra en que aplica los principios de libre acceso a las fuentes de conocimiento que propugna la ciencia al ámbito del software.

Así pues, al igual que en ciencia toda persona que se lo proponga puede acceder al conocimiento, patrimonio de todos, en el Software Libre no existen trabas para que cualquier individuo pueda disponer de toda la enciclopedia informática que constituye el código fuente de los programas.<sup>33</sup>

Siguiendo con la analogía, en ciencia podemos considerar que el compartir los resultados de nuestras investigaciones y colaborar en otras es mejor:

- Epistemológicamente.<sup>34</sup>
- Éticamente.

La segunda aproximación, la que ve una ganancia ética en el Software Libre, tiene una clara ventaja sobre la primera; siempre existe. El Software Libre podrá decir que puede compartir el software creado bajo sus reglas pero si no tiene a nadie con quién compartirlo no deja de ser Software Libre, puesto que en potencia puede compartirse y, ya que el depositario del derecho de copia ha decidido utilizarlo de esa forma, será a vista de todos una pieza de

33 HERNÁNDEZ Saldivar, Ignacio. De la Globalización a la Sociedad del Conocimiento: Las TIC y la Educación desde el enfoque CTS. Tesis para optar al Posgrado de la Facultad de Ciencias Políticas y Sociales. México: UNAM, Enero 2004.

34 Epistemología: Parte de la filosofía que trata de los fundamentos y los métodos del conocimiento científico.

### Software Libre.<sup>35</sup>

Su disposición como autor de cierto software, a liberarlo según unas reglas libres, le garantiza inmediatamente un punto a favor en el conjunto de las conductas sociales positivas. Un vecino que comparte es considerado mejor vecino que el que no lo hace, por poco que comparta y pocos que puedan beneficiarse de su actitud.

Existe una comunidad de productores/consumidores de Software Libre que entienden suficiente esta visión, sin exigir contrapartidas prácticas de ningún tipo. Deciden liberar su código, permitir su copia y estudio y no poner trabas a su uso simplemente porque creen que son mejores ciudadanos que si hicieran lo contrario.

Junto a ellos, y no contra ellos en ningún caso, existe otro grupo más pragmático, que podríamos asociar con la visión epistemológica del Software Libre. Este colectivo opina que este tipo de software obtiene su fuerza en las ventajas materiales que de su práctica se derivan. Si deseáramos resumir en una sola frase cuál es su motivación, ésta sería *El Software Libre es el mejor Software posible* donde *mejor* se refiere a todos los apartados que la Ingeniería del Software analiza en un proyecto informático; desde la corrección hasta la estabilidad.

Regresando a la analogía con la ciencia, observamos que hay científicos que comparten lo que han aprendido y permiten que sus investigaciones sean la base de aquéllas de otras comunidades científicas porque creen que es un deber cívico no importa si reciben compensación alguna o no. Otros defenderán que existe un impulso más elemental que nutre tales decisiones; la ciencia hecha en colaboración es mejor ciencia debido a que produce mejores resultados y los genera en menos tiempo.

*Tabla 3: Ventajas del Software Libre<sup>36</sup>*

|         | Ventaja.                               | Descripción adicional.            |
|---------|--|-----------------------------------|
| Por sus | Bajo costo de adquisición y libre uso. | La principal remuneración esta en |

35 CIBERAULA.COM. Breve análisis del Software Libre [en línea]. [Fecha de consulta: diciembre 2008]. Disponible en: [http://linux.ciberaula.com/articulo/breve\\_analisis\\_software\\_libre/](http://linux.ciberaula.com/articulo/breve_analisis_software_libre/).

36 Ídem 25.

|                            |  |   |
|----------------------------|--|---|
| características.           |  | los servicios y no en el licenciamiento o la venta del producto, sin embargo esta es posible.   |
|                            | Acceso al diseño.  |   |
|                            | Modificable y expansible.                                |   |
|                            | Distribución libre.                                      |   |
| Por la libertad.           | Tiende a la eficiencia.                                  | Porque mucha gente lo optimiza, mejora.   |
|                            | Tiende a ser robusto.                                    |   |
|                            | Tiende a la diversidad.                                  | La gente que contribuye tiene muchas necesidades diferentes y esto hace que se adapte a una cantidad más grande de problemas.                 |
| Por motivación al público. | Fomenta el desarrollo de nuevo software.                 |   |
|                            | Posibilita controlar el software.                        | Esto es importante para aplicaciones de misión crítica donde es imperante tener un control total sobre posibles problemas en cualquier punto. |
|                            | Creación de aplicaciones de bajo costo.                  |   |
|                            | Reutilización del conocimiento contenido en el software. | Esto es equivalente a la manera en la que la ciencia se desarrolla: no se   |



