

INFORME

SOBRE LA SITUACIÓN DEL SOFTWARE DE FUENTES ABIERTAS
EN LAS EMPRESAS ESPAÑOLAS PROVEEDORAS
E INTEGRADORAS DE SISTEMAS EMPOTRADOS

2010

05

INFORMES CENATIC

Observatorio Nacional
del Software de Fuentes Abiertas



Centro Nacional de Referencia
de Aplicación de las TIC basadas
en fuentes abiertas

INFORME

SOBRE LA SITUACIÓN DEL SOFTWARE DE FUENTES ABIERTAS
EN LAS EMPRESAS ESPAÑOLAS PROVEEDORAS
E INTEGRADORAS DE SISTEMAS EMPOTRADOS

2010

Elaborado por:

Equipo CENATIC. Centro Nacional de Referencia de Aplicación de las TIC basadas en Fuentes Abiertas, Observatorio Nacional del Software de Fuentes Abiertas (ONSFA).

Con la colaboración de:

Ministerio de Industria, Turismo y Comercio, a través de red.es, la Junta de Extremadura, la Junta de Andalucía, el Principado de Asturias, el Gobierno de Aragón, el Gobierno de Cantabria, la Generalitat de Catalunya, el Govern de les Illes Balears y Eusko Jaurlaritza así como las empresas Atos Origin, Bull, Sun Microsystems, Telefónica y Gpex.

Edita:

©2010 CENATIC

C/. Vistahermosa, 1 - 3ª Planta

06200 Almendralejo (Badajoz)

Algunos derechos reservados

Depósito Legal: BA-000334-2010

ISBN: 978-84-693-2424-0

Maquetación y Producción: **ESI-Tecnalia**

Esta obra está bajo una licencia Reconocimiento 3.0 España de Creative Commons.



Para ver una copia de la licencia, visite
<http://creativecommons.org/licenses/by/3.0/es/>

Imágenes obtenidas de Wikimedia Commons (licencia GNU libre) y de stock.xchng

Este informe está disponible en el sitio web de CENATIC:
<http://www.cenatic.es>

INDICE

Prólogo	16
Presentación del informe	18
1. Resumen ejecutivo	21
1.1 Conclusiones generales	23
2. Introducción y plan del informe	27
2.1 Objetivos del estudio	28
2.2 Conceptualización de sistemas empotrados	29
2.3 Conceptualización de Software de Fuentes Abiertas	35
2.4 Relación entre sistemas empotrados y Software de Fuentes Abiertas	40
2.5 Plan del informe	42
3. Metodología de la investigación	45
3.1 Introducción	46
3.2 Diseño de la muestra y trabajo de campo	46
3.3 Modelo muestral	50
3.3.1. Distribución geográfica	50

3.3.2. Distribución por sector de actividad	51
3.3.3. Distribución por tecnología	52
3.3.4. Distribución por tamaño de empresa	53
4. Situación del Software de Fuentes Abiertas en los Sistemas Empotrados	57
4.1 Características estructurales de las empresas que trabajan en sistemas empotrados basados en Software de Fuentes Abiertas	58
4.1.1. Porcentaje de empresas que trabajan sistemas empotrados basados en Software de Fuentes Abiertas	59
4.1.2. Distribución de las empresas por CC.AA	60
4.1.3. Distribución de empresas por tamaño	61
4.1.4. Distribución de las empresas por sectores de actividad	61
4.1.5. Distribución de las empresas por tecnología	63
4.1.6. Visión de la cadena de valor	65
4.1.7. Mapa de capacidades de las empresas	67
4.2 Portafolio tecnológico: Software y Hardware de Fuentes Abiertas en las empresas de sistemas empotrados	69
4.2.1. Visión de las soluciones globales en el ciclo de desarrollo de sistemas empotrados	69
4.2.2. Mapa de soluciones globales de fuentes abiertas por sectores de actividad	72
4.2.3. Mapa de utilización de Software de Fuentes Abiertas	76
4.2.4. Uso del Hardware de fuentes abiertas	77
4.2.4.1 Herramientas de fuentes abiertas de Diseño Hardware	78
4.2.4.2 Microprocesadores de fuentes abiertas	79
4.2.4.3 Diseño de Hardware de fuentes abiertas	79
4.2.5. Uso del Software de Fuentes Abiertas: aplicaciones, sistemas operativos, librerías y herramientas	81
4.2.5.1 Sistemas Operativos	81
4.2.5.2 Aplicaciones - herramientas	83
4.2.5.3 Librerías	86
4.2.5.4 Licencias	88

4.2.5.5 Entornos de programación	88
4.2.5.6 Herramientas de integración	89
4.3 Actividades de I+D relacionadas con Software de Fuentes Abiertas en el ámbito de sistemas empotrados	90
4.3.1. Actividades de colaboración I+D	90
4.4 Comercialización y distribución de sistemas empotrados basados en Software de Fuentes Abiertas	92
4.4.1. Tipología de clientes	92
4.4.2. Valoración del Software de Fuentes Abiertas como seña de identidad de la empresa	93
4.4.3. Publicidad de los productos	93
4.5 Recursos Humanos dedicados a sistemas empotrados basados en Software de Fuentes Abiertas	94
4.5.1. Formación	94
4.5.2. Experiencia del personal dedicado a sistemas empotrados basados en Software de Fuentes Abiertas	95
4.6 Colaboración con las comunidades de desarrollo	96
5. Análisis estratégico	99
5.1 Debilidades	102
5.2 Amenazas	102
5.3 Fortalezas	104
5.4 Oportunidades	106
6. Conclusiones y recomendaciones	109
6.1 Conclusiones	110
6.2 Resumen DAFO	111
6.3 Recomendaciones y estrategias	112
6.3.1. Consideraciones estratégicas respecto a sistemas críticos empotrados	114
7. Equipo realizador y panel de expertos	119
8. Bibliografía y referencias	123

9. Anexo. Soluciones específicas para sistemas empotrados	129
9.1 Productos/Tipo software utilizados en infraestructura y componentes hardware base	130
9.1.1. IPCores	130
9.1.1.1 Leon3	130
9.1.1.2 Nios	131
9.1.1.3 PicoBlaze	131
9.1.2. Arquitectura y Plataformas hardware	131
9.1.2.1 Universal Plug&Play (UpnP)	131
9.1.2.2 Plataformas	131
9.1.2.3 Rifidi	132
9.1.3. Diseño Hardware	132
9.1.4. Herramientas	132
9.1.4.1 gEDA	132
9.1.4.2 ISE Web Pack (Xilinx)	133
9.1.4.3 SPICECAD	133
9.1.4.4 SystemC	133
9.2 Productos/Tipo software utilizados en infraestructura y herramientas software base	134
9.2.1. Sistemas operativos	134
9.2.1.1 Linux	134
9.2.1.2 Toppers	136
9.2.1.3 eCOS	137
9.2.1.4 FreeBSD	138
9.2.1.5 CentOS	139
9.2.1.6 FreeRTOS	140
9.2.1.7 MicroC/OS II	141

9.2.1.8 Micro Núcleos	142
9.2.1.9 Contiki OS	142
9.2.1.10 RTAI (Real Time Application Interface)	142
9.2.2. Middleware	142
9.2.2.1 TAO/ACE	142
9.2.2.2 gSOAP	143
9.2.2.3 CORBA	143
9.2.2.4 D-Bus	143
9.2.2.5 OSGi	144
9.2.2.6 OpenSplice DDS	145
9.2.3. Entornos de desarrollo	145
9.2.3.1 Eclipse	145
9.2.3.2 Kdevelop	146
9.2.4. Herramientas	146
9.2.4.1 Papyrus	146
9.2.4.2 UML TOPCASED	146
9.2.4.3 Cmake	147
9.2.4.4 MinGW	147
9.2.4.5 Bugzilla	147
9.3 Productos/Tipo software específicos del dominio	148
9.3.1. Procesado de imágenes	148
9.3.1.1 OpenGL	148
9.3.1.2 Cimg	148
9.3.2. Interfaces gráficas	148
9.3.2.1 QT	148

9.3.2.2 X, Xfree86	149
9.3.2.3 GNOME Mobile, GTK, Clutter	150
9.3.3. Comunicaciones	152
9.3.3.1 lwIP	152
9.3.3.2 Telepathy	152
9.3.4. Móviles	152
9.3.4.1 Maemo	152
9.3.4.2 Moblin	153
9.3.4.3 LiMo	154
9.3.4.4 Android	154
9.3.4.5 OpenMoko	155
9.3.5. Multimedia	155
9.3.5.1 Gstreamer	155

Tabla de Figuras

Figura 2.1: Porcentaje de capacidad de procesamiento. Sistemas Empotrados vs. Sistemas de computación generalistas	30
Figura 2.2: Estudio sobre sistemas empotrados en EEUU. Razones por las que no se utiliza un Sistema Operativo [Nasss 2008]	33
Figura 2.3: Estudio sobre sistemas empotrados en EEUU. Reutilización de código en proyectos de sistemas empotrados [Nass 2008]	42
Figura 3.1: Pregunta cerrada selección única	48
Figura 3.2: Pregunta cerrada selección múltiple	48
Figura 3.3: Pregunta semi-abierta	48
Figura 3.4: Pregunta abierta	49
Figura 3.5: Distribución geográfica	51
Figura 3.6: Distribución por sector de actividad	51
Figura 3.7: Distribución por tecnologías	52
Figura 3.8: Distribución de empresas por tamaño	54

Figura 4.1: Pregunta: Uso de Software de Fuentes Abiertas en alguna etapa del desarrollo	59
Figura 4.2: Distribución de Empresas de Sistemas Empotrados que utilizan Software de Fuentes Abiertas en alguna fase de su desarrollo	59
Figura 4.3: Distribución geográfica de las empresas	60
Figura 4.4: Distribución de empresas por tamaño	61
Figura 4.5: Distribución de las empresas por sectores de actividad	62
Figura 4.6: Distribución de las empresas por tecnología	63
Figura 4.7: Evolución de la lógica de negocio para los sistemas empotrados (Fuente: IBM Institute for Business Value[1])	65
Figura 4.8: Arquitectura de referencia	67
Figura 4.9: Mapa de capacidades	67
Figura 4.10: Mapeo de herramientas de fuentes abiertas según el ciclo en V.	69
Figura 4.11: Visión de soluciones de fuentes abiertas según la cadena de valor.	71
Figura 4.12: Mapa de soluciones globales de fuentes abiertas por sectores de actividad (I)	74
Figura 4.13: Mapa de soluciones globales de fuentes abiertas por sectores de actividad (II)	74
Figura 4.14: Matriz de capacidades	75
Figura 4.15: Empresas que utilizan herramientas Hardware de fuentes abiertas	77
Figura 4.16: Empresas que utilizan Micros de fuentes abiertas	78
Figura 4.17: Empresas que utilizan diseño Hardware de fuentes abiertas	80
Figura 4.18: Sistemas Operativos	81
Figura 4.19: Aplicaciones	83
Figura 4.20: Librerías	86
Figura 4.21: Licencias	87
Figura 4.22: Entornos	88
Figura 4.23: Herramientas de integración	88
Figura 4.24: Actividades de colaboración I+D	89
Figura 4.25: Tipología de clientes	91

Figura 4.26: Valoración del Software de Fuentes Abiertas	92
Figura 4.27: Publicidad	92
Figura 4.28: Formación de los recursos humanos	93
Figura 4.29: Experiencia del personal dedicado a sistemas empotrados basados en Software de Fuentes Abiertas	94
Figura 4.30: Colaboración con comunidades de desarrollo	95
Figura 5.1: Estructura de un análisis DAFO	101
Figura 6.1: Resumen DAFO	111

Tabla de Tablas

Tabla 2.1: Sistemas operativos empotrados	34
Tabla 2.2: Licencias Software de Fuentes Abiertas	38
Tabla 2.3: Compatibilidad entre Licencias Software de Fuentes Abiertas	38
Tabla 3.1: Empresas contactadas	48
Tabla 5.1: Debilidades del Software de Fuentes Abiertas en las empresas de sistemas empotrados	103
Tabla 5.2: Amenazas del Software de Fuentes Abiertas en las empresas de sistemas empotrados	104
Tabla 5.3: Fortalezas del Software de Fuentes Abiertas en las empresas de sistemas empotrados	106
Tabla 5.4: Oportunidades y perspectivas del Software de Fuentes Abiertas en las empresas de sistemas empotrados	107
Tabla 6.1: Matriz DAFO	112
Tabla 9.1: Arquitecturas Linux empotrado	135
Tabla 9.2: Arquitecturas eCOS	137
Tabla 9.3: Arquitecturas CentOS	140



PRÓLOGO

La industria de los sistemas empotrados y en general de los sistemas con inteligencia integrada depende cada vez más del software. El impacto en la productividad y eficiencia de los productos finales depende en gran medida del tipo de software implicado en las diferentes fases del ciclo de vida, constituyendo parte del producto en forma de avanzadas plataformas software o sistemas operativos en tiempo real o un medio de soporte, mediante herramientas básicas o avanzadas en forma de IDE, integrando capacidades de co-diseño, análisis, verificación o simulación.

Centrándonos en nuestro entorno cercano, la industria española desarrolladora e integradora de sistemas empotrados se encuentra

actualmente alineada con sus equivalentes europeos e internacionales en diferentes ámbitos y sectores, adoleciendo al mismo tiempo de los mismos problemas y dificultades. El mercado de los sistemas empotrados se ha caracterizado tradicionalmente por modelos restrictivos de patentado como elemento diferenciador entre empresas; no obstante, el competitivo entorno internacional, apunta a la implantación del Software de Fuentes Abiertas como una tendencia prometedora en la productividad de las empresas, adquiriendo de forma creciente visibilidad de cara al usuario final. Desde PROMETEO se percibe cómo el Software de Fuentes Abiertas se posiciona de forma global como un medio y fin cada vez más extendido en la industria.

La existencia de estudios que abordan tendencias en los sistemas empotrados es clara y visible, tanto nacional como internacional, no así el impacto del Software de Fuentes Abiertas en este ámbito. Este panorama presenta la necesidad de abordar un análisis que permita mostrar la utilización del Software de Fuentes Abiertas dentro de este mercado tan competitivo, abarcando la totalidad de la cadena de valor y los diferentes sectores en los que se puede enmarcar.

Con el objetivo de conocer la casuística existente, CENATIC, atendiendo a su carácter investigador y difusor ha promovido la realización del presente Informe sobre la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados.

PROMETEO, en la figura de su Comité Ejecutivo, quiere declarar su apoyo a las nuevas tecnologías orientadas a la productividad del sector, como las ofrecidas por el modelo de Software de Fuentes Abiertas. Es por ello que PROMETEO se congratula en secundar este estudio que establece un punto de partida a la hora de descubrir e indicar nuevas líneas estratégicas que permitan un liderazgo de nuestro tejido empresarial.

Comité Ejecutivo de PROMETEO

Francisco Ramos, Jesus Bermejo. Grupo Telvent.

Sergio Bandinelli. TecNALIA Corporación Tecnológica.

José Luis Burón. Acciona.

Juan Carlos Dueñas. Universidad Politécnica de Madrid.

Jesús Ángel Sánchez. Indra.

Iñaki Larrañaga. Mondragón.

PRESENTACIÓN

El presente informe, elaborado por CENATIC (Centro Nacional de Referencia de Aplicación de las TIC basadas en Fuentes Abiertas), la Fundación European Software Institute (ESI-Tecnalia) y la Asociación de Investigación y Cooperación Industrial de Andalucía, nace con el objetivo de estudiar la situación actual de las soluciones basadas en fuentes abiertas en España, desde el punto de vista de las empresas que constituyen la cadena de valor de los sistemas empotrados y tiempo real. El estudio se plantea de forma constructiva, constituyendo una reflexión que facilite el alineamiento de las capacidades reales de las empresas proveedoras de tecnología y los nichos de negocio existentes y oportunidades a futuro en el contexto global.

Debido a la alta heterogeneidad y diversificación de las empresas que tienen su actividad en el contexto de este informe, se toma como referencia la Plataforma Tecnológica Nacional de Sistemas con Inteligencia Integrada, PROMETEO, representativa del amplio espectro de empresas activas en el dominio multisectorial de los sistemas integrados.

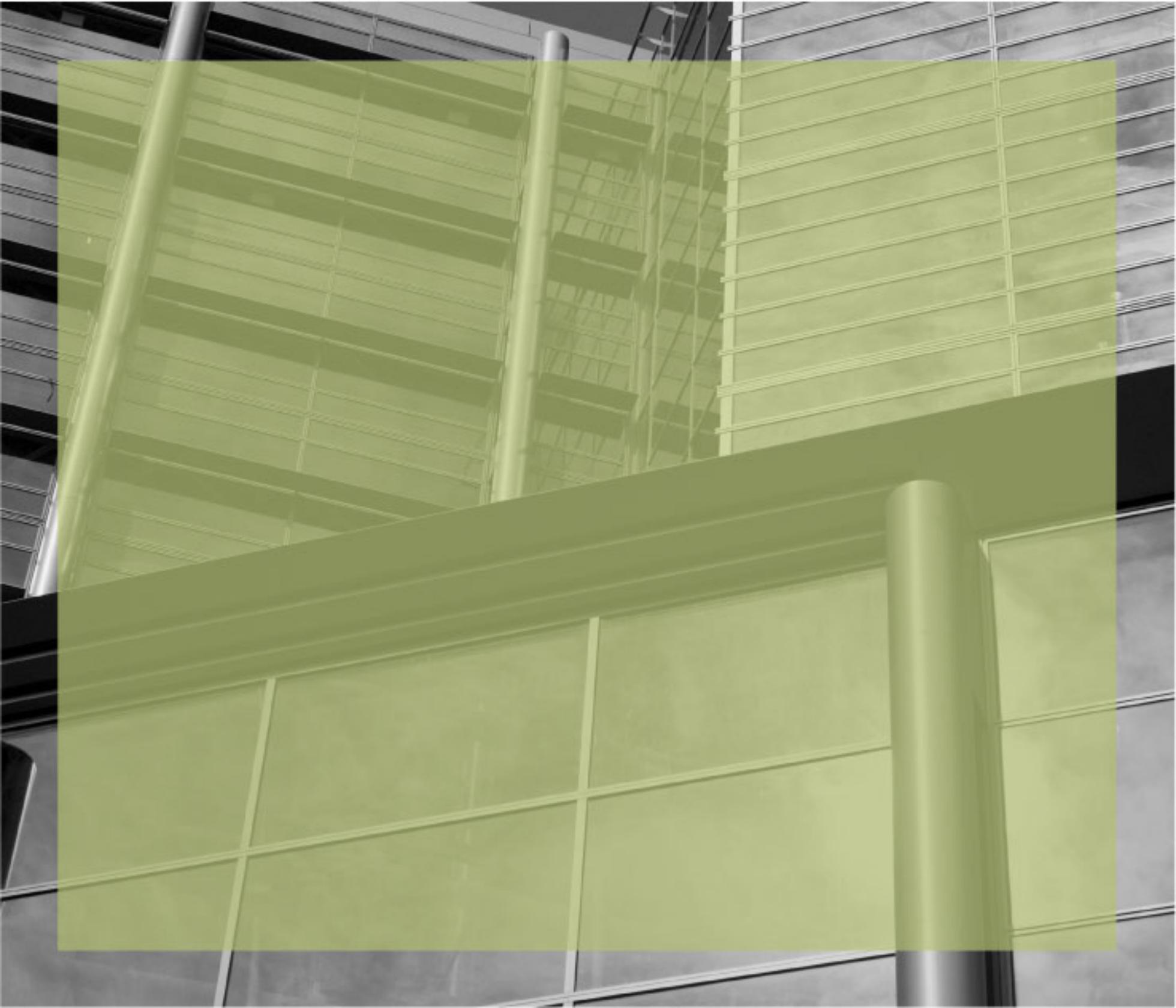
Por una parte, se ha consultado a las empresas integradoras sobre la utilización de Software de Fuentes Abiertas en las diferentes fases del desarrollo y gestión de sus productos, desde diferentes puntos de vista como costes, valor añadido, publicidad, madurez y avance tecnológico, formación tecnológica, I+D, internacionalización, etc. A partir de estos datos, se han identificado los posibles obstáculos (ya sean económicos, técnicos o legales) que pueden dificultar el desarrollo e implantación de estas tecnologías, así como los beneficios obtenidos y las oportunidades de negocio que se pueden establecer con la utilización de software libre dentro de los diferentes sectores.

Por otro lado, se ha definido un catálogo tecnológico, mapa de soluciones y capacidades de las empresas proveedoras de tecnologías abiertas en los diferentes niveles de la cadena de valor, evidenciándose los aspectos y tecnologías clave. Como síntesis de los resultados, se presenta un análisis estratégico de la situación actual, realizándose su análisis desde los dos puntos de vista: del proveedor y del consumidor/integrador de tecnología.

Por último, se plantean una serie de recomendaciones y consideraciones estratégicas que aceleren el posicionamiento de las empresas españolas de tecnología basada en fuente abierta en el tejido industrial, y en particular de los sistemas críticos empotrados.

Miguel Jaque Barbero

Director Gerente de CENATIC



01

RESUMEN EJECUTIVO

A continuación se ofrece un compendio de las conclusiones ejecutivas más importantes para cada uno de los ámbitos del estudio.

01

RESUMEN EJECUTIVO

El presente informe ha sido elaborado por CENATIC, la Fundación European Software Institute (Tecnalia) y la Asociación de Investigación y Cooperación Industrial de Andalucía, con el objetivo de estudiar la situación en la que se encuentra el Software de Fuentes Abiertas en empresas españolas proveedoras e integradoras de sistemas empotrados.

El Software de Fuentes Abiertas ha tenido un gran impulso en las últimas décadas, inicialmente debido al trabajo de desarrolladores a título particular, y a las comunidades de desarrollo soportadas por pequeñas y medianas empresas, y posteriormente debido al impulso que ha recibido esta tecnología por parte de las Administraciones Públicas, ya que favorece la interoperabilidad, garantiza la independencia con respecto a los proveedores, son soluciones más seguras y a la vez promueven el desarrollo del tejido TIC local. Así mismo, las grandes empresas cada vez son más conscientes de las posibilidades que se presentan y han adoptado este tipo de software como núcleo y objetivo de su actividad.

Existen numerosos estudios e informes que describen la situación del Software de Fuentes Abiertas, pero pocos se han centrado hasta el momento en los sistemas empotrados. **El mercado de sistemas empotrados, condicionado por los restrictivos modelos de patentado, modelos de utilidad del hardware y la gran pluralidad existente, no ha sido tradicionalmente objetivo de este tipo de software. Sin embargo en los últimos años, con los grandes avances que se están produciendo en la capacidad tecnológica de estos equipos, las iniciativas de estandarización abierta y la fuerte competencia global permiten el aprovechamiento de las ventajas competitivas aportadas por este tipo de software.**

Este informe presenta los resultados del estudio realizado sobre la situación de sistemas empotrados basados en Software de Fuentes Abiertas en España, desde el punto de vista de los proveedores e integradores de tecnología. En el transcurso del estudio se han analizado diferentes estudios sobre sistemas empotrados, principalmente enfocados hacia diferentes actividades comerciales, pero no existe un

referente claro que detalle y analice la situación del Software de Fuentes Abiertas dentro de los sistemas empotrados en los diferentes sectores.

Durante la realización de este informe se ha elaborado un mapa de capacidades existentes en el país y se han identificado los posibles obstáculos (ya sean económicos, técnicos o legales) que pueden dificultar el desarrollo e implantación de las tecnologías basadas en fuentes abiertas en el sector industrial, así como los beneficios obtenidos y las oportunidades de negocio que se pueden establecer con la utilización de Software de Fuentes Abiertas dentro de los diferentes sectores que desarrollan sistemas empotrados.

Para ello, se ha analizado la importancia que las propias empresas expresan sobre la utilización de Software de Fuentes Abiertas en las diferentes fases del desarrollo y gestión de sus productos, afrontándolo desde diferentes puntos de vista como costes, valor añadido, publicidad, madurez y avance tecnológico, formación tecnológica, colaboración con Universidades y centros tecnológicos, internacionalización, etc.

El análisis realizado identifica cómo la introducción del Software de Fuentes Abiertas en la industria de los sistemas empotrados se puede considerar ya un hecho, reconociendo el valor añadido que presenta el software de fuente abierta, pese a no ser generalizado aún en todas las fases de desarrollo. Así mismo, el avance en el conocimiento popular y por parte del personal técnico y de gestión de este tipo de software hace que éste tipo de software se incluya en la mayoría de las empresas dentro de las alternativas a considerar en la planificación del desarrollo de los productos. Aún así, existen todavía barreras, algunas de ellas culturales además de tecnológicas y legales, que influyen en su no generalización dentro de los sistemas empotrados.

A pesar de que el estudio abarca un mercado multisectorial, mediante unas líneas estratégicas comunes a la mayoría de sistemas empotrados se

ha identificado que las tecnologías de fuentes abiertas más ampliamente implantadas son aquellas que aportan una madurez internacional más extensa. Otra característica que está asociada a una fuerte implantación es el respaldo proporcionado por una comunidad o una gran empresa volcada en el proyecto.

1.1

>> Conclusiones generales

- **Los mayores usuarios de Software de Fuentes Abiertas en las empresas nacionales son las PYMES y Micro-empresas.** La necesidad de crecimiento, internacionalización, búsqueda de innovación, colaboración en proyectos I+D y colaboración con Universidades y centros tecnológicos justifican que sean este tipo de empresas las que presentan una mayor implantación de software de fuentes abiertas en sistemas empotrados.
- **El contexto abierto y creativo de la I+D en España y Europa contribuye de manera importante en la difusión y utilización de Software de Fuentes Abiertas.** Existe un impulso significativo del Software de Fuentes Abiertas por parte de los proyectos y unidades I+D de las empresas, derivado de aspectos propios relacionados con las oportunidades de colaboración entre entidades y unidades de desarrollo innovadoras sobre comunidades abiertas. Este aspecto, unido a la creciente importancia que se da a la I+D dentro de las empresas españolas, crea un ambiente propicio para el

fortalecimiento de las comunidades de Software de Fuentes Abiertas.

- **Existen nichos aplicativos importantes del software de fuentes abiertas para sistemas empotrados.** Las aplicaciones con exigencias críticas de fiabilidad representan un mercado difícil para el software de fuentes abiertas, lo cual requiere tareas de I+D en certificación de Software de Fuentes Abiertas y en la definición de estándares de interoperabilidad abiertos. Por otro lado, los mercados con menores implicaciones de criticidad en fiabilidad y confianza facilitan la adopción del software de fuente abierta.
- **La credibilidad del Software de Fuentes Abiertas está fuertemente ligada al grado de implicación de compañías fuertes en sistemas empotrados.** En particular, dentro de los sistemas empotrados para aplicaciones empotradas, un aspecto relevante que ha fomentado el uso del Software de Fuentes Abiertas es la entrada de empresas consolidadas en comunidades de referencia o en modelos de licenciamiento abiertos, tales como Wind River, Green Hills o MontaVista.
- **La experiencia con herramientas de desarrollo de fuente abierta representa un modelo a seguir en otras áreas de interés para el desarrollo de sistemas empotrados.** En especial, el aumento en el uso de Eclipse es resultado de su modelo de gestión basado en una Fundación sin intereses de lucro y con contribuciones de compañías grandes. La creación y fomento de ecosistemas de desarrollo abiertos y orientados a la generación de propiedad industrial, ligados a modelos de licenciamiento que favorecen la protección de la propiedad intelectual (como la licencia EPL de Eclipse) permite adelantar la futura expansión de las herramientas de desarrollo de fuente abierta en los mercados ligados a los sistemas empotrados.

- **El hardware abierto es la solución a la mayoría de los problemas asociados a cores propietarios.** Cada core abierto contendrá una amplia base de usuario, lo cual asegura un mejor soporte, una mejor documentación y unos mejores ejemplos de implementación desde los que trabajar. El código está disponible, por lo que cualquier desarrollador puede averiguar lo que necesita saber sobre el core. Finalmente, cuando los cores y sus estándares se desarrollen, los cores llegarán a ser más compatibles que los cores propietarios.



La introducción del Software de Fuentes Abiertas en la industria de los sistemas empotrados se puede considerar ya un hecho, reconociendo el valor añadido que presenta el software de fuente abierta, pese a no ser generalizado aún en todas las fases de desarrollo. Así mismo, el avance en el conocimiento popular y por parte del personal técnico y de gestión de este tipo de software hace que éste tipo de software se incluya en la mayoría de las empresas dentro de las alternativas a considerar en la planificación del desarrollo de los productos. Aún así, existen todavía barreras, algunas de ellas culturales además de tecnológicas y legales, que influyen en su no generalización dentro de los sistemas empotrados.





02

INTRODUCCIÓN Y PLAN DE INFORME

Como ya se ha comentado anteriormente, este informe pretende ayudar a CENATIC en sus labores de conocer, describir, medir y evaluar los efectos que el SFA (Software de Fuentes Abiertas) está teniendo sobre las empresas y la economía, así como su implicación en la construcción de un modelo de desarrollo empresarial sostenible, basado en la cooperación, la innovación, la transferencia de información y conocimiento y la excelencia.

02

INTRODUCCIÓN Y PLAN DE INFORME

2.1

>>Objetivos del estudio

El presente estudio, basado en el análisis de la literatura existente y los datos obtenidos mediante la aplicación de una encuesta a empresas del sector, pretende ofrecer una visión actualizada del nivel de la implantación del Software de Fuentes Abiertas en las empresas, que constituyen la cadena de valor de los sistemas empotrados:

- Diseño
- Fabricación
- Integración
- Comercialización

Se hace especial hincapié en los beneficios y retos ante los que se

enfrenta el uso e implantación de Software de Fuentes Abiertas dentro de este entorno industrial, recabando información sobre las posibilidades de apertura de nuevas oportunidades de negocio facilitadas por estos nuevos modelos de explotación. Esto incluye el análisis de la problemática específica existente en las migraciones de sistemas en entornos de software privativo a uno de fuente abierta, dentro del ámbito de los sistemas empotrados.

Dentro de la estrategia seguida en el análisis de los resultados de la encuesta realizada, se han establecido una serie de líneas maestras para la determinación y contextualización de los resultados dentro del ámbito que nos ocupa.

En primer lugar, se han estudiado las características estructurales de las empresas que trabajan en sistemas empotrados, buscando una maximización del conocimiento extraído en los distintos tipos y estructuras de empresa que se pueden encontrar en el panorama nacional. Este estudio pretende no sólo evaluar los sectores propiamente dichos, sino también los nichos de industria que pueden ser abordados gracias a la utilización de Software de Fuentes Abiertas.

Los beneficios, oportunidades y dificultades en la implantación de este tipo de software es otro de los puntos relevantes del estudio, sobre todo a la hora de comprender su implantación total o parcial dentro de la cadena industrial. Esta definición va a permitir establecer una visión comparativa de las implicaciones que genera el uso del Software de Fuentes Abiertas respecto al software privativo, marcando el valor que proporciona al producto y a la marca de la empresa.

Dentro del marco de implicaciones de su uso, hay un factor que se presume de gran incidencia sobre la apreciación y valoración del uso del Software de Fuentes Abiertas dentro de la empresa: el nivel de profesionalización que es capaz de presentar el mercado laboral ligado al software de fuente abierta, sobre todo referido al valor productivo que pueden aportar estos profesionales con sus capacidades específicas dentro del campo de Software de Fuentes Abiertas.

Como se ha comentado anteriormente, España se encuentra en una situación excelente dentro del ámbito europeo. Es por ello que los resultados del presente estudio mostrarán no sólo la situación a nivel nacional, sino también el ámbito internacional en el que se enmarcan las empresas.

El informe presenta un análisis que enmarca y contextualiza el tipo de empresas, al mismo tiempo que establece el uso que se hace del

Software de Fuentes Abiertas, así como la importancia tanto a nivel comercial como de utilidad dentro del desarrollo de la cadena de producción. Se han establecido también comparativas con los tipos de herramientas, aplicaciones y sistemas operativos que tienen más éxito entre las empresas.

Otro de los factores a los que está íntimamente ligado el uso de Software de Fuentes Abiertas es la innovación. Se ha buscado arrojar luz sobre la relevancia de tener departamentos de I+D dentro de la empresa para el uso y promoción del Software de Fuentes Abiertas dentro de las mismas.

2.2

>> Conceptualización de sistemas empotrados

El análisis que se ha desarrollado en la ejecución de este estudio está basado en la situación en la que se encuentra el Software de Fuentes Abiertas en una disciplina en auge como son los sistemas empotrados que abarcan hoy en día una gran variedad de sectores industriales y son de un uso cotidiano.

Actualmente se estima que el 90% de la capacidad de computación se encuentra en sistemas empotrados, frente al 10% en sistemas de computación generalistas [ARTEMIS SRA2006].

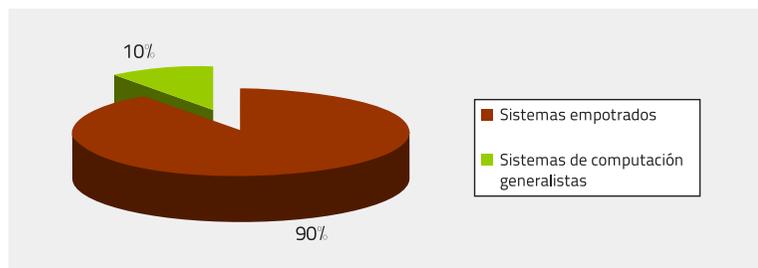


Figura 2.1: Porcentaje de capacidad de procesamiento. Sistemas Empotrados vs. Sistemas de computación generalistas

Actualmente, los sistemas empotrados están presentes en una amplia variedad de ámbitos, como puede ser en automóviles, dispositivos médicos, aviones, fábricas, redes eléctricas, en nuestros hogares y en el trabajo. La tasa de crecimiento anual de los sistemas empotrados se encuentra en el 10% y se prevé alcanzar aproximadamente la cantidad de 40 billones de dispositivos a lo largo del mundo en 2020.

La importancia de los sistemas empotrados en muchos sectores industriales se puede ver claramente en la industria automovilística. Este sector tiene



un volumen de negocio en Europa alrededor de 500 billones de euros y emplea cerca de 2,7 millones de personas.

Dentro de la fabricación de automóviles, la inclusión de sistemas empotrados aporta hoy en día un 20% del coste de cada coche, y hay previsiones de que aumente dicho porcentaje hasta un 35-40% para el año 2015; en otro

tipo de automóviles, como los híbridos, el porcentaje de coste debido a los componentes electrónicos puede llegar fácilmente a un 50% [Ciria Automoción].

Si un coche de combustible fósil de gama media en los años 80 incluía alrededor de cinco dispositivos electrónicos instalados, en la actualidad esa cifra se incrementa hasta 30 dispositivos. Si se considera un coche de gama alta, la cifra se puede disparar hasta la centena aproximadamente: control del motor, consumo, gases, climatización, seguridad, control del frenado, instrumentación, control de la velocidad, antirrobo, mantenimiento, sistema de navegación, control de luces, ordenador de viaje, centro multimedia, etc.

Este incremento del valor añadido proporcionado por la electrónica dentro del coche creará en los próximos años más de 600.000 nuevos empleos en Europa, únicamente en la industria de Sistemas Empotrados para el sector automovilístico [ARTEMIS SRA2006].



En sectores como el aeronáutico se prevé que los sistemas empotrados, a un corto plazo mejoren la eficiencia energética de los aviones en un 30%. En este campo, los avances en alta precisión, predicción de comportamiento de sistemas y fiabilidad se da mediante los avances dentro de los sistemas empotrados. Dentro de este sector existen innumerables campos en los que los sistemas empotrados pueden establecer una clara diferencia en las mejoras que se persiguen: sensorización específica, introducción a los sistemas *fly by wire*, certificación, estándares como ARINC, diagnósticos avanzados, mantenimiento predictivo que permita alargar el ciclo de vida de los aviones, navegación, simuladores de vuelo, realidad virtual y aumentada etc. Gran parte de todos estos sistemas son especialmente críticos para el funcionamiento del aparato, por lo que todos los avances en sistemas críticos de tiempo real, fiabilidad, seguridad, robustez son de una gran aplicabilidad en este sector.

Pero no sólo se pueden apreciar sistemas empotrados dentro de los aviones, muchos de los servicios ofertados por las compañías aéreas están sustentados por sistemas empotrados, en este sentido, se prevén mejoras sustanciales dentro de la conectividad en vuelo, así como servicios ofertados en aeropuertos (gestión de la facturación y emisión de billetes electrónicos, gestión de equipajes, información al usuario, etc.).

Sectores como la manufactura e industria hacen un uso exhaustivo de los sistemas empotrados. En este campo los principales objetivos están centrados



en la reducción de tiempo de fabricación, al mismo tiempo que se busca una flexibilidad total para una rápida adaptación a unos mercados muy cambiantes y particularizados a cada cliente. Reducciones de

fabricación del orden de 8-12 semanas a 1-2 semanas son objetivos que se pretenden alcanzar mientras se producen mejoras de la calidad y aprovechamiento de residuos. Avances en este campo pueden mover capacidades humanas desde la fabricación de productos hacia el diseño y fabricación del equipamiento.

Donde se puede apreciar de forma visible el gran auge de los sistemas empotrados es en el sector de bienes de consumo y comunicaciones. Estos sistemas empotrados están fuertemente implantados en la vida diaria de los ciudadanos. Por ejemplo, dentro del sector de comunicaciones, según la CMT (Comisión del Mercado de las Telecomunicaciones) la telefonía móvil en España alcanzó los 52 millones de líneas durante el 2009, llegando a 109, lo que equivale a 1 línea por cada 100 habitantes. La telefonía fija, pese a no aumentar el número de líneas contratadas, se mantiene en un ratio de 43,6 líneas por 100 habitantes, y la tasa de penetración de los servicios de acceso dedicado a Internet es de 20,19 líneas por cada 100 habitantes [CMT2009].

Nuevos mercados, como el de la televisión digital terrestre se presentan como expectantes oportunidades de desarrollo para la industria del sector. Desde finales del 2005, cuando se produjo el lanzamiento de la TDT en España, hasta abril del 2010, se procederá a un salto tecnológico en el 95% de los receptores, bien sea por la adquisición de un nuevo aparato con sintonizador incorporado, o con la adquisición de un sintonizador independiente. Este avance tecnológico implica un fuerte aumento de ventas dentro del sector, que se prevé que siga aumentando según se vaya alcanzando la televisión digital terrestre de pago y la alta definición.

La tendencia a incluir nuevas funcionalidades y la interconexión de diferentes aparatos, ha exigido a industrias manufactureras de bienes de consumo incorporar electrónica más compleja dentro de sus dispositivos. Los edificios, por ejemplo, demandan nuevas características como ahorro energético, gestión eléctrica, automatización, mayor grado de confort, gestión multimedia, diferentes tipos de comunicaciones, alarmas, seguridad, etc. que son suministradas por un ejército de sistemas empotrados: los nuevos electrodomésticos inteligentes, diferentes equipamientos con inteligencia integrada, sensores y redes de comunicaciones definidos en un hogar digital o vivienda inteligente, etc. En otras palabras, el sector de la inmótica/domótica propone nuevos y exigentes campos de aplicación para las tecnologías software/hardware integradas en el hogar digital.

Considerando el sector público, existen varios retos y oportunidades para mejorar la operación y seguridad de las infraestructuras públicas para llegar a una economía competitiva. Las infraestructuras públicas deben aprovechar la movilidad de las personas y de los equipamientos, la mejor red de comunicaciones, las nuevas capacidades ofrecidas por los adelantos tecnológicos en los sistemas empotrados para mejorar la simplicidad de uso, la conectividad, mantenimiento, la interoperabilidad, flexibilidad y seguridad en campos como el transporte, edificación, mobiliario urbano, control de acceso a edificios públicos, seguridad ciudadana, etc.

flexibilidad y seguridad en campos como el transporte, edificación, mobiliario urbano, control de acceso a edificios públicos, seguridad ciudadana, etc.

INNOVACIÓN
CREATIVIDAD
PRECIO

El mercado de los sistemas empotrados se caracteriza por una constante búsqueda de innovación y nuevos productos y servicios que incrementen funcionalidades al mismo tiempo que una disminución del precio. Dicha caracterización se da en prácticamente todos los sectores industriales donde se encuentra involucrado. En este contexto, el Software de Fuentes Abiertas puede contribuir al logro de dichos objetivos.

Los **sistemas empotrados** se pueden definir de manera técnica como un conjunto de componentes electrónicos y de software asociados que son capaces de cumplimentar funciones específicas. Dichos dispositivos verifican requisitos de tamaño, coste, consumo, eficiencia, fiabilidad y transparencia al usuario final dependientes de las especificaciones finales del producto.

Tradicionalmente, los equipos industriales de medida y control están basados en un microprocesador o microcontrolador con un sistema operativo propietario o específico para la aplicación correspondiente. Con los modernos sistemas multipropósito empotrados, basados en muchos casos en microprocesadores de propósito general como los x86, se llega a integrar el mundo del PC compatible con las aplicaciones industriales. Entre otras ventajas se pueden destacar las siguientes:

- Posibilidad de utilización de sistemas operativos potentes que ya realizan numerosas tareas: comunicaciones por redes de datos, soporte gráfico, concurrencia con lanzamiento de threads, etc.

- Herramientas de desarrollo software potentes.
- Reducción en el precio de los componentes hardware y software.

Una de principales implicaciones del uso de sistemas empotrados se encuentra focalizado en aquellos aspectos relacionados con la eficiencia y rendimiento de los sistemas ligados a aplicaciones con altos requerimientos de explotación hardware o temporal. A continuación se va a realizar una pequeña sinopsis de los principales elementos involucrados en el diseño y desarrollo de sistemas empotrados.

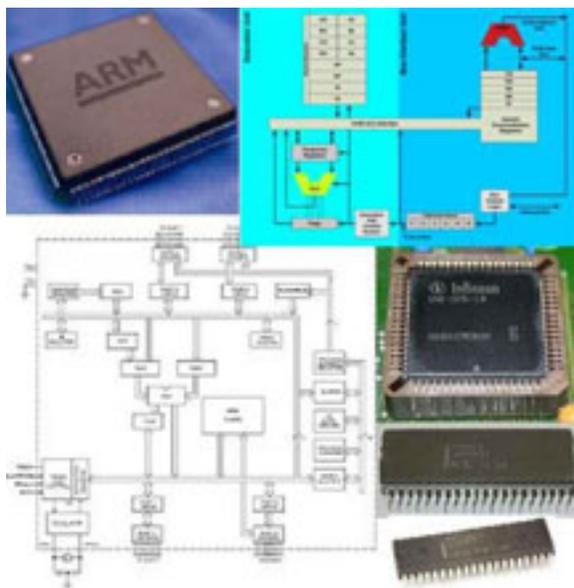
Uno de los aspectos a destacar es la **plataforma hardware** a la que están ligados la implantación de este tipo de sistemas empotrados, como pueden ser el caso de x86, ARM, arquitecturas RISC o VILW (de 4-bits a 64-bits), etc.

La plataforma hardware es la parte tangible involucrada en el desarrollo de sistemas empotrados. Su elección se decide en base a diferentes factores que dependen del enfoque práctico y de la finalidad y características funcionales requeridas para el sistema empotrado. Su elección va a determinar la elección de varios elementos, como es el caso de los lenguajes de programación asociados, así como las características de procesado de señal involucrado en la definición del tratamiento de la información y su manejo.

Los **lenguajes de programación** más habituales, pasan por ensambladores específicos de cada arquitectura de microprocesadores, microcontroladores y DSPs, hasta los diferentes estándares del lenguaje C.

Actualmente, se están definiendo nuevas alternativas gracias al incremento de la capacidad de procesamiento en nuevos dispositivos como es el caso de dispositivos móviles, asistentes personales digitales,

etc. En ellos es posible la utilización de software más pesado como es el caso de aplicaciones tipo Java que hasta ahora ha estado completamente excluido en este tipo de dispositivos electrónicos.



Las limitaciones de recursos, así como la capacidad de procesamiento proporcionada por los primeros microprocesadores y controladores han impulsado desde los primeros **sistemas empotrados** el desarrollo de aplicaciones que realizaban las funciones de los sistemas operativos, logrando de esta manera que se hiciera un uso eficiente de los recursos de los que se disponía. Estos sistemas operativos propios, integrados en la propia aplicación del dispositivo o parte de la propia aplicación, incorporan al sistema todos los controladores necesarios para los componentes, así como una gestión completa de todos los recursos: memoria, reloj, comunicaciones, interfaces, gestión de las tareas e interrupciones, etc.

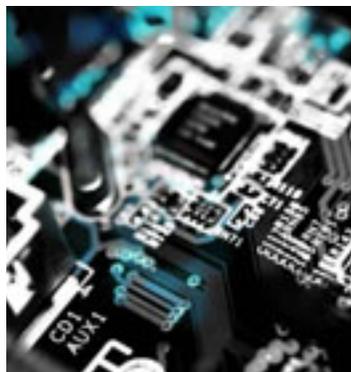
En los sistemas empotrados, el sistema operativo debe prestar especial atención hacia los mencionados aspectos técnicos que van a tener una influencia vital sobre el comportamiento, eficiencia y eficacia del sistema.

En un estudio realizado en el mercado estadounidense de sistemas empotrados en el año 2008 [Nass 2008] se comprueba que todavía un amplio sector de los desarrolladores de sistemas empotrados no implementa sistemas operativos en sus productos, pero es de señalar como disminuye la opinión generalizada de que la utilización de sistemas operativos en empotrados requiere mucha memoria, capacidad de computación, un esfuerzo económico y de personal.



Figura 2.2: Estudio sobre sistemas empotrados en EEUU.
Razones por las que no se utiliza un Sistema Operativo [Nass 2008]

Con el desarrollo y abaratamiento de los componentes electrónicos, así como con el avance de la capacidad de procesamiento, en la actualidad muchos productos incorporan sistemas operativos específicos para sistemas empotrados sobre los que corren las aplicaciones. Entre otras características, en los sistemas operativos orientados hacia el ámbito empotrado la gestión de memoria, las interrupciones, multitarea, relojes de tiempo real, etc. son elementos clave a tener en cuenta.



Una de las áreas de aplicación más importantes de los sistemas empotrados, reside en aquellos relacionados con los **sistemas críticos**. Este tipo de sistemas están íntimamente ligados a aplicaciones donde un fallo o un mal funcionamiento llevaría a:

- Muerte o graves efectos para los clientes.
- Pérdida o serio daño sobre los equipamientos.
- Daño medioambiental.

Se pueden considerar sistemas críticos el control en automóviles, aviones, equipamiento médico, etc. Este tipo de sistemas deben funcionar durante periodos de tiempo extremadamente largos, tolerando errores humanos, o fallos de hardware o de software, y proporcionando a la vez un nivel aceptable de servicios en todo momento.

Otro de los campos de actuación de los sistemas empotrados se encuentra enmarcado dentro de los sistemas en tiempo real, puesto que hacen un uso extensivo de sus características específicas para aumentar la agilidad de las comunicaciones y aprovechamiento máximo de las propiedades hardware y software de los sistemas.

Un **Sistema en Tiempo Real (STR)** es aquel sistema digital que interactúa activamente con un entorno con dinámica conocida en relación con sus entradas, salidas y restricciones temporales, para darle un correcto funcionamiento de acuerdo con los conceptos de predicción, estabilidad, control y alcance. Se pueden clasificar en *Soft* y *Hard Real Time*, dependiendo de las restricciones temporales impuestas por la aplicación en concreto y la criticidad del sistema [Walls 2006] [Heath 1997]. Ejemplos de sistemas operativos de tiempo real empotrados son MicroC/OS-II, VxWorks y LynxOS entre otros.

Debido a la gran variabilidad y heterogeneidad hardware de dispositivos que componen un sistema empotrado, las **comunicaciones** entre los

Sistema operativo	Arquitecturas HW-microprocesadores-microcontroladores	Tipo de dispositivo en el que se utiliza hasta la actualidad
Embedded Linux	ARM, AVR32, Blackfin, ETRAX CRIS, FR-V, H8300, M32R, M68k, MIPS, MN10300, PowerPC, SuperH, Xtensa	EMPOTRADOS
FreeBSD	AMD64, ARM, i386, IA-64, MIPS, PC98, PowerPC, SPARC64, sun4v, xbox	EMPOTRADOS
FreeRTOS	Nios II, Cortex M3, ARM, Coldfire, PIC, V850, 8051, x86	EMPOTRADOS
Inferno	Intel x86 (386 & higher), Intel XScale, IBM PowerPC, ARM StrongARM (ARM & Thumb), Sun SPARC	EMPOTRADOS, PDAs
LynxOS	Motorola 68010, Intel 80386, SVR3, ARM, MIPS y PowerPC.	EMPOTRADOS
uC/OS	Cortex, Nios II, ARM, TriCore, M68HC08/16, 80C16x, 80x86, OR12, PowerPC, Z80, eZ80	EMPOTRADOS
.NET Micro Framework	ARM architecture processors, on Analog Devices Blackfin	EMPOTRADOS
VxWorks	CPUs x86, MIPS, PowerPC, SH-4, ARM, StrongARM, xScale	EMPOTRADOS
RTLinux	x86, PPC.	EMPOTRADOS
µCLinux	MC68328, Intel i960, Motorola DragonBall	EMPOTRADOS
Windows CE	ARM, MIPS, SH, x86	EMPOTRADOS, PDAs
Palm OS	ARM	PDAs
Linux	Motorola 68k, Intel i386	PDAs
Internet Tablet OS (MAEMO)	OMAP	EMPOTRADOS, PDAs, MÓVILES, MEDIA
BlackBerry OS	ARM, Intel 80386, Intel PXA901, Qualcomm	BLACKBERRY
Android	Qualcomm, ARM, MIPS	EMPOTRADOS, MÓVILES, MEDIA, ULTRAPORTÁTILES
Openmoko Linux	Neo 1973, Neo FreeRunner.	MÓVILES

Tabla 2.1: Sistemas operativos empotrados

distintos elementos del mismo, y las comunicaciones con el exterior determinan uno de los ejes centrales. De manera general, las comunicaciones bus serie y paralelo han competido dentro de este ámbito y muchas propuestas se han definido para la optimización de las comunicaciones dentro de los sistemas empotrados. Todo ello ha derivado en la definición de diferentes protocolos de comunicación que enfatizan diferentes características dependientes de las aplicaciones del dominio para este tipo de sistemas, cubriendo determinadas debilidades. Ejemplos de protocolos de comunicación de amplio uso a nivel mundial serían las Serial Communication Interfaces (SCI), cuyo máximo exponente es el RS-232, RS-485, o el Universal Serial Bus (USB), entre otros.

El desarrollo de los sistemas empotrados se alimenta de una amplia gama de herramientas que buscan en la **simulación y emulación** la clave para el éxito del diseño y depuración y pruebas de los sistemas. Estas herramientas van a permitir, no sólo el diseño e implementación de este tipo de sistemas dentro de un entorno controlado, sino su depuración a nivel algorítmico y a nivel de sistema, para poder presentar un sistema libre de errores, validado y verificado. Depuradores, entornos como Matlab, Scilab/Scisos o compiladores específicos con diferentes funcionalidades disponibles, van a proveer un completo juego de herramientas a disponibilidad del programador y experto para el desarrollo de sus tareas. Entornos de Software de Fuentes Abiertas proporcionan una gran flexibilidad para adaptarse y cubrir las nuevas necesidades que demanda el mercado en el desarrollo de sistemas empotrados. En este punto nos encontramos con opciones claras como Linux, NetBSD, OSGI o Embedded Java.



El desarrollo de los sistemas empotrados se alimenta de una amplia gama de herramientas que buscan en la simulación y emulación la clave para el éxito del diseño y depuración y pruebas de los sistemas.

2.3

>> Conceptualización de Software de Fuentes Abiertas

La definición de Software de Fuentes Abiertas fue propuesta por la Open Source Initiative (OSI). Software de Fuentes Abiertas no sólo significa tener acceso al código fuente, sino que presenta una serie de implicaciones sobre la política de distribución entre otras. Las condiciones definidas por la OSI para considerar un Software como de Fuentes Abiertas son:

- Libre distribución: el software no tendrá limitaciones en cuanto a su distribución pudiendo ser ofrecido de manera gratuita o habiendo un lucro económico por ello.
- Código fuente: el código fuente debe estar incluido u obtenerse libremente lo que permitirá estudiar el código, modificarlo o adaptarlo a las necesidades propias.
- Trabajos derivados: las modificaciones realizadas sobre el código podrán distribuirse de manera libre.
- Integridad del código fuente del autor: existe la posibilidad que la licencia del software requiera que las modificaciones al código original sean distribuidas como parches de modo que queden bien diferenciadas del código del autor original.
- Sin discriminación de personas o grupos: debe reconocerse la autoría de todos aquellos que contribuyan a la generación del software.

- Sin discriminación de áreas de iniciativa: no se limitará el uso que se haga del software ya sea el uso de este con fines comerciales, militares, etc.
- La licencia debe ser tecnológicamente neutral: no debe requerirse la aceptación de la licencia por medio de ningún tipo de tecnología única, estilo o interfaz.
- Distribución de la licencia: deben aplicarse los mismos derechos a todo aquel que reciba el software.
- La licencia no debe ser específica de un producto: los derechos asociados a un programa no deben depender de si éste forma parte o no de una distribución de software mayor. Si un programa se extrae de una distribución dada y se usa, distribuye o modifica será bajo las condiciones de la licencia original.
- La licencia no debe restringir otro software: la licencia no debe obligar a que el software distribuido de manera conjunta posea la misma licencia y por ello sea de código abierto.

La definición de Software de Fuentes Abiertas propuesta por la Free Software Foundation, se basa en cuatro libertades básicas que cualquier programa considerado libre debe proporcionar libertad para:

- Utilizar el programa para cualquier propósito.
- Poder estudiar cómo funciona el programa. Implica acceso al código fuente del mismo.
- Redistribuir el programa.
- Hacer modificaciones y distribuir las mejoras. Implica también

acceso al código fuente del mismo.

Hay que remarcar que a pesar de compartir características comunes, no se deben confundir ambos términos. Dicha confusión viene derivada del hecho de que en la práctica ambas filosofías están muy próximas, resultando muy difícil su diferenciación. El Software de Fuentes Abiertas posee la libertad de copiar y redistribuir código de fuentes abiertas y toda mejora que se haga seguirá siendo Software de Fuentes Abiertas. Sin embargo bajo la filosofía *open source*, estas mejoras se pueden realizar pero la distribución no tiene por que ser libre.

En contraposición a la definición de software libre, o de código de fuente abierta, se encuentra el software cerrado, que es todo aquel software que no cumple alguna de las cuatro libertades anteriores.

Las licencias libres determinan que el código denominado como "código libre" es libre para su uso, copia, modificación y redistribución. Los principales beneficios debidos al uso de licencias libre estándar son:

- Ahorro de tiempo para el programador.
- Garantizar la compatibilidad del software bajo la misma licencia.
- Ofrece seguridad a los programadores y usuarios sobre su consistencia legal.

Las licencias de Software de Fuentes Abiertas se pueden agrupar en los siguientes tipos:

- Licencias robustas (Copyleft): el autor autoriza la modificación siempre que el nuevo trabajo derivado conserve las mismas libertades que el original. En este grupo se pueden encuadrar GPL, GFDL, etc.

- Licencias permisivas: el autor solicita que se mantenga su atribución. Las modificaciones se permiten, e incluso el trabajo derivado se puede establecer como cerrado. Dentro de este campo caen licencias como BSD, Apache, EPL, etc.

Dentro de la clasificación anterior, existen una serie de licencias que son importantes debido a su gran difusión a nivel mundial y extensiva utilización. Es el caso en primer lugar de la licencia GNU GPL (General Public License o licencia pública general). Su propósito es declarar que el software cubierto por esta licencia es Software de Fuentes Abiertas y protegerlo de intentos de apropiación que restrinjan esas libertades a los usuarios. La última actualización de esta licencia, la versión 3, aporta protecciones frente a demandas por infracción de patentes, frente a intentos de evitar el cumplimiento de la licencia mediante el uso de tecnologías DRM.

Como licencia permisiva, tenemos como ejemplo la LGPL (Lesser GPL), que se aplica a cualquier programa o trabajo que contenga una nota puesta por el propietario de los derechos del trabajo estableciendo que su trabajo puede ser distribuido bajo los términos de ésta. La principal diferencia entre la GPL y la LGPL radica en que la última puede enlazarse contra (en el caso de una librería 'ser utilizada por') un programa no-GPL, que puede ser Software de Fuentes Abiertas o software no.

El concepto de Hardware de fuentes abiertas es relativamente nuevo comparándolo con su análogo Software de Fuentes Abiertas, por este motivo, a la hora de licenciar este hardware, se ha dirigido la mirada principalmente a las licencias software, con una gran difusión y ya consolidadas.

De todas formas, se han propuesto nuevas licencias que contemplan términos específicos del diseño hardware. En estas nuevas licencias

portan muchos conceptos y principios de las licencias software al dominio hardware. En muchos casos, estas licencias hardware se explican y definen como las equivalentes a las licencias software reconocidas como GPL, LGPL, etc.

Muchas comunidades y proyectos tienden a adscribirse a una determinada licencia, de esta forma la comunidad OpenCores suele utilizar LGPL, FreeCores utiliza GPL y la Open Hardware Foundation prefiere *copyleft*.

Una diferencia de las licencias hardware es que se refieren a patentes en lugar de a derechos de autor (*copyright*). Los derechos de autor proporcionan a éste el control sobre la publicación, distribución y adaptación de su trabajo, abarca la expresión de un contenido, no la idea original. En cambio las patentes regulan el derecho a utilizar la idea. La definición de una patente implica su reconocimiento por parte de entidades reguladas, mientras que los derechos de autor son inherentes a la propia obra. De esta forma, los derechos de autor pueden controlar la distribución de código o diseños mientras que una patente controla el uso y fabricación de dispositivos utilizando un determinado diseño o tecnología.

Debido a las particularidades del desarrollo en hardware, se ha extendido el uso de licencias para hardware basadas en las ya existentes del Software de Fuentes Abiertas, sobre todo de las conocidas GPL y MIT. Sin embargo, estas licencias no ofrecen una cobertura específica para las necesidades particulares de los diseños y desarrollos hardware. Es por ello que en los últimos tiempos, han aparecido proyectos en los que se han intentado desarrollar licencias específicas para hardware. En este sentido, la mayoría de las licencias disponibles hoy día para hardware siguen el modelo GPL con restricciones específicas para hardware. A continuación se enumeran algunas de las licencias de hardware de

fuentes abiertas disponibles:

- TAPR Open Hardware License: similar a la licencia GPL garantiza la copia, distribución y modificación de documentación, así como la creación y uso de productos basadas en dicha documentación.
- Freedom CPU.
- MIT Open Source License.
- OpenIPCore Hardware General Public License "OHGPL".
- The Open NDA.
- OpenPPC (basada en Apple Public Source License).
- Simputer GPL.

La TAPR open Hardware Licence (OHL) es la más conocida. Está basada en GPL y garantiza la copia, distribución y modificación. La mayor restricción que impone es que la distribución del original o modificaciones de un código con licencia OHL tiene que ser aceptado por el receptor, no pudiendo ser negados los derechos adquiridos con OHL.

La Freedom CPU licence está restringida al dominio de los diseños en microelectrónica, y se centra en la cobertura del código fuente.

La OpenIPCore por su parte está asociada a diseños de FPGA. La mayor ventaja que ofrece es la posibilidad de ser utilizada conjuntamente con diseños propietarios. Sin embargo existen otras versiones de licencias que resultan ser más la definición de relaciones contractuales, como es el caso de Open NDA y Simputer GPL. Esta última es la más restrictiva de todas puesto que se trata de una mezcla de protecciones basadas en *copyright*, patente de marca y secreto comercial.

A continuación se presentan dos tablas, la primera establece las principales diferencias existentes entre las licencias de Software de Fuentes Abiertas más comunes (zonas a color indican características soportadas por las licencias). La segunda tabla especifica las compatibilidades entre las diferentes licencias.

	4 Libertades	Copyleft	Permite enlazar código cerrado	Protección DRM	Protección patentes
GPL					
GPLv3					
BSD					
LGPL		No aplica restricciones copyleft a software enlazado	Con restricciones		
Mozilla Public License					
Apache					
Eclipse Public license (EPL)					
European Union Public Licence (EUPL)		En caso de relicenciar		Posible en un futuro	

Tabla 2.2: Licencias Software de Fuentes Abiertas

Destino	Origen	GPL	GPLv3	BSD	LGPL	MPL	Apache	EPL	EUPL
GPL									
GPLv3									
BSD									
LGPL									
MPL									
Apache									

Tabla 2.3: Compatibilidad entre Licencias Software de Fuentes Abiertas

■ incompatible ■ compatible con restricciones ■ compatible

Otro aspecto fundamental a la hora de considerar el Software de Fuentes Abiertas es la definición de comunidades. Dentro de la terminología de Software de Fuentes Abiertas, el término comunidad se utiliza para referirse a usuarios y desarrolladores de dicho software en un sentido restringido, o en un sentido más amplio la comunidad es una población (de varios millones de individuos), con una demanda sociocultural de carácter técnico (el Software de Fuentes Abiertas) y unos recursos que utilizan para satisfacer, con mejor o peor acierto en el empeño, tal demanda [Segovia-Abella].

Aunque en los inicios estas comunidades estaban formada prácticamente en su totalidad por académicos y programadores, actualmente muchas empresas participan en ellas de forma activa, liderándolas en muchos casos. Ejemplos mundialmente reconocidos pueden ser las comunidades instauradas alrededor de soluciones software como Eclipse, Mozilla Firefox, Open Office o Apache HTTP Server.

Ejemplos de comunidades de Software de Fuentes Abiertas existentes en España:

Morfeo

La comunidad MORFEO persigue como objetivos:

- Acelerar el desarrollo de estándares software relacionados con Arquitecturas Orientadas a Servicios que son clave tanto en la integración de sistemas como en la evolución de la red como ecosistema en el que proliferen los servicios.
- Crear oportunidades de negocio en el ámbito e integración de soluciones dirigidas a empresas y a la Administración que estén basadas en plataformas y aplicaciones estándares desarrolladas en el marco de la comunidad.

- Mejorar la productividad y garantizar la calidad de los desarrollos ligados a proyectos software de código abierto que puedan integrarse dentro de la infraestructura de desarrollo software estándar en este tipo de proyectos (Gforge).
- Catalizar proyectos de I+D+i en el ámbito software que integren de manera natural a diversos agentes científicos y tecnológicos, contribuyendo al impulso de actividades de I+D+i y al desarrollo de un tejido industrial fuerte en los países donde los miembros del consorcio operan.

La comunidad MORFEO proporciona un marco abierto de colaboración entre empresas (grandes empresas y PYMEs), Universidades, centros de investigación y la Administración que confían en (o desean experimentar) las posibilidades que el software de código abierto ofrece de cara a la consecución de los objetivos antes indicados.

HispaLINUX

HispaLiNux es la asociación de usuarios españoles de Linux. Su objetivo es divulgar y facilitar el acceso al sistema Linux a los usuarios; así como coordinar, apoyar y dar organización a los distintos grupos no organizados que ya existen en España.

Para ello establece los siguientes fines:

- Divulgación y Promoción de la plataforma de software de libre distribución "Linux", en Español,
- Apoyo y organización a los grupos de usuarios y desarrolladores de Linux que funcionan actualmente en España sin una base organizativa.

- Se promocionarán también otros sistemas software de libre uso y distribución, para con ello dar a conocer al gran público esta particular manera de entender el software.

ASOLIF

ASOLIF está compuesta por las siguientes asociaciones empresariales:

- AGASOL: Asociación Gallega de Empresas de Software Libre
- CatPL: Cataluña Programari Lliure
- CESLA: Clúster de Entidades pro Software Libre de Aragón
- ESLE: Asociación de Empresas de Software Libre de Euskadi
- ESLIC: Asociación de Empresas de Software Libre de Canarias
- SoliMadrid: Asociación de Empresas de Software Libre de Madrid
- AndaLibre: Asociación de Empresas de Software Libre de Andalucía

Los principales objetivos son:

- Promocionar y divulgar el uso del Software de Fuentes Abiertas.
- Fomentar la investigación y el desarrollo de tecnologías basadas en Software de Fuentes Abiertas, apoyando así una mayor competitividad empresarial.
- Servir de vínculo e intermediario útil ante la AA.PP.
- Fomentar la promoción y colaboración de las asociaciones y

empresas de Software de Fuentes Abiertas.

- Colaborar con otras asociaciones o federaciones de intereses comunes a través del intercambio de experiencias en el campo del Software de Fuentes Abiertas.

2.4

>> Relación entre sistemas empotrados y Software de Fuentes Abiertas

Como se ha visto, los sistemas empotrados están presentes en muchos campos de la vida cotidiana. Con el tiempo, los requerimientos de este tipo de dispositivos han cambiado. Por un lado, la complejidad ha aumentado, al mismo tiempo que se exige una disminución en el tiempo de llegada al mercado para aumentar la ventaja competitiva en el sector de mercado de las empresas. Además, los costes de mantenimiento son altos, las funcionalidades limitadas y proveen de poca flexibilidad.

A todos estos aspectos se une el factor del **software cerrado**, altamente extendido en este dominio, y que impone unos costes de producción muy altos. Esto lleva a un mayor esfuerzo para evitar una ralentización en el desarrollo de software empotrado cerrado debido a los altos costes de inversión. El Software de Fuentes Abiertas tiene una oportunidad en este mercado para impulsarse como alternativa para la inversión en innovación.

En este sentido, se empiezan a ver casos de alianzas entre diferentes empresas dentro del mismo sector para impulsar la innovación mediante Software de Fuentes Abiertas. Un ejemplo puede ser la fundación de OSADL (Open Source Automation Development Lab) [OSADL], organización formada inicialmente por once empresas del sector automovilístico alemán, pero abierta a la inclusión de empresas de otros países. Las empresas que conforman la alianza incluyen empresas fabricantes de hardware, vendedores de software, empresas de servicios open source, etc.

Esta organización tiene como objetivo el desarrollo de un sistema operativo en tiempo real basado en Linux que satisfaga las necesidades de la industria automovilística. Conjuntamente se pretende el desarrollo de un entorno de desarrollo que favorezca la implementación del software y la mejora de la interoperabilidad. El grupo pretende que el trabajo desarrollado dentro de la fundación permita la elaboración de estándares y niveles de calidad que permitan nuevas certificaciones de interoperabilidad entre las plataformas HW y los productos SW.

A día de hoy, el software de fuente abierta se puede empezar a ver como una realidad en el mundo de los sistemas empotrados. Los sistemas empotrados y pequeños dispositivos cada vez están más presentes en nuestras vidas, facilitando aspectos básicos de la vida actual como las comunicaciones o el transporte en base a soluciones software integradas en plataformas hardware de propósito específico. Desde el software de fuente abierta en dispositivos, proyectos como Maemo, Open-Moko, Android/Java etc. van mostrando cómo el Software de Fuentes Abiertas puede ser una gran plataforma para los pequeños dispositivos.

Existen diferentes iniciativas más específicas y concretas que contribuyen a abrir campos de desarrollo a programadores que en caso contrario estarían delimitados en un entorno mucho más específico. Android se puede considerar el caso más ejemplar, la licencia de software Apache proporcionada

por la Open Handset Alliance [OHA] ha permitido a infinidad de programadores implementar numerosas aplicaciones que enriquecen la plataforma.

La lógica de negocio en el ámbito de los sistemas empotrados presenta en la actualidad un alto índice de complejidad, aproximándose al modelo denominado "red de valor añadido" o "value network", en la que los proveedores colaboran también entre sí y con el integrador de sistemas para lograr el producto final. Sin embargo, en determinadas etapas podríamos aún encontrar un modelo más parecido a la cadena de valor jerárquica, en el sentido que el integrador de sistemas (OEM) es el que provee al comerciante, siendo este último el que finalmente llega al cliente final.

El primer planteamiento, se ajusta de forma idónea a la forma de negocio subyacente en el código abierto y su creación de valor; no obstante, en esquemas jerárquicos, es valorable el uso de Software de Fuentes Abiertas y se debe analizar su potencial para determinar los nichos concretos de negocio.

En estudios realizados en EE.UU [Nass 2008], se ha comprobado cómo la reutilización del código es una práctica común en el desarrollo de sistemas empotrados, aunque con una fuerte incidencia en la reutilización de código propio. El código de fuentes abiertas de momento es una opción minoritaria, muchas veces incluso frente al desarrollo de código nuevo, pero por las características que ofrece puede ser una gran opción de futuro.

El uso de opciones de código de fuentes abiertas produce una mayor eficiencia en el TTM (time to market). Del mismo modo, el soporte de lenguajes y formalismos de modelado es ampliamente soportado por este tipo de códigos. El código de fuente abierto permite la reducción de las barreras entre proyectos multiplataforma, sobre todo con aquellos relacionados con temática de tiempo real.

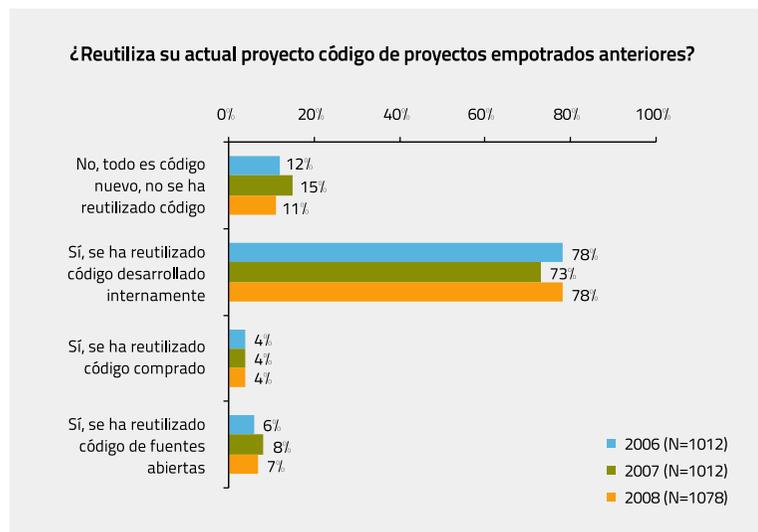


Figura 2.3: Estudio sobre sistemas empotrados en EE.UU.
Reutilización de código en proyectos de sistemas empotrados [Nass 2008]

La industria electrónica ha llegado a un punto en el que la competitividad del mercado obliga a que muchos productos requieran el uso de soluciones ASICs o SoC (*System on Chip*), pero su elevado precio hoy en día dificulta el acceso a los mismos por parte de pequeñas compañías. Esta situación resulta paradójica cuando muchas propiedades de los SoC se ajustan perfectamente a las características de las pequeñas compañías o empresas que comienzan.

Dentro de las principales causas del elevado precio de estos dispositivos se pueden encontrar el precio de la propiedad intelectual (IP) y el riesgo asociado al desarrollo de chips particularizados. El uso de IP de fuentes abiertas puede mejorar enormemente esta situación dando a las pequeñas compañías acceso a propiedad intelectual que se puede implementar en SoCs, como CPUs, controladores Ethernet, etc., todo ello con unos costes asumibles.



El mercado actúa como elemento regulador y para garantizar el éxito de los proyectos, los desarrolladores tienen que no perder de vista al usuario final.

2.5

>> Plan del informe

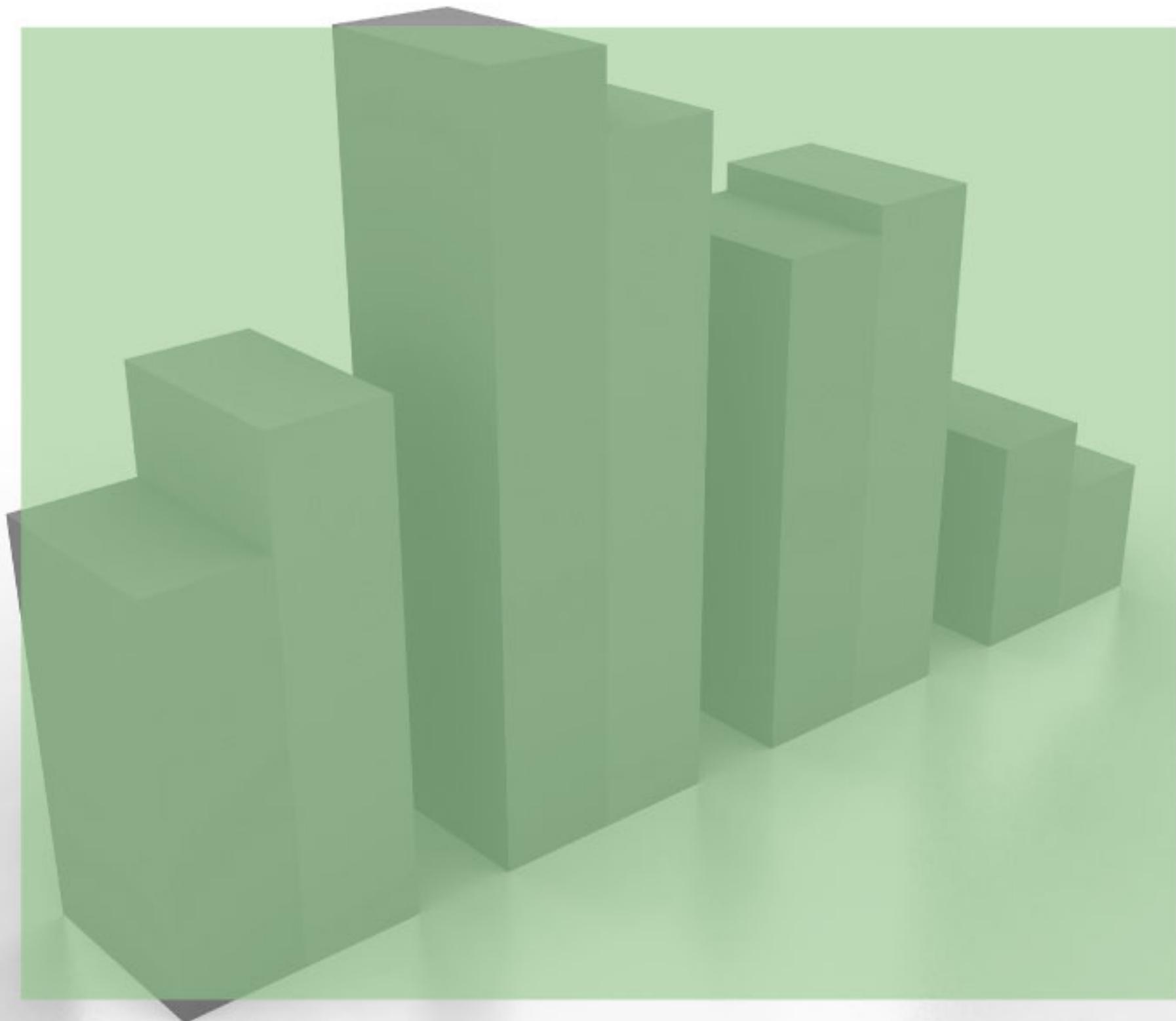
El estudio que se ha realizado pretende proporcionar una visión sobre la importancia que tiene actualmente el Software de Fuentes Abiertas en los sectores tecnológicos del diseño, la fabricación y comercialización de sistemas empotrados. Estos sectores tecnológicos están ampliamente implantados en la industria española pero históricamente no se han asociado al Software de Fuentes Abiertas.

Este informe describe la situación real, los beneficios obtenidos así como las oportunidades de negocio que se puedan establecer con la utilización de Software de Fuentes Abiertas dentro de los diferentes sectores que desarrollan sistemas empotrados a partir de información proporcionada por empresas situadas en diferentes sectores industriales, todas ellas enmarcadas dentro de los sistemas empotrados.

Al mismo tiempo, se ha analizado la problemática empresarial de la migración de sistemas empotrados basados en software cerrado a Software de Fuentes Abiertas, o la implementación de nuevas líneas de producto de sistemas empotrados con Software de Fuentes Abiertas, introduciendo un salto tecnológico en el desarrollo de los productos.

El contenido de este informe está estructurado en los siguientes capítulos:

- 1.** **Resumen ejecutivo:** Contiene un breve análisis de los aspectos más importantes del estudio y las conclusiones generales que se han derivado del mismo.
- 2.** **Introducción y plan de informe:** En este capítulo se presenta el contexto en el que se ha definido el estudio, así como la estructura y contenidos del informe.
- 3.** **Metodología de la investigación:** Este capítulo describe la metodología empleada para realizar el estudio.
- 4.** **Situación del Software de Fuentes Abiertas en los Sistemas Empotrados:** Presenta la visión obtenida en el estudio mediante un análisis de la situación del Software de Fuentes Abiertas en las empresas entrevistadas. Se detalla la visión de las características estructurales de las empresas que utilizan Software de Fuentes Abiertas en alguna fase del desarrollo de sus productos, un portafolio tecnológico (hardware y software), actividades I+D, comercialización y distribución, recursos humanos y colaboración con comunidades de desarrollo de Software de Fuentes Abiertas.
- 5.** **Análisis estratégico:** Detalla las dificultades, beneficios, tendencias de la utilización del Software de Fuentes Abiertas dentro de los sistemas empotrados, así como un análisis de las debilidades, amenazas, fortalezas y oportunidades en este campo.
- 6.** **Conclusiones y recomendaciones:** Principales conclusiones del estudio, así como recomendaciones y posibles estrategias frente al análisis DAFO realizado.
- 7.** **Equipo de trabajo:** Descripción del equipo de trabajo y del panel de expertos.
- 8.** **Bibliografía y referencias**
- 9.** **Anexo:** Presenta breves referencias y descripciones de las diferentes soluciones de software y hardware de fuentes abiertas que se han detectado durante la realización del informe.



03

METODOLOGÍA DE LA INVESTIGACIÓN

Este capítulo del informe describe la metodología empleada en la elaboración del estudio, indicando qué población y muestra se ha tomado para la investigación, así como la forma en la que se han recogido los datos.

03

METODOLOGÍA DE LA INVESTIGACIÓN

3.1

>> Introducción

Este capítulo del informe describe la metodología empleada en la elaboración del estudio, indicando qué población y muestra se ha tomado para la investigación, así como la forma en la que se han recogido los datos.



Incluso cuando se enmarcan dentro de un mismo sector de actividad, el objetivo o la aplicación final de los sistemas empotrados puede ser completamente diferente.

3.2

>> Diseño de la muestra y trabajo de campo

Como se ha indicado en el capítulo 2 de este informe, los sistemas empotrados se encuentran en prácticamente la totalidad de sectores de actividades, incluso cuando se enmarcan dentro de un mismo sector de actividad, el objetivo o la aplicación final de los sistemas empotrados puede ser completamente diferente.

Esta atomización implica una seria dificultad en la elaboración de un censo de empresas que desarrollan sistemas empotrados. Los diferentes grados de complejidad de dichos sistemas empotrados y los avances tecnológicos que aumentan las capacidades de computación y procesamiento presentan otra dificultad a la hora de determinar el límite entre lo que hasta ahora se ha conocido como sistema empotrado y computadora personal. Estas características han contribuido a que no se reconozca o no exista un censo de empresas de sistemas empotrados oficial, bajo el amparo de una autoridad competente.

Otra dificultad añadida en la definición de dicho censo es el número de actores que intervienen en el desarrollo de sistemas empotrados. En muchos casos, todavía se plantea una única empresa encargada del diseño, implementación hardware, software, fabricación, integración, testes y comercialización de un sistema empotrado completo, pero la creciente complejidad de los mismos está favoreciendo la aparición de

empresas dedicadas exclusivamente a las diferentes fases del desarrollo de los sistemas empotrados. Esta característica plantea una disyuntiva sobre qué empresas incluir en un potencial censo de empresas que fabriquen sistemas empotrados, aquellas que realicen todo el proceso, aquellas que intervengan en alguna fase del desarrollo del mismo, etc.

Finalmente, considerando cómo el esfuerzo empresarial se va dirigiendo también en gran medida en la generación de nuevas actividades focalizadas en diferentes fases de la generación de un sistema completo, se han tenido en cuenta todo tipo de empresas en la elaboración de este informe. Se han considerado tanto empresas que desarrollan sistemas empotrados completos, como aquellas proveedoras de componentes de sistemas más complejos o empresas proveedoras de herramientas para el desarrollo de sistemas empotrados.

Debido a la dificultad en la elaboración de un censo completo, por la heterogeneidad de las empresas que tienen su actividad en el objetivo de este informe, se ha decidido tomar como referencia la plataforma tecnológica nacional de sistemas con inteligencia integradas, PROMETEO, nacida en 2005 y que agrupa un amplio espectro de empresas activas en el sector multisectorial de los sistemas integrados.

PROMETEO es la Plataforma Tecnológica Española en el área de los Sistemas con Inteligencia Integrada (Embedded Systems) y constituye una red de cooperación científico-tecnológica integrada por los agentes tecnológicos relevantes del área (Empresas, Universidades, Centros Tecnológicos, etc.).



España muestra una capacidad de liderazgo elevada dentro de este ámbito, tal y como ha sido constatado en el ámbito de las plataformas tecnológicas PROMETEO y la europea ARTEMIS. Como muestra de ello basta destacar la elevada participación de empresas y agentes españoles (número 1 en Europa) en la segunda convocatoria de proyectos de la Joint Technology Initiative ARTEMIS. Estas plataformas están conformadas con la participación de las principales empresas europeas que cubren el amplio espectro y variabilidad y heterogeneidad presente en este sector.



Artemisia es la **Asociación Industrial Artemis** que representa no sólo a la comunidad investigadora sino también a la industria (tanto gran industria, como pymes), así como a universidades e institutos de investigación a nivel europeo. Dentro de los objetivos de esta plataforma se puede distinguir: la mejora de la seguridad en sectores de transporte e industrial y la prevención, cuidado y bienestar mediante el uso de sistemas empotrados, y otros referentes a la productividad y competitividad industrial, mediante la reducción de costes/ tiempo de puesta en mercado y aumento de la calidad final. Estos objetivos son abordados mediante la definición de nuevas arquitecturas, paradigmas de diseño y tecnologías transversales de aplicación en los dominios de aplicación de los sistemas empotrados, etc.

Existen otras muchas plataformas que cuentan entre sus miembros empresas dedicadas a la fabricación de componentes o sistemas empotrados completos, pero el objetivo de las mismas no son los sistemas empotrados, sino que están focalizadas hacia sectores de actividad comercial definidos como la automoción (SERtec), software y servicios (INES), del hogar digital (PTHD), etc.

PROMETEO es la única plataforma española que engloba a empresas, centros de investigación, universidades y fundaciones que tienen dentro de su actividad el desarrollo de sistemas empotrados, ya sea con el objetivo de la fabricación de un sistema completo, componentes o herramientas. Es por ello que se ha seleccionado PROMETEO como referencia para definir el diseño de la muestra para este estudio.

En el momento de elaborar este informe la plataforma PROMETEO consta de 153 miembros oficiales, de los cuales 91 son empresas (59,4%), tanto grandes empresas, como pymes y micro empresas. A este número de empresas hay que añadir otras 10 empresas que están en proceso de admisión o en contactos con la plataforma. El resto de miembros no se han considerado para este estudio por ser centros tecnológicos, universidades, fundaciones, etc.

Para la elaboración del presente estudio, se ha contactado con las 101 empresas de la plataforma (empresas miembros y empresas en proceso de admisión o en contacto) recibiendo contestación de 47 empresas, lo que implica una tasa de respuesta del 45,19%, con un margen de error de +/- 10,6% para una situación de máxima indeterminación ($p=q=0,5$) y un nivel de confianza del 95%.

Miembros de PROMETEO	153	
Empresas de PROMETEO	91	
	Contactadas	Participan en el estudio
Empresas de PROMETEO	91	37
Empresas en contacto con PROMETEO	10	10
Total de empresas	101	47

Tabla 3.1: Empresas contactadas

A todas las empresas contactadas se les ha pedido que completen un cuestionario vía web, telefónicamente o en papel. Dicho cuestionario indaga sobre determinadas características de la empresa y sobre la utilización de Software de Fuentes Abiertas en las diferentes fases de desarrollo de los productos de la empresa. La respuesta mayoritaria ha sido vía web puesto que ha sido la más adaptable a las necesidades de tiempo de los participantes.

Las preguntas han sido de varios tipos:

- Preguntas cerradas con opción de respuesta de selección única

Figura 3.1: Pregunta cerrada selección única

- Preguntas cerradas con opción de respuesta de selección múltiple

Figura 3.2: Pregunta cerrada selección múltiple

- Preguntas semi-abiertas. Se proporcionan diversas opciones de respuesta múltiple y un apartado para comentarios

Figura 3.3: Pregunta semi-abierta

- Preguntas abiertas

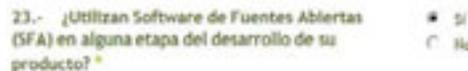


Figura 3.4: Pregunta abierta

En todas ellas se proporcionaba una ayuda explicando lo que se pretendía obtener de ellas.

Las tres primeras partes de la encuesta estaba dirigida a obtener información básica sobre la persona que ha cumplimentado la encuesta, su rol en la empresa y sobre las características estructurales de la misma.

A continuación se les pregunta por la utilización del Software de Fuentes Abiertas en alguna fase del desarrollo de su producto y en caso afirmativo se accede a las preguntas referentes al Software de Fuentes Abiertas. En caso contrario, se finaliza la encuesta preguntando si se ha estudiado la utilización de este tipo de software, en qué fases y por qué causas se ha desestimado.



En el caso de contestar afirmativamente, se accede a una serie de preguntas sobre la utilización del software, hardware y plataformas de fuentes abiertas, así como los beneficios, dificultades y posibles tendencias del Software de Fuentes Abiertas.

La elaboración de las preguntas que se han remitido a las empresas se han elaborado en base a determinadas variables definidas por el equipo que ha elaborado el estudio (equipo formado por ESI y AICIA en

colaboración con CENATIC). Preguntas y variables han sido revisadas por el panel de expertos, teniéndose en cuenta sus sugerencias para realizar la versión final de las variables y de las preguntas que han formado el cuestionario.

Los datos obtenidos a través de la encuesta son los siguientes:

- **Características estructurales de la empresa**
 - Por Comunidad Autónoma
 - Por sector de actividad comercial
 - Por tecnología
 - Por tamaño
- **Utilización de Hardware de fuentes abiertas**
 - Herramientas Hardware de fuentes abiertas
 - Microprocesadores/microcontroladores de fuentes abiertas
 - Diseño Hardware de fuentes abiertas
- **Utilización de Software de Fuentes Abiertas**
 - Aplicaciones
 - Sistemas operativos
 - Librerías
 - Licencias
 - Entornos de programación
 - Utilización de herramientas de integración de fuentes abiertas
- **Actividades de I+D**
- **Comercialización y distribución**
 - Tipología de clientes
 - Valoración del Software de Fuentes Abiertas como seña de identidad de la empresa

- Publicidad

- **Recursos humanos**

- Formación

- Experiencia

- **Colaboración con comunidades de desarrollo**

En la variable "sector de actividad comercial" se han agrupado parte de las categorías de las actividades económicas CNAE 2009 puesto que la primera clasificación aporta una información muy genérica, y el siguiente nivel de clasificación es muy extenso y puede desvirtuar la información obtenida. Se ha tomado como referencia el **"Plan Estratégico para el impulso de proyectos tecnológico-empresariales innovadores de la industria TIC en el ámbito de los dispositivos electrónicos, los sistemas empotrados y los componentes electrónicos"** encargado por Sandetel (Sociedad Andaluza para el desarrollo de las Telecomunicaciones S.A.).

Las categorías de las actividades económicas agrupadas son las siguientes: agroalimentario, control industrial, aeronáutica, naval, bienes de consumo, infraestructuras, turismo, construcción, logística, espacio, defensa y seguimiento, medioambiente, automoción, energía, salud, ferroviario, transporte y una categoría varios, que englobaría a aquellas empresas que no puedan catalogar su actividad comercial en los sectores anteriormente comentados. Las empresas seleccionadas para la muestra de este estudio, se han clasificado en base a estas categorías para poder definir y establecer la red de valor, los actores y la interacción de los mismos.

3.3

>> Modelo muestral

A continuación se presenta una descripción de la muestra obtenida. Se analizarán las siguientes variables:

- Ámbito Geográfico
- Sector de actividad comercial
- Tecnología
- Tamaño de empresa

Los datos que se muestran en este apartado describen las características de las empresas que han participado en el presente estudio.

3.3.1

>> Distribución geográfica

Para analizar la distribución geográfica de la muestra obtenida en el estudio, se ha estudiado las respuestas sobre la localización de las oficinas centrales de la empresa.

Como se puede apreciar en la Figura 3.5 las Comunidades Autónomas que se han visto más representadas en el estudio son la Comunidad Autónoma de Madrid, con un 33% de las empresas que han participado en el estudio y Euskadi con un 27%. Les sigue Cataluña con un 13%, Andalucía con un 10% y Galicia con un 8%.

El resto de Comunidades Autónomas que se encuentran representadas en el estudio, con un porcentaje inferior son la Comunidad Valencia (4%), Murcia (2%) y Asturias (2%).

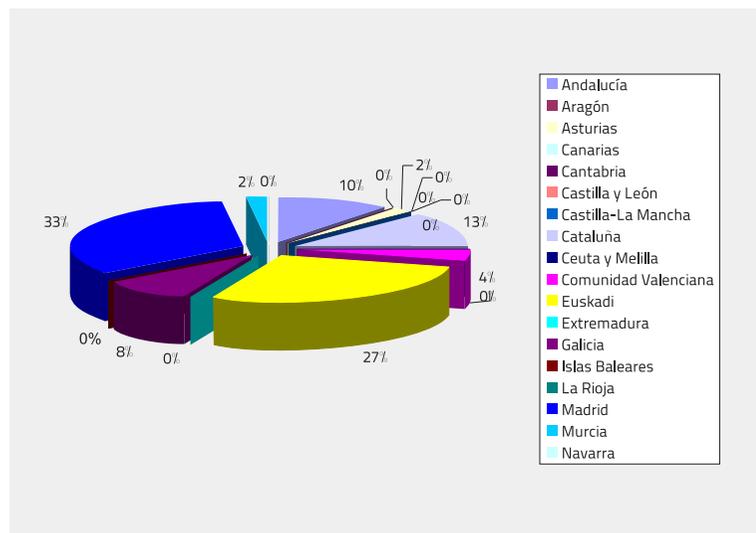


Figura 3.5: Distribución geográfica

Estos resultados indican una importante participación en el estudio de dos Comunidades en la que existe un fuerte tejido empresarial dirigido a los sistemas empotrados como Madrid y Euskadi, fortaleza empresarial que también se da en Cataluña y Andalucía.

Otro factor que influye en el alto porcentaje de Madrid, es el que muchas empresas tengan localizada su razón social en la capital del país, aunque tengan oficinas repartidas por todo el estado.

3.3.2 >> Distribución por sector de actividad

Se solicitó a las empresas participantes en el estudio que definieran los sectores de actividad en los que ellos mismos se engloban, dándoles la opción de definir todos los sectores que creyeran convenientes. Los

porcentajes que se indican en la Figura 3.6 son el tanto por ciento de empresas que han señalado dicho sector de actividad como concerniente a su empresa.

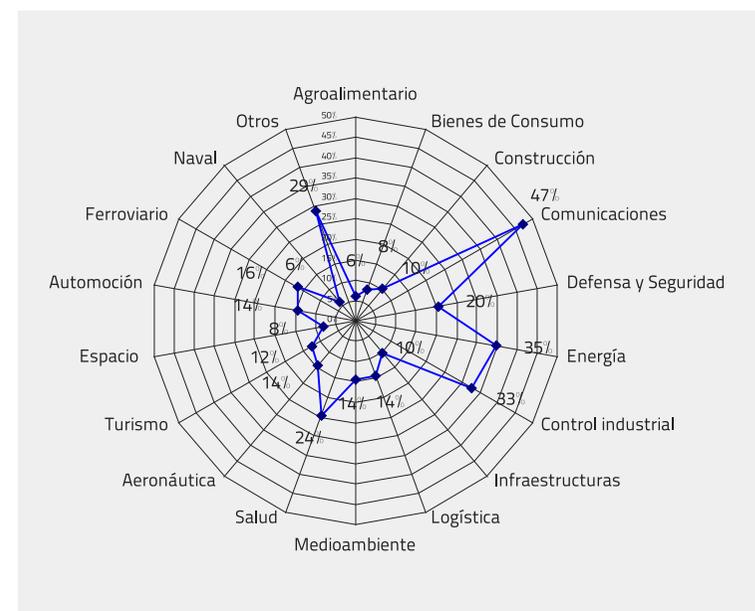


Figura 3.6: Distribución por sector de actividad

Como se aprecia en la Figura 3.6 en mayor o menor medida se ha logrado una representación de todos los sectores definidos. El sector de actividad en el que participa el mayor número de empresas es el sector de Comunicaciones con un 47% de empresas que indican que trabajan en dicho sector, aunque no sea el único en el que se defina su actividad comercial. Los siguientes sectores en los que un mayor número de empresas han definido su actividad es el sector de la Energía, con un 35% de empresas y Control Industrial con un 33%.

Les siguen los sectores Otros y Salud con un 29% y un 24%. A continuación hay una serie de sectores con una representatividad similar, entre un 10% y un 16%, son los sectores Ferroviario (16%), Automoción (14%), Logística (14%), Medioambiente (14%), Aeronáutica (14%), Turismo (12%), Construcción (10%), e Infraestructuras (10%).

Los sectores que tienen una menor representación entre las empresas participantes en el estudio son Bienes de Consumo (8%), Espacio (8%), Agroalimentación (6%), y Naval (6%) con un porcentaje menor del 10%.

El porcentaje logrado por las empresas del sector de las Comunicaciones es previsible puesto que las comunicaciones son una parte integrante de gran parte de los sistemas empotrados, además de la existencia de gran parte de sistemas empotrados dedicados exclusivamente a las comunicaciones.

El alto nivel de penetración de los sistemas empotrados en los diferentes sectores de actividad es en gran parte debido a las facilidades de conexión e intercambio de datos entre dispositivos, así como con aplicaciones de gestión. El acceso de sistemas empotrados a redes como Internet, proporciona nuevas funcionalidades y posibilidades de acceso a la información desde la movilidad, lo que no resultaba fácilmente accesible con anterioridad, y que hace que este sector esté en continua expansión.

Los sectores de Energía y Control Industrial tienen una larga experiencia en la inclusión de sistemas empotrados en diferentes partes de su actividad comercial. Son sectores en los que los sistemas empotrados presentan una implantación consolidada. Esta presencia consolidada se advierte también en los sectores Ferroviario, Automoción, Logística, Aeronáutica y Medioambiente. En todos estos sectores, los sistemas empotrados tienen todavía un largo recorrido de expansión mediante la innovación y el asentamiento de nuevas tecnologías.

3.3.3 >> Distribución por tecnología

En el análisis de la distribución de la muestra por tecnologías se ha tenido en cuenta la información proporcionada por las propias empresas referentes a las tecnologías que utilizan para el desarrollo de su producto.

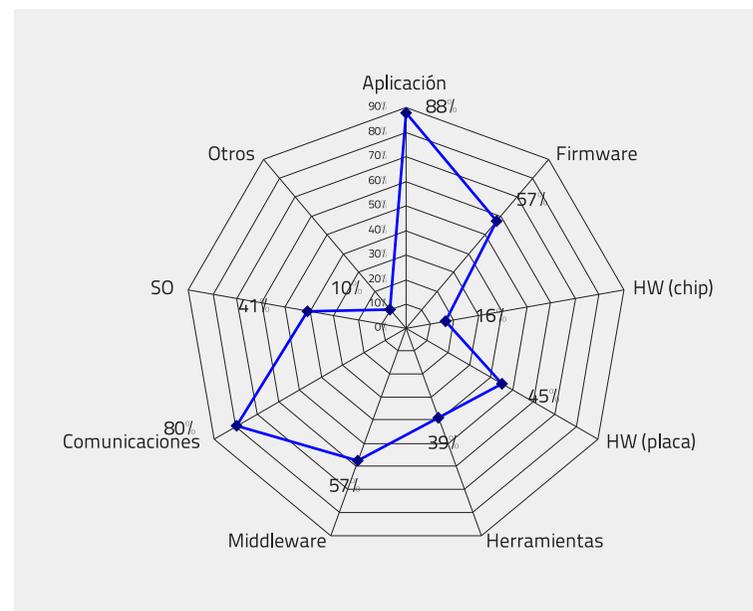


Figura 3.7: Distribución por Tecnologías

Los porcentajes de la Figura 3.7 indican el tanto por ciento de empresas que han seleccionado cada tecnología como base de su actividad. Cada empresa ha podido seleccionar más de una tecnología.

Como se aprecia en la Figura 3.7 las tecnologías más utilizadas por las empresas participantes en el estudio son: en primer lugar, Aplicación con un 88% de empresas que han definido que trabajan en ese ámbito.

Le sigue Comunicaciones con un 80%, siendo estas dos tecnologías las que disponen de más amplia cobertura por parte de las empresas participantes.

El 57% de las empresas participantes han definido que en el desarrollo de sus productos trabajan realizando Firmware y Middleware. Las siguientes tecnologías más empleadas son Hardware (placa) (45%) y Sistemas Operativos (41%).

El porcentaje de empresas participantes cuya actividad es la implementación de herramientas para el desarrollo de sistemas empujados se sitúa en un 39%.

Finalmente, solo un 16% de empresas han declarado que trabajan con Hardware a nivel de chip.

Prácticamente la totalidad de sistemas empujados utilizan la tecnología de Aplicación, es en lo que basan su hecho diferencial, sistemas contruidos con un propósito específico, normalmente definido en la aplicación. Como se ha comentado antes, las Comunicaciones se han convertido en un componente esencial de los sistemas empujados.

Gracias a los avances tecnológicos, se ha evolucionado de un tipo de empresas que realizaban todas las fases de desarrollo de sistemas empujados, desde el Hardware, Firmware, Comunicaciones y Aplicación, a empresas especializadas en cada una de estas mismas tecnologías. El aumento de capacidad de procesamiento que se ha dado en los sistemas empujados ha permitido pasar de unos Sistemas Operativos definidos en el propio Firmware o Aplicación, a la inclusión de Sistemas Operativos específicos para sistemas empujados, lo que ha dado pie a que ésta sea una tecnología que empieza a ser tenida en cuenta por una gran cantidad de empresas.

3.3.4

>> Distribución por tamaño de empresa

El análisis de la distribución por tamaño de empresa se ha realizado mediante las respuestas obtenidas a la categorización que las empresas han realizado al participar en el estudio. Se les ha preguntado por su tamaño definido por el número de empleados y por la facturación.

Las opciones que disponían eran las siguientes

- **Número de empleados**
 - Micro-empresa < 10 personas empleadas
 - Pyme-Pequeña empresa < 50 personas empleadas
 - Pyme-mediana empresa < 250 personas empleadas
 - Gran Empresa > 250 personas empleadas
- **Facturación**
 - Micro < 2M€ facturados
 - Pyme-Pequeña empresa < 10M€ facturados
 - Pyme-mediana empresa < 50M€ facturados
 - Gran Empresa > 50M€ facturados



Gracias a los avances tecnológicos, se ha evolucionado de un tipo de empresas que realizaban todas las fases de desarrollo de sistemas empujados, desde el Hardware, Firmware, Comunicaciones y Aplicación, a empresas especializadas en cada una de estas mismas tecnologías.

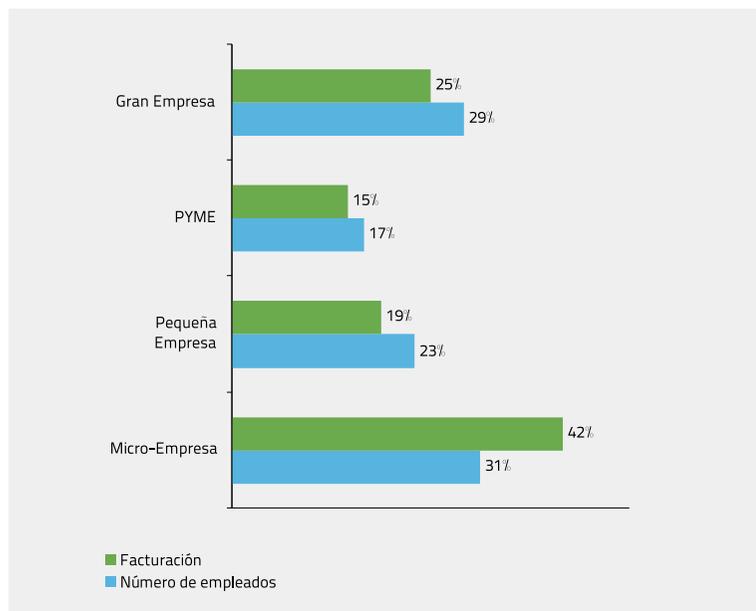


Figura 3.8: Distribución de empresas por tamaño

En la Figura 3.8 se ha realizado una comparativa de ambas variables: número de empleados y facturación. De todas las empresas que han participado en el estudio, un 31% son micro-empresas si se tiene en cuenta el número de empleados, este porcentaje sube hasta un 42% si se atiende a la facturación.

El segundo bloque nos indica el porcentaje de empresas que se definen como pyme-pequeña empresa. Según el número de empleados el porcentaje asciende a un 23% y según la facturación a un 19%.

El tercer bloque de la gráfica indica el porcentaje de empresas que se declaran pyme-mediana empresa. Según el número de empleados, un 17% y según la facturación un 15%.

El cuarto bloque, finalmente, indica el porcentaje de grandes empresas que han participado en el estudio y que se sitúa en el 29% según el número de empleados, mientras que este porcentaje desciende a un 25% si se mide según la facturación de las empresas.

Es de destacar cómo dentro de las micro-empresas se da una alta discordancia entre el número de empleados y la facturación, siendo el porcentaje por facturación superior al porcentaje por número de empleados en un 11%. Esto indica la existencia de empresas con los mismos recursos que otras micro-empresas pero que no logran entrar en el mismo rango de facturación que ellas.

Este deslizamiento entre los porcentajes según el número de empleados y la facturación se va relativizando según se aumentan los recursos de las empresas. Esta discordancia puede deberse al tipo de producto. Mayor complejidad del producto normalmente deriva en unas cantidades de facturación mayores, las grandes empresas afrontan estos retos con una mayor capacidad que las medianas y pequeñas empresas. Esta característica hace que los porcentajes por facturación de las pymes, pequeña empresa y grandes empresas sean menores que los de números de empleados, excepto en la categoría de menor rango.





04

SITUACIÓN DEL SOFTWARE DE FUENTES ABIERTAS EN LOS SISTEMAS EMPOTRADOS

En este capítulo del informe se realiza el análisis de diferentes características de las empresas participantes en el estudio que han declarado que utilizan Software de Fuentes Abiertas en alguna fase del desarrollo de sus productos. Todos los datos que se mencionan en el presente capítulo están referidos exclusivamente a dichas empresas. Se describirá la situación del Software de Fuentes Abiertas en sistemas empotrados atendiendo a las diferentes variables de análisis definidas en el capítulo 3, Metodología de la investigación.

04

SITUACIÓN DEL SOFTWARE DE FUENTES ABIERTAS

4.1

>>Características estructurales de las empresas que trabajan en sistemas empotrados basados en Software de Fuentes Abiertas

En esta sección se van a analizar las características estructurales de las empresas que han participado en el estudio y han declarado utilizar Software de Fuentes Abiertas en alguna fase del desarrollo de sus productos.

Se atenderán las siguientes variables para el estudio de las características estructurales:

- Distribución de empresas por Comunidades Autónomas
- Distribución de empresas por tamaño
- Distribución de empresas por sector de actividad
- Distribución de empresas por tecnología
- Distribución de empresas según la cadena de valor

4.1.1

>> Porcentaje de empresas que trabajan sistemas empotrados basados en Software de Fuentes Abiertas

La Figura 4.2 indica el porcentaje de empresas que han participado en el estudio y que han respondido "Sí" a la pregunta:

23.- ¿Utilizan Software de Fuentes Abiertas (SFA) en alguna etapa del desarrollo de su producto? Sí No

Figura 4.1: Pregunta: Uso de Software de Fuentes Abiertas en alguna etapa del desarrollo

Como se aprecia en la Figura 4.2 el 90% de las empresas participantes han declarado utilizar Software de Fuentes Abiertas en alguna etapa del desarrollo de su producto. Este porcentaje incluye a empresas cuyo producto final contiene software o hardware de fuentes abiertas, como las empresas que utilizan Software de Fuentes Abiertas en el diseño, implementación, fabricación, mantenimiento, configuración o cualquier otra etapa definida en su conjunto de procesos que concluyen en la elaboración de su producto. Existen aplicaciones y herramientas de fuentes abiertas que se utilizan en fases igualmente decisivas a la hora de elaborar un sistema empotrado, como entornos de aplicación, depuradores, compiladores, gestores de documentación, gestores de versiones, etc.

El porcentaje obtenido puede parecer en un principio inusualmente alto, pero es necesario reseñar que en la elaboración de componentes de sistemas empotrados, o sistemas empotrados completos, se definen muchas etapas en las cuales es posible la inclusión de herramientas basadas en Software de Fuentes Abiertas.

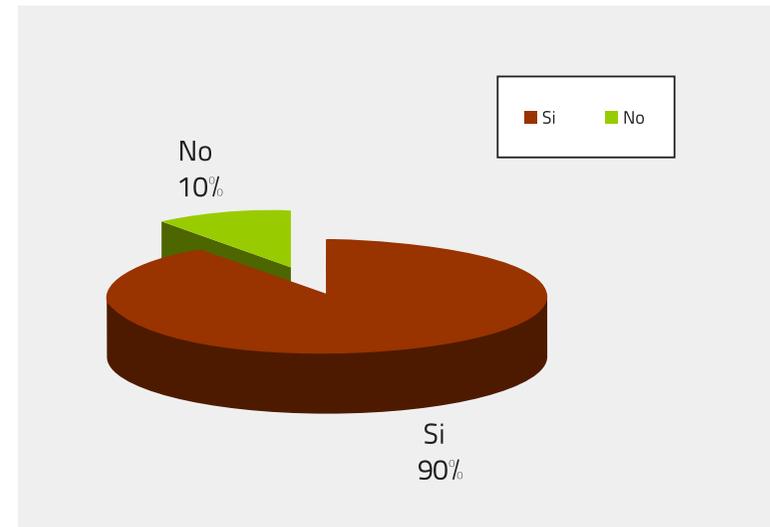


Figura 4.2: Distribución de Empresas de Sistemas Empotrados que utilizan Software de Fuentes Abiertas en alguna fase de su desarrollo

Como se ha definido en el capítulo 2, la oferta comercial de Software de Fuentes Abiertas no excluye otras estrategias. Así, existen muchas empresas dedicadas al software de sistemas empotrados que al mismo tiempo que venden entornos de programación, compiladores y demás herramientas, proporcionan software incluyendo el código, opciones de modificado, en cualquier tipo de tecnología, etc. Son muchas las licencias admitidas dentro de las fuentes abiertas, lo que permite englobar a una gran cantidad de herramientas, software y aplicaciones, no limitando exclusivamente el Software de Fuentes Abiertas al software completo incluido en los sistemas empotrados.

4.1.2 >>Distribución de las empresas por CC.AA

Esta variable se ha analizado mediante las respuestas que han dado las empresas participantes en el estudio que utilizan Software de Fuentes Abiertas a la localización de sus oficinas principales. Las empresas sólo han podido seleccionar una Comunidad Autónoma.

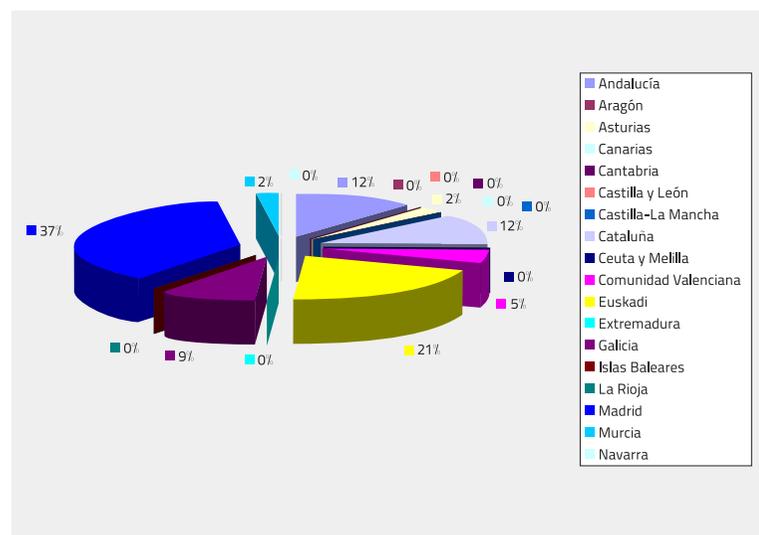


Figura 4.3: Distribución geográfica de las empresas

Como se aprecia en la Figura 4.3 la distribución de las empresas que utilizan Software de Fuentes Abiertas es muy similar a la definida en el apartado 3.3.1 Distribución geográfica correspondiente a todas las empresas participantes. La Comunidad Autónoma con mayor representatividad es Madrid (37%), seguida de Euskadi (21%). Las siguientes comunidades con mayor representatividad son Cataluña y Andalucía, ambas con un 12% de las empresas.

La comunidad de Galicia presenta un 9% de empresas que utiliza Software de Fuentes Abiertas, y la Comunidad Valenciana un 5%.

El resto de comunidades que aparecen en esta distribución son Murcia y Asturias, ambas con un 2%.

Es comprensible, que dado el alto porcentaje de empresas que han declarado que utilizan Software de Fuentes Abiertas en alguna etapa del desarrollo, la distribución geográfica de estas empresas sea muy similar a la de todas las empresas participantes del estudio.

Como se menciona en el apartado 3.3.1 Distribución geográfica, Madrid es la comunidad con mayor representatividad. La definición de las oficinas centrales en la capital del Estado es una práctica común en muchas empresas, lo que le da a dicha comunidad una representatividad en el tejido empresarial muy alta. Su localización geográfica, en el centro de la península favorece en muchos casos el traslado de las oficinas centrales a dicha comunidad.

La comunidad de Euskadi cuenta desde hace tiempo con un fuerte tejido empresarial ya consolidado con una gran representación de medianas y pequeñas empresas, muchas de ellas en un sector de marcado ámbito industrial que trabaja con sectores empotrados. Cataluña es un caso similar, ya que posee desde hace tiempo un fuerte tejido empresarial que hace posible que la utilización de Software de Fuentes Abiertas haya tenido más tiempo para su implantación, puesto que es un tipo de software y modelo de negocio relativamente nuevo en los sistemas empotrados.

La representatividad de Andalucía viene dada por su emergente fuerza y potencial en el sector de los sistemas empotrados.

4.1.3

>>Distribución de las empresas por tamaño

Las empresas sobre las que se basan los datos de este capítulo se han definido como micro-empresas, pymes-pequeñas empresas, pymes-medianas empresas o grandes empresa, según el número de empleados y según la facturación, tal y como se ha definido en el apartado 3.3.4 Distribución por tamaño de empresa.

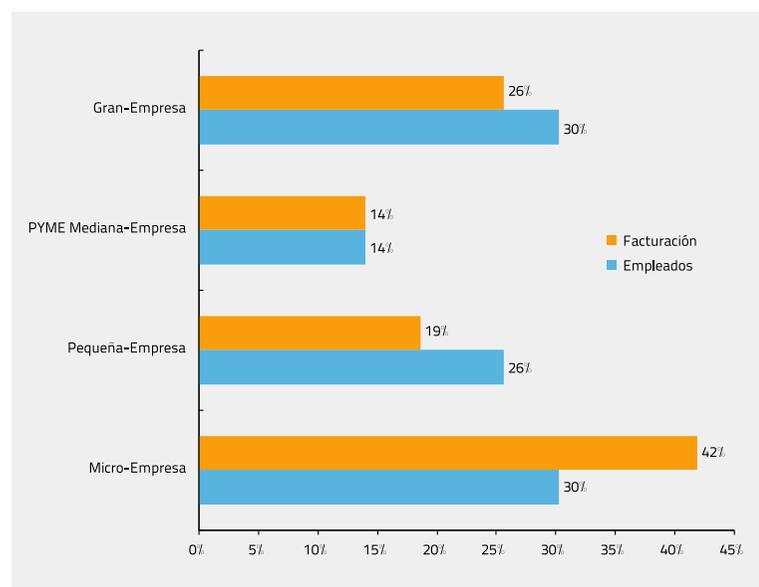


Figura 4.4: Distribución de empresas por tamaño

Con una distribución muy similar a la que se ha dado en las empresas participantes en el estudio, la Figura 4.4 indica que un 30% de las empresas que utilizan Software de Fuentes Abiertas son micro-empresas si se considera el número de empleados. Este porcentaje sube hasta un 42% si se analiza por facturación. Un 26% de las empresas son pequeñas empresas según el número de empleados y un 19% según la

facturación.

El 14% de las empresas que utilizan Software de Fuentes Abiertas son medianas, ya se considere el factor de número de empleados como facturación. Finalmente, un 30% de las empresas son grandes empresas por número de empleado, mientras que este porcentaje baja a un 26% según la facturación.

La micro-empresa se posiciona como la categoría con más representatividad entre las que utilizan Software de Fuentes Abiertas si se realiza la distribución según la facturación, mientras que si se considera el número de empleados las categorías con mayor representatividad son la micro empresa y la gran empresa, ambas con un 30%.

El alto porcentaje indicado en el apartado 4.1.1 sobre la utilización de Software de Fuentes Abiertas (90%) puede disfrazar esta distribución por tamaño de empresa, ya que es muy similar a la obtenida en el apartado 3.3.4. Sin embargo, esta variable asociada con otras como la formación de recursos humanos, colaboración con empresas, etc., puede ilustrar la importancia del tamaño de la empresa en la implantación del Software de Fuentes Abiertas.

4.1.4

>>Distribución de las empresas por sectores de actividad

Esta distribución de las empresas refleja los diferentes sectores de actividad en los que se han enmarcado las empresas participantes que utilizan Software de Fuentes Abiertas. Los sectores de actividad que las empresas han podido elegir son los definidos en el capítulo 3.2

Diseño de la muestra y trabajo de campo. Las empresas han podido seleccionar tantos sectores de actividad como han creído conveniente.

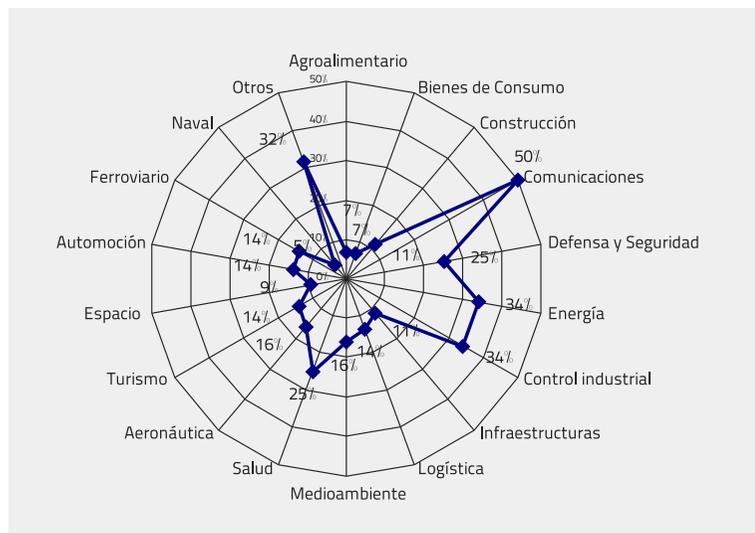


Figura 4.5: Distribución de las empresas por sectores de actividad

Los porcentajes indicados en la Figura 4.5 indican el porcentaje de empresas que utilizan Software de Fuentes Abiertas según sector de actividad principal. De esta forma, el sector de actividad que más empresas han seleccionado como parte de su actividad es el sector de Comunicaciones, con un porcentaje del 50%. Los siguientes sectores con mayor representatividad son los sectores de Energía (34%) y Control Industrial (34%).

La categoría Otros, categoría definida para aquellas empresas que por su variedad o por cualquier otra circunstancia no han podido definirse en el resto de categorías, ha sido elegida por un 25% de las empresas

en cuestión.

La siguiente serie de sectores de actividad han sido escogidos por un porcentaje de empresas similar: Medioambiente (16%), Aeronáutica (16%), Logística (14%), Turismo (14%), Automoción (14%), Ferroviario (14%), Construcción (11%) e Infraestructuras (11%).

Finalmente, los sectores de actividad que han sido escogidos por menos del 10% de las empresas contabilizadas en este capítulo son Espacio (9%), Agroalimentario (7%), Bienes de Consumo (7%) y Naval (5%).

El sector de las Comunicaciones es un sector de actividad presente en un gran porcentaje de sistemas empotrados como se ha mencionado anteriormente en este informe, esto implica que el Software de Fuentes Abiertas esté altamente implantado en este sector. Un conjunto grande y fuerte de empresas involucradas en un sector proporcionan una gran base que permite la consolidación de iniciativas como las de Software de Fuentes Abiertas. Este sector además se apoya fuertemente en estándares, bien sean éstos instaurados por la industria, o por comunidades como las de Software de Fuentes Abiertas, que proporcionan una universalización a través de un uso continuado y promocionado.

Avances tecnológicos en sensores, su miniaturización, la capacidad de comunicación de los mismos, indican un futuro en expansión dentro del sector de las Comunicaciones, que afectará también directamente a otros como la Logística, Medioambiente e Infraestructuras. El diseño de plataformas hardware de fuentes abiertas puede facilitar la expansión y difusión de redes de sensores en las que estos últimos sectores han encontrado una opción de futuro.

El sector de la Energía y el Control Industrial, los siguientes más

seleccionados, son sectores fuertemente implantados en el tejido empresarial de las Comunidades Autónomas con mayor representatividad. Estos sectores, junto con los de Aeronáutica, Automoción, Ferroviario y Espacio son sectores en los que los sistemas empotrados suelen ser sistemas críticos y en tiempo real. El Software de Fuentes Abiertas comienza ahora a ofrecer soluciones con estas dos características (sistemas críticos y en tiempo real), por lo que su inclusión en estos sistemas se asienta en un alto grado de confianza con respecto a estas soluciones.

4.1.5 >>Distribución de las empresas por tecnología

A continuación se muestran las diferentes tecnologías que las empresas usuarias de Software de Fuentes Abiertas utilizan dentro de las diferentes etapas de desarrollo de sus productos. Las empresas han tenido la opción de seleccionar todas las tecnologías que han considerado necesario.

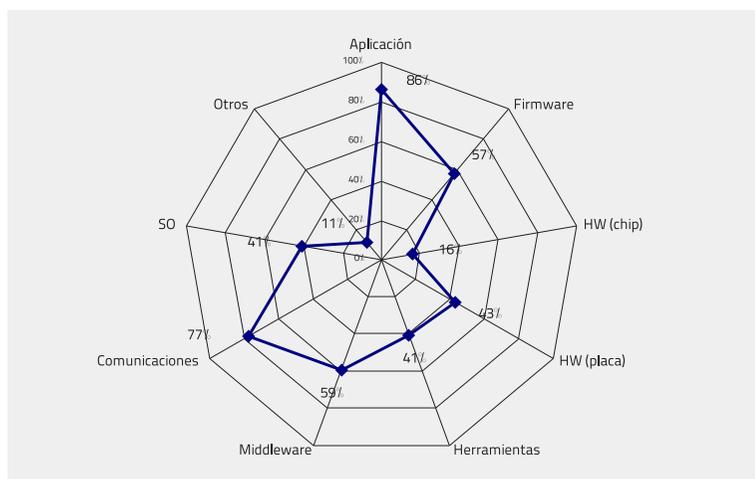


Figura 4.6: Distribución de las empresas por tecnología

La Figura 4.6 muestra como las tecnologías que han sido seleccionadas en un porcentaje significativamente mayor son Aplicación (86% de empresas la han seleccionado) y Comunicaciones (77%).

Las siguientes tecnologías más seleccionadas han sido Middleware y Firmware con unos porcentajes del 59% y 57% respectivamente.

Las tecnologías Hardware (placa), Sistemas Operativos y Herramientas han sido seleccionadas por un 43%, 41% y un 41% de las empresas respectivamente. Finalmente, la tecnología Hardware (chip) ha sido seleccionada únicamente por un 16% de las empresas. Un 11% de las empresas han definido como "Otros" las tecnologías que emplean.

Estas tecnologías son indicativas de las actividades que realizan las empresas objeto del estudio, sin embargo a través de esta pregunta no queda reflejado el uso de Software de Fuentes Abiertas en dichas tecnologías, cuestión que analizaremos más adelante.

De todas formas, una fuerte implantación de una tecnología determinada, en un tejido empresarial concreto, posibilita la aparición de soluciones de fuentes abiertas ya que genera un ecosistema que facilita su aplicación y despliegue.

Por otro lado, las funcionalidades específicas de cada sector de actividad favorece que las soluciones de fuentes abiertas aparezcan más profusamente en los componentes y aplicaciones más genéricas. En sectores más específicos, esto resulta más difícil, a menos que haya un gran respaldo por parte del sector o por una gran compañía líder de sector. En este sentido, al ser las Comunicaciones una tecnología empleada en la práctica totalidad de los sistemas empotrados se intuye una amplia definición de soluciones de fuentes abiertas en este sector.

Dentro del sector Aplicación, gracias a la creciente complejidad de los sistemas empotrados, así como a los grandes avances en capacidad de procesamiento, memoria y demás recursos, se está posibilitando el acercamiento de soluciones en aplicaciones de fuentes abiertas de escritorio y esas mismas aplicaciones, para sistemas empotrados.

Esta misma característica es aplicable al sector de Sistemas Operativos. En este sector se aprecia claramente cómo sistemas operativos diseñados en un principio para sistemas de escritorio, se han modificado acotando los recursos de los que pueden disponer para su adaptación a sistemas empotrados. Los últimos años se ha podido comprobar la continua aparición de sistemas operativos dentro del mercado de los dispositivos móviles (LiMo, Android, Maemo, etc.), en una gran mayoría de fuentes abiertas, con comunidades fuertes tras ellos. Estas comunidades suelen estar respaldadas por grandes compañías del mercado. Es tal la profusión de sistemas operativos en este mercado, que sistemas operativos originalmente cerrados se han abierto para poder competir con el resto, como ha ocurrido por ejemplo con Symbian, que se ha abierto mediante la Fundación Symbian bajo licencia EPL.

Otra tecnología en expansión es la de Herramientas para el desarrollo de sistemas empotrados. Al tener una gran conexión con el software de escritorio se puede beneficiar enormemente de desarrollos existentes y de comunidades ampliamente consolidadas. El entorno de programación Eclipse es una buena prueba de ello.

El bajo porcentaje de empresas que han señalado trabajar la tecnología Hardware a nivel de chip indica que es una tecnología incipiente en el tejido empresarial español, con lo que las fuentes abiertas probablemente no estarán muy difundidas.

Esta tecnología viene dada por los últimos avances tecnológicos que permiten el acceso a tecnologías que antes eran desarrolladas únicamente por grandes compañías, fabricantes de chips para empresas de menor capacidad y recursos. Tradicionalmente el diseño Hardware a nivel de chip ha estado en manos de unas pocas compañías que ejercían su monopolio a nivel global, en cambio en la actualidad el desarrollo de chips con una gran capacidad de diseño interno (FPGAs, SoC-System on Chip, ASIC-Application Specific Integrated Circuit, etc.) facilitan su manejo por parte de empresas con menores recursos. La inclusión de soluciones de fuentes abiertas en el diseño de estos chips facilitan su accesibilidad a empresas que anteriormente lo hubieran encontrado imposible.



Una fuerte implantación de una tecnología determinada, en un tejido empresarial concreto, posibilita la aparición de soluciones de fuentes abiertas ya que genera un ecosistema que facilita su aplicación y despliegue.

4.1.6

>>Visión de la cadena de valor

Los sistemas empotrados encuentran aplicación en multitud de sectores tan diversos como la automoción, las telecomunicaciones, el aeroespacial, etc. No es por lo tanto posible presentar una red de valor que represente en detalle todas estas industrias, los actores que en ellas existen y cómo interactúan entre ellos. Éste tampoco es el objeto de esta sección.

No obstante, resultará útil para subsiguientes secciones disponer de un dibujo genérico que incluya la lógica básica del negocio de los sistemas empotrados, los dispositivos y componentes electrónicos tal y como se presenta hoy. Como veremos, la dinámica de esta industria está en continua evolución.

Los principales actores identificados en el ámbito de los sistemas empotrados, los dispositivos y componentes electrónicos son los siguientes:

- Proveedores de infraestructura y herramientas software base: sistemas operativos, compiladores, herramientas de desarrollo, etc.
- Proveedores de infraestructura y componentes hardware base.
- Proveedores de plataformas y diseño hardware.
- Proveedores de software específico del dominio: software para aplicaciones médicas, cálculo, aeronáutica, etc.
- Otros proveedores de servicios.
- Integradores de sistemas.

- Entidades comercializadoras.

Dependiendo del sector de actividad, alguno de estos actores puede no existir o la dinámica del negocio puede resultar ligeramente diferente. Por ejemplo, la comercialización de los dispositivos móviles dista enormemente de la comercialización de vehículos aeroespaciales.

En la actualidad, la lógica de negocio de los sistemas empotrados presenta un alto índice de complejidad, aproximándose al modelo denominado *value network*, que se muestra en la Figura 4.7, y en el cual los proveedores colaboran también entre sí y con el integrador de sistemas para lograr el producto final. Sin embargo, en las etapas *downstream* podríamos aún encontrar un modelo más parecido a la cadena de valor jerárquica, en el sentido que el integrador de sistemas (OEM) es el que provee al comerciante, siendo este último el que finalmente llega al cliente final.

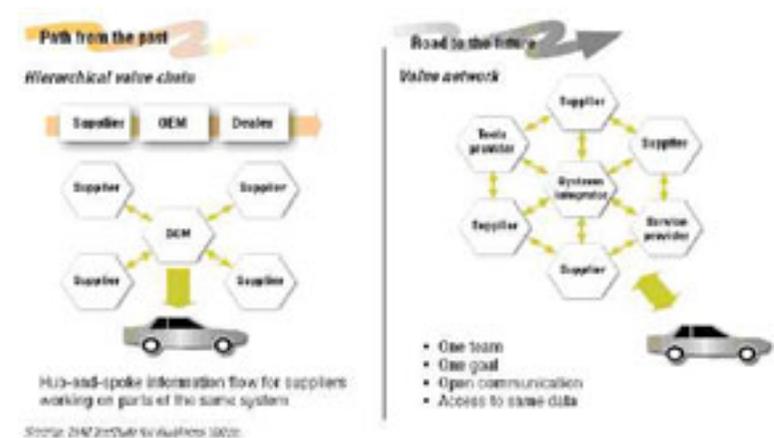


Figura 4.7: Evolución de la lógica de negocio para los sistemas empotrados (Fuente: IBM Institute for Business Value[1])

El presente estudio centra su análisis en aquellos agentes proveedores de la tecnología; para facilitar una visión global sobre el Software de Fuentes Abiertas se plantea una arquitectura de sistema esquemática que sirve como referencia y dónde se proyecten las capacidades y posibilidades existentes.

Un sistema se puede contemplar desde dos perspectivas ortogonales:

- **Visión física del dispositivo o equipo**, que describe los diferentes elementos que constituyen los recursos físicos que soportan las funcionalidades implementadas. Básicamente, una o varias placas conectadas constituidas por una tarjetas de circuito impreso, donde se integran una serie de componentes, entre los que se encuentran:

- Chipset, que sirve como centro de conexión entre el procesador, la memoria RAM, los buses de expansión y otros dispositivos.

- Micros. El microprocesador es un circuito integrado que contiene todos los elementos que conforman la CPU (Central Process Unit).

- DSPs. Un DSP (Digital Signal Processor) es un sistema basado en un procesador o microprocesador que posee un juego de instrucciones, un hardware y un software optimizados para aplicaciones que requieran operaciones numéricas a muy alta velocidad.

- ASICs. Un ASIC, Circuito Integrado para Aplicaciones Específicas, es un circuito integrado hecho a la medida para un uso en particular, en vez de ser concebido para propósitos de uso general.

- FPGA. Una FPGA (del inglés Field Programmable Gate Array) es un dispositivo semiconductor que contiene bloques de lógica cuya interconexión y funcionalidad se puede programar.

- SoC (System-on-chip). Integración de los componentes de un computador o en otros sistemas electrónicos en un circuito integrado (chip). Este dispositivo, contienen las funciones de procesamiento digital, analógica, señal mixta, y a menudo radio-frecuencia.

- Sensores. Un sensor es un dispositivo capaz de transformar magnitudes físicas o químicas, llamadas variables de instrumentación, en magnitudes eléctricas.

- Actuadores. Se denominan actuadores a aquellos elementos que pueden provocar un efecto sobre un proceso automatizado.

- **Visión lógica**, en la que se presentan las diferentes capas de abstracción SW.

- Firmware. Bloque de instrucciones de programa para propósitos específicos, grabado en una memoria de tipo no volátil (ROM, EEPROM, flash,...), que establece la lógica de más bajo nivel para el control de la electrónica de un dispositivo. Funcionalmente, el firmware es el intermediario (interfaz) entre las órdenes externas que recibe el dispositivo y su electrónica, ya que es el encargado de controlar a esta última para ejecutar correctamente dichas órdenes externas.

- Sistema Operativo. Los sistemas operativos son capas software que abstraen el hardware, gestionando la relación con el hardware de forma transparente y posibilita que un mismo programa pueda ser reutilizado en diferentes plataformas hardware. Un sistema operativo de tiempo real (RTOS), es un sistema operativo que ha sido desarrollado para aplicaciones de tiempo real donde la corrección de su comportamiento debe ser tanto funcional como sujeta a ciertas restricciones de tiempo.

- Comunicaciones (pilas/protocolos de comunicaciones).
- Middleware. El middleware es un software de conectividad que ofrece un conjunto de servicios que hacen posible el funcionamiento de aplicaciones distribuidas sobre plataformas heterogéneas.
- Aplicación. Funcionalidad a usuario final o integrador de sistemas.

La Figura 4.8 muestra esta arquitectura de referencia en detalle, así como sus componentes e interfaces habituales entre capas. Cada uno de estos niveles constituye un nicho tecnológico de mercado, el cual provee servicios al nivel superior.

[1] IBM Consulting Services. "Embedded Systems Overhaul". 2004

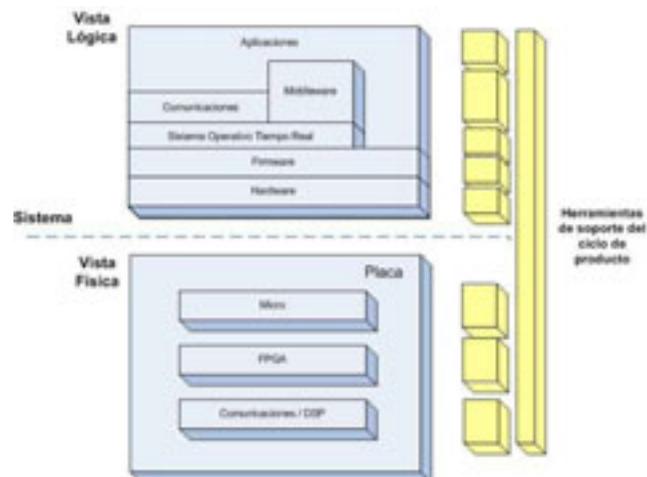


Figura 4.8: Arquitectura de referencia

4.1.7 >> Mapa de capacidades de las empresas

Los sectores de actividad seleccionados, debido a las necesidades inherentes al propio sector, tendrán una especial influencia o preferencia en el desarrollo hacia una determinada tecnología. En la Figura 4.9 se muestra la correspondencia existente entre los sectores de actividad y las tecnologías. El objetivo de la presente gráfica es mostrar no sólo los sectores más destacados, en lo que a número de empresas involucradas en el ámbito de los sistemas empuotrados se refiere, sino también consiste en observar la importancia de las diferentes tecnologías acorde con el sector. A continuación se va a realizar un análisis buscando los elementos que interrelacionan los dos conceptos.

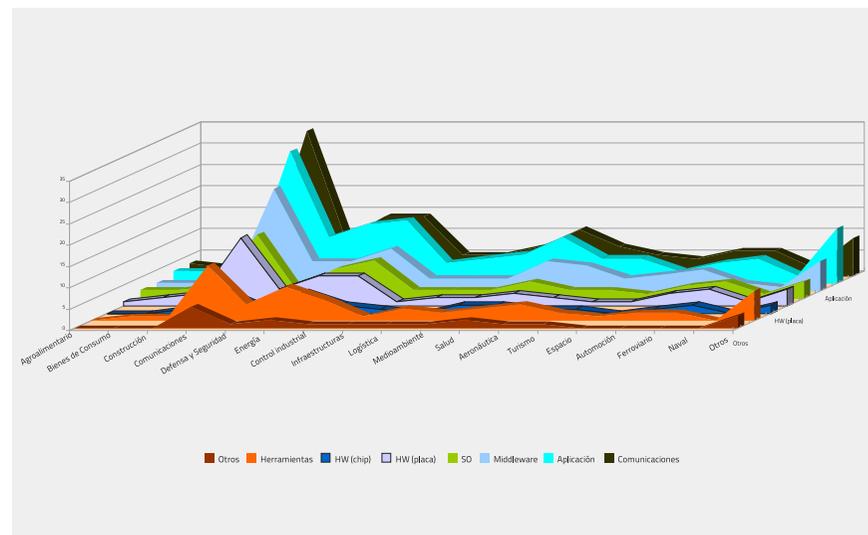


Figura 4.9: Mapa de capacidades

El sector de las Comunicaciones es uno de los más fuertes dentro del ámbito de los sistemas empotrados. Es por ello que la representación de empresas que realizan desarrollos para este sector dentro de la muestra es también importante, tal y como se puede ver en los picos de la gráfica. Otros sectores de gran relevancia representativa dentro de la muestra son Energía, Control industrial, Salud y Defensa y seguridad. Todos estos sectores, debido a su naturaleza y necesidades de negocio particulares, tienen perfiles muy específicos en los que los sistemas empotrados son uno de los puntos claves de su producción.

En otros tipos de sectores, aunque los sistemas empotrados pueden encontrar su nicho de mercado para cubrir determinadas necesidades, estos no son tan críticos como en el caso de los sectores mencionados anteriormente.

Desde el punto de vista tecnológico, se puede observar que el desarrollo de aplicaciones y herramientas se produce en empresas de absolutamente todos los sectores. Particularmente, se puede decir que existe un bloque de tecnologías que está siempre presente en todas las empresas y en todos los sectores: aplicaciones, comunicaciones y middleware.

Otra tecnología importante, aunque no la más destacada, son los sistemas operativos. Además se puede observar que en proporción es mucho mayor en comparación con otras tecnologías. Le siguen en importancia el bloque Firmware y herramientas. Esta división en bloques puede ser debida al propio desarrollo de sistemas empotrados que conlleva un ciclo completo que engloba a todos estos bloques en conjunto. En ese sentido, determinadas tecnologías pueden ser externalizadas y otras tienden más a ser implantadas y realizadas por la propia

compañía, lo que daría la explicación de agrupar determinadas tecnologías para unificar su proceso dentro del ciclo de desarrollo del sistema empotrado.

Entre los sectores involucrados en hardware ya sea en la placa o chip, se encuentran Comunicaciones, Defensa y seguridad, Energía, Control industrial y el Sector ferroviario. En este sentido, se tratan de sectores en los que las necesidades específicas pueden marcar la necesidad de desarrollar hardware específico, siendo este punto crucial en sus ciclos de desarrollo. Esto explicaría también el hecho de que en estos sectores, la proporción de tecnologías relacionadas con la parte software (Aplicaciones, Comunicaciones, Middleware, SO) sea también importante. En este tipo de empresas, cubrir el ciclo completo con hardware y software implica mejorar los beneficios de una definición global del sistema empotrado.

Otra cuestión a tener en cuenta, es remarcar el hecho de que en aquellos sectores en los que la parte hardware no es de gran importancia o no existe, si que existe, sin embargo, una relación importante con las tecnologías Firmware. Esto puede ser para cubrir y customizar determinadas necesidades específicas.

Aquellos sectores de actividad más enfocados a sistemas empotrados son con diferencia Comunicaciones, Control Industrial y Energía. Como se ha comentado anteriormente, en estos ámbitos la utilización de sistemas empotrados es inherente a la propia naturaleza de los problemas y necesidades del sector. Es por ello que en estos sectores se aprecia una importante presencia de prácticamente todas las tecnologías. En el resto de sectores, sin embargo, las tecnologías no se presentan con la misma uniformidad.

4.2

>>Portafolio tecnológico: Software y Hardware de Fuentes Abiertas en las empresas de sistemas empotrados

4.2.1

>>Visión de las soluciones globales en el ciclo de desarrollo de sistemas empotrados

La Figura 4.10 proporciona una visión general de las soluciones de Software de Fuentes Abiertas existentes, clasificándolas en las distintas fases del ciclo de desarrollo de un sistema empotrado genérico. Algunas de estas soluciones han sido mencionadas por las empresas participantes que han indicado que utilizan Software de Fuentes Abiertas en alguna etapa de su desarrollo. La Figura 4.10 se ha completado con un análisis de las soluciones de fuentes abiertas existentes actualmente en el mercado.

Las etapas que se definen en el ciclo de desarrollo según el modelo en V son las siguientes:

- Definición de conceptos
- Requisitos y arquitectura
- Diseño detallado
- Integración, Test y Verificación
- Validación y Verificación del Sistema
- Despliegue y mantenimiento

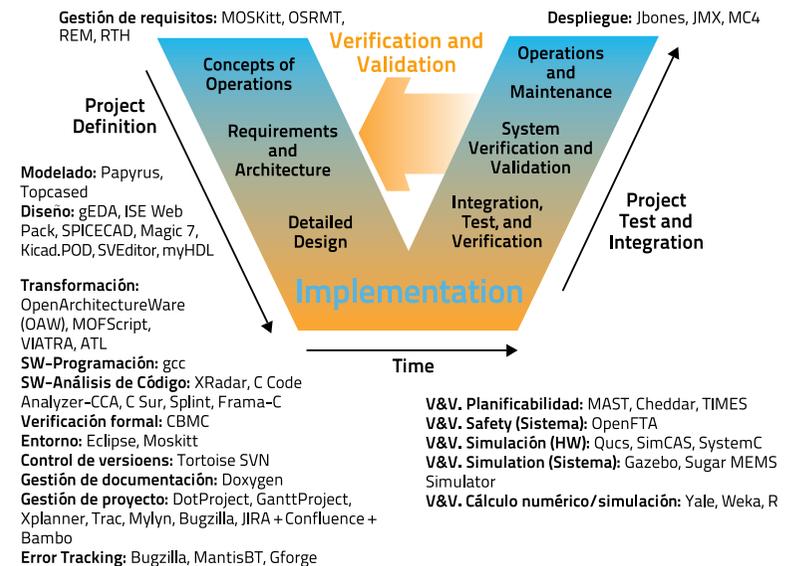


Figura 4.10: Mapeo de herramientas de fuentes abiertas según el ciclo en V

Como se observa en la Figura 4.10, es posible encontrar herramientas de fuentes abiertas a lo largo de casi todo el ciclo evolutivo de desarrollo de sistemas empotrados. La parte inicial del ciclo de desarrollo, definición de conceptos, es similar a un desarrollo de un sistema de computación generalista. Salvo en algún caso particular, la gestión de conceptos de alto nivel como lo que el sistema debe hacer, consideraciones del entorno, interrelación con otros productos, presupuestos y demás conceptos son extrapolables a cualquier tipo de sistema, con lo que las herramientas existentes son perfectamente válidas para los sistemas empotrados.

En la fase de requisitos y arquitectura, los sistemas empotrados empiezan a tener consideraciones específicas que tienen que ser tenidas en cuenta en las herramientas, si se quiere que éstas sean plenamente competitivas. Los requisitos de un sistema empotrado pueden ser

estudiados de una forma similar a cualquier tipo de desarrollo. Será en el contenido de los mismos donde se especifiquen las particularidades de los sistemas empotrados, pero las herramientas de gestión de requisitos ya soportan prácticamente todas las singularidades que pueden presentar los sistemas empotrados.

Al contrario que en otro tipo de desarrollo, la fase de definición de la arquitectura sí que presenta posibles diferencias a otro tipo de desarrollo software. La influencia del hardware en la arquitectura, posibilidades técnicas, recursos de procesamiento, implementación software, etc. obliga a su consideración en la definición de la arquitectura completa del sistema. Las herramientas utilizadas para la definición de la arquitectura y el diseño de la misma tienen que tener en consideración las limitaciones que el hardware impone a los sistemas empotrados. Los recursos que una aplicación tiene en un sistema empotrado no son los mismos que en un sistema de computación genérica, por lo que es necesario un soporte por parte de la herramienta.

Herramientas de modelado, como Papyrus basada en UML 2, son lo suficientemente abiertas como para integrar perfiles específicos para sistemas empotrados como MARTE, (Modeling and Analysis of Real Time and Embedded systems).

Dentro de lo que es el diseño e implementación del sistema empotrado, los analizadores de código son unas herramientas muy solicitadas. Existen diferentes aplicaciones que realizan este trabajo, aunque las dificultades que se dan en este campo son bastante grandes, la extensa gama de arquitecturas hardware implica en muchos casos la utilización de lenguajes C no totalmente compatibles, suponiendo un esfuerzo por parte de los desarrolladores la adaptación del código a C estándar o la particularización de las herramientas de análisis a lenguajes adaptados a diversas arquitecturas hardware.

Donde se aprecia una mayor carencia de herramientas es en la fase de validación, verificación y test. El espectro de sectores de sistemas empotrados es tan amplio, que la definición de herramientas genéricas de validación y verificación se convierte en una ardua tarea. Existen herramientas específicas de un sector o tipo de sistema empotrado, como por ejemplo OpenFTA, herramienta de análisis FTA (Fault Tree Analysis) enfocada a la fiabilidad de sistemas en ámbitos médicos, aeroespaciales, nucleares y de defensa.

Un reto genérico, no solo en los sistemas empotrados sino ampliable a otro tipo de sistemas es la definición de un marco global de herramientas interconectadas que permitan el flujo de información entre ellas y que abarquen la totalidad del ciclo evolutivo de desarrollo de sistemas. Existe una gran demanda de integración de herramientas y sus flujos de datos e información que permitan englobar todas las fases, conceptos y actores del ciclo de desarrollo.

La Figura 4.11 muestra las Soluciones de Fuentes Abiertas desde un punto de vista de la cadena de valor. En esta figura se aprecia claramente en qué etapas de la cadena de valor existen unas grandes carencias (zonas sin color).

Dentro del diseño hardware, nuevos campos en continua expansión como sensores y actuadores que supondrán una nueva revolución en la definición de sistemas integrados, carecen de herramientas de diseño a un nivel hardware, herramientas de modelado. Nuevas tecnologías en este campo, que proporcionan unas características de gran eficiencia energética a estos dispositivos, están permitiendo su implantación en sectores en los que era impensable. Sectores como medioambiente o salud se están beneficiando del bajo consumo logrado por este tipo de dispositivos para definir nuevos productos y servicios que proporcionan un valor añadido en cuestiones como la monitorización y control.

4.2.2 >>Mapa de soluciones globales de fuentes abiertas por sectores de actividad

Durante los últimos años la oferta y aplicación de soluciones abiertas ha mantenido un ritmo continuo de crecimiento. Los proveedores se han servido de este crecimiento en la adopción e integración de soluciones genéricas y orientan su valor añadido hacia su conocimiento o know-how específico del sector

Con esta idea en mente, no debe sorprendernos que el mayor número de iniciativas, se centren precisamente en los niveles más genéricos y alejados de esta diferenciación, facilidades de integración y por último la definición de herramientas de soporte al desarrollo colaborativo de sistemas.

Los sistemas operativos hace décadas que fue identificado como uno de los ámbitos clave a la hora de trabajar de forma colaborativa entre empresas. La primera referencia en este tipo de iniciativas fue TRON, originado en Japón en los 80 y soportado por las principales empresas del país, surgió como alternativa a Microsoft en un sector emergente: electrodomésticos, teléfonos, faxes, etc. Realmente, corresponde a una definición de interfaces y guías para la creación de un kernel, que habilita la implementación interoperable por parte de diferentes industrias y sectores.

De esta forma, es posible encontrar implantaciones particulares para dominios concretos como: negocio, industria, telecom o evoluciones como Toppers, compatible con la especificación OSEK en el sector de automoción. Pese a ver frenada su expansión mundial mediante un bloqueo comercial por parte de Estados Unidos, es considerado como el sistema operativo con mayor número de implantaciones: 2.700 millones de nuevos aparatos salen al mercado cada año, toda la electrónica asiática, una muestra clara del acierto de la estrategia nipona.

En paralelo, iniciativas europeas y norteamericanas, soportadas por la industria aeroespacial (ESA, NASA) y departamentos de Defensa, han propiciado la aparición de sistemas operativos de tiempo real, de nuevo casos éxito en el ámbito de Software de Fuentes Abiertas en sistemas empotrados.

Desde la perspectiva física, con la aparición de los nuevos dispositivos de lógica programable reconfigurables, el concepto de fuentes abiertas ha logrado superar los límites del software y empieza a dar sus primeros pasos también en el ámbito del hardware. Los métodos y lenguajes asociados a estas tecnologías, permiten la compartición del código HDL (Hardware Description Language) de forma directa, evitando el intercambio de esquemáticos, más complejo y sensible. Estas descripciones HDL son usadas comúnmente para sintetizar sistemas SoC en FPGA o directamente en fabricación de diseños ASIC. Los módulos HDL, así distribuidos, se denominan semiconductor intellectual property cores (IP-Cores), o núcleos IP.

A día de hoy existen diferentes comunidades que trabajan en esta línea: en el diseño y desarrollo, asumiendo también la necesidad de la calidad de los productos y realizando tareas de verificación, testeo, formación y soporte. Las más destacables son Open Collector, OpenCores u OpenHardware.

- Open Hardware (OH) es un proyecto en el que los diseñadores de hardware comparten su trabajo revelando los diagramas esquemáticos y el software (drivers) usados en sus diseños. Open hardware constituye un foro de discusión en torno a los aspectos específicos del diseño electrónico y supone una ayuda especializada al diseñador, asimismo constituye una oportunidad para la exhibición de diseños y la capacitación de nuevos ingenieros.

- OpenCores es una fundación que procura formar una comunidad de diseñadores para apoyar los núcleos libres (diseños lógicos) para CPU, periféricos y otros dispositivos, la interoperabilidad necesaria para la integración de los mismos, esta soportada mediante un bus de interconexión en-chip libre (Wishbone), que es utilizado como referencia para la interconexión de los diferentes diseños.
- OpenColector centrado principalmente en la provisión de herramientas de soporte abiertas para el proceso de ingeniería electrónica, alrededor de estas herramientas y lenguajes, proveen un repositorio de diseños.

Estas comunidades, particularmente activas, hacen accesibles a día de hoy miles de diseños abiertos, abarcando todos y cada uno de los diferentes elementos que constituyen el soporte físico de los dispositivos: placas, micros, DSPs, etc.

En España existen diferentes iniciativas alineadas con el hardware abierto: Microbótica [MICROBOTICA] posee un proyecto de hardware libre, donde ofrece esquemas de robots para su construcción, además de otros diseños. Algunas universidades por todo el país están fijando sus ojos en este movimiento y tienen líneas de investigación abiertas sobre hardware de código abierto, siendo el Grupo de Diseño HwSw de la Universidad Rey Juan Carlos [ESCET] uno de los pioneros en este campo en España. Sus componentes colaboran habitualmente en el proyecto OpenCores.

Mención aparte merece OpenSoCDesign [OpenSoC], empresa española de reciente creación, que ofrece servicios de integración de diseños de hardware abierto, procesos de verificación externa y consultoría tecnológica para empresas del sector.

De forma específica, destacan algunos diseños concretos que han supuesto la convergencia de diferentes actores en contextos muy concretos alineados con estrategias europeas: Leon3 como micro ampliamente utilizado por la Agencia Espacial Europea (ESA) y que tracciona diferentes iniciativas académicas y empresariales en toda Europa, o los desarrollos de sensores en sectores de alto crecimiento económico como son la salud, motivado por la concienciación del envejecimiento de la población o la monitorización del entorno: ambiental o doméstica.

Según elevamos el nivel de abstracción, en la capa firmware, encontramos escasos resultados. El firmware como se mencionó anteriormente, se encuentra absolutamente integrado con los elementos físicos, lo cual condiciona que sean los fabricantes, conocedores de los detalle de diseño, quienes realicen la programación a bajo nivel. De hecho, las iniciativas de apertura surgen en torno al concepto de Open Firmware u OpenBoot, estándar que define las interfaces de un sistema de firmware de computadora. Impulsado por Sun y utilizado por los principales diseñadores de chipsets, permite al sistema cargar drivers de plataforma independiente directamente desde la tarjeta PCI, mejorando la compatibilidad. Recientemente, han sido otorgadas a la comunidad del Open Source varias implementaciones comerciales, gestionándose el código fuente desde el proyecto OpenBIOS [OpenBios].

Por último, destacar una serie de sistemas cuyo diseño completo (HW/SW/FW) es abierto; en dichos casos la estrategia reside bien en la provisión de servicios de valor añadido y donde el sistema constituye una herramienta para su implementación o bien, con una fabricación/integración de bajo coste en nichos de mercado de demanda masiva.

Las siguientes figuras muestran, en las zonas coloreadas, los sectores de actividad donde se han encontrado un mayor número de soluciones globales de fuentes abiertas.

			Agroalimentario	Bienes de Consumo	Hogar	Construcción	Comunicaciones	Defensa y Seguridad	Energía	Control industrial	Infraestructuras	Logística	Medioambiente	Salud	Aeronáutica	Turismo	Espacio	Automoción	Ferrovionario	Naval	Otros	
HW (chip)		Asociaciones (OpenCores, OpenHardware - OpenCollector) Cores (LEON3-NIOS- PicoBlaze- MicroBlaze- OpenSPARC- OpenRISC- Mico32)																				
HW (placa)		Asociaciones (OpenCores, OpenHardware - OpenCollector)		Arduino/ Freeduino									CodeBlue, Arduino/ Freeduino	Etag, Etag2, X-10 (DexterNet), Ardin			LEON-3					
Comunicaciones / DSP	Como IP HW	MANTIS- ibelium (HW)- Radioactive Foundations (HW)- RifiDi (HW)																				
Firmware																						
SO	BSD y derivadas	FreeBSD																				
	Distribuciones Linux	Linux-Linux, uSEL, RTLinux, Ubuntu, Fedora, Debian, Red Hat																				
	Tiempo Real	RTLinux- FreeRTOS- eCOS- MicroC OS II- Linux Empotrado- Toppers- eTRON- RTAI- RTMES-		RTLinux, MicroC OS II		Android, Limo, OpenMoko, MicroC OS II	MicroC OS II	RTLinux, TRON	MicroC OS II				MicroC OS II	RTMES		RTMES	Toppers, TRON	eCos				
	.NET	.Net,micro																				
	Microkernels