|                    |   |   |
|--------------------|---|---|
|                    |   | parte de cero, se parte de los hallazgos previos y se innova sobre el conocimiento que ya se tiene.   |
|                    | Adaptación de necesidades específicas.                      |   |
|                    | Aprendizaje.  | Ayuda en el aprendizaje de un lenguaje de programación y procesos de la computadora.  |
|                    | Reconocimiento.   | Por parte de colegas y el público en general.   |
| Por su desarrollo. | Innovación tecnológica.                                     | Compartir información y trabajar de manera cooperativa favorecen la innovación.   |
|                    | Requisitos de hardware menores y durabilidad de soluciones. |   |
|                    | Escrutinio público.   | El proceso de revisión pública al que está sometido el desarrollo imprime un gran dinamismo al proceso de corrección de errores. Esta característica es importante sobre todo al implementar servicios de gobierno electrónico. |
|                    | Independencia del proveedor.                                | Cualquier empresa o profesional, con los conocimientos adecuados, puede seguir ofreciendo desarrollo de servicios para la aplicación.   |

|  |  |  |
|--|--|--|
|  | Industria local.   | Un profesional local puede dar soporte o elaborar nuevos programas.  |
|  | Seguridad.   | Cuando los datos son procesados electrónicamente, su vulnerabilidad está dada por el software que lo procesa.          |
|  | Adaptación.  | La personalización es un área muy importante en que el SL puede responder mucho mejor con costos mucho más razonables. |
|  | Lenguas minoritarias, traducción, uso e impulso de difusión. | Estas lenguas pueden ser incluidas mucho más fácilmente debido a la libertad y el acceso al código fuente.             |

El modelo de desarrollo basado en Software Libre es una herramienta realmente eficiente, sin embargo, el uso de este software no es tan maduro como se pensaría. La mayor parte del software desarrollado de esta forma fue hecho inicialmente para ser utilizado por programadores o gente experta en el computo. Esto ha tenido como consecuencia principal que el aprendizaje sea difícil, los programas poco intuitivos y los esfuerzos para cambiar esto igualmente complicados.

Incluso es posible asegurar que las mencionadas como ventajas del Software Libre cuando son descuidadas acaban siendo sus más grandes desventajas. Es el caso de la documentación del software y las pocas garantías que algunos proyectos ofrecen al usuario común que no tiene los conocimientos para revisar por si mismo un código fuente y su funcionamiento.

Afortunadamente las cosas están cambiando para este modelo que poco a poco atrae la atención de más empresas para su uso y nuevas compañías que respaldan el software también

son fundadas y bien posicionadas en diferentes partes del mundo haciendo acuerdos con compañías como Microsoft, lo cual demuestra que el Software Libre no puede ser ignorado.

Aun con un largo camino por recorrer este modelo se caracteriza hoy en día por tener las siguientes desventajas.

*Tabla 4: Desventajas del Software Libre<sup>37</sup>*

| Desventaja.   | Descripción adicional.   |
|---|--|
| Curva de aprendizaje mayor.   | Esto es notable si se trata de una migración al SL.  |
| No hay garantía del autor.  |  |
| Los contratos de licenciamiento no se hacen responsables por daños económicos y de otros tipos por el uso de sus programas. | El software libre se adquiere sin garantías explícitas del fabricante, sin embargo, puede haber garantías específicas para situaciones muy específicas.                            |
| Se necesita dedicar recursos a la reparación de errores.  | Sin embargo en el software de código cerrado es imposible reparar errores, hay que esperar a que saquen a la venta otra versión.   |
| No existen compañías únicas que respalden toda la tecnología.   |  |
| Las interfaces gráficas de usuario y la multimedia apenas se están estandarizando.  | Dependiendo el sistema elegido esto puede no ser verdad hoy día.   |
| La mayoría de la configuración de hardware no es intuitiva.   | Sin embargo la documentación referente a la configuración del hardware es tan explícita y detallada que permite al usuario novato profundizar en el conocimiento de su hardware en |

<sup>37</sup> Ídem 25.

|   |  |
|---|--|
|   | muy pocas horas y una vez teniendo ese conocimiento la configuración se vuelve trivial.  |
| Únicamente los proyectos importantes y de trayectoria tienen buen soporte.  | Estos proyectos importantes que tienen un excelente soporte cubren más del 90 % de las necesidades de cómputo del usuario promedio |
| El usuario debe tener nociones de programación.   |  |
| En sistemas con acceso a Internet hay que estar pendiente de corrección de errores.   | Sin embargo, el usuario puede reparar su propio sistema sin esperar la actualización del fabricante.                               |
| La diversidad de distribuciones, métodos de empaquetamiento, licencias de uso, herramientas con un fin común, etc., pueden crear confusión en el usuario inexperto. |  |

A lo largo de este capítulo se ha visto que en buena parte las ventajas y desventajas de los modelos de software y la mayoría son debidas a restricciones o libertades de cada modelo. Sin embargo existen factores comunes a todo software que deben ser tomados en cuenta al momento de elegir un programa o un servicio. Un sistema de alta calidad es aquél que cumple con las necesidades del cliente. El software debe ser útil y utilizable, es decir, hace más fácil o mejor la vida a las personas. Confiable, con pocos errores. Flexible porque las necesidades cambian con el tiempo, aún cuando el software se está desarrollando, entonces es importante poder hacer cambios posteriores. Debe ser posible darle mantenimiento después de liberado. La accesibilidad, tanto para comprar como para mantener, debe ser razonablemente fácil y rápido poderlo desarrollar o darle mantenimiento. La disponibilidad es esencial de otra forma no importa que tan bueno es. Debe ser capaz de ejecutarse el hardware disponible y con el sistema operativo disponible, etc. Debe existir y entregarse el software prometido.

Tabla 5: Comparativa entre Software Libre y Software de Código Cerrado<sup>38</sup>

| Característica. | Software Libre.   | Software Código Cerrado.   |
|-----------------|---|--|
| Utilidad        | Si. Sus aplicaciones resuelven las necesidades del usuario común y otras adicionales debido a aportes de la comunidad.  | Si. Sus aplicaciones resuelven todas las necesidades del usuario común con funcionalidades agregadas solo en la última versión.  |
| Intuitivo       | Parcialmente. Existen programas y distribuciones destinados al usuario final que han aportado ambientes amigables, sin embargo, aun no abarca todas las tareas.       | Si, todos los programas son fáciles de usar y con ambientes gráficos amigables.  |
| Confiabilidad.  | Si. Las comunidades toman en cuenta riesgos de filtración, imposibilidad de acceso o manipulación de la información. Los problemas pueden ser resultados rápidamente. | Parcialmente. Si bien existen soluciones certificadas por diversos organismos, la seguridad de los sistemas es en general dudosa; no es comprobable debido a la falta del código fuente. |
| Flexible.       | Si. Se puede adaptar fácilmente a las necesidades específicas de un usuario u organización a un bajo costo.   | No. En caso de ser posible es difícil y/o costoso es su modificación porque existe dependencia a proveedores.  |
| Disponibilidad  | Excelente. Puede adquirirse libremente en el momento que  | Regular. En el mejor de los casos es posible descargarlo de Internet y comprarlo en línea.   |

<sup>38</sup> Ídem 25.

|                           |  |  |
|---------------------------|--|--|
|                           | <p>el usuario lo necesite a bajo costo. Existen repositorios que facilitan una instalación en minutos. Adaptable al hardware disponible.</p> | <p>La adquisición es costosa. Tiene limitaciones de hardware para funcionar y dependencias adicionales de software.</p>  |
| <p>Accesible en costo</p> | <p>Accesible: Los costos de adquisición, adaptación, mantenimiento, son menores y retornan la inversión más prontamente.</p>                 | <p>Costosa: El pago de licencias soporte técnico y actualización son solo accesibles para grandes organizaciones. Imposibilidad de desarrollarlo fuera de la compañía creadora.</p>  |
| <p>Estandarizado</p>      | <p>Si. Funciona basado en protocolos estandarizados libres y sigue también estándares de programación para su fácil modificación.</p>        | <p>Si. Aunque implementa protocolos abiertos sus principales funciones se basan en protocolos privados y de difícil implementación para terceros. Se han tenido problemas con el seguimiento de estándares oficiales en diversos países.</p> |
| <p>Estable</p>            | <p>Si. Debido al escrutinio público y mantenimiento de la comunidad.</p>   | <p>Algunas veces. A falta del código fuente las herramientas pueden fallar inesperadamente.</p>  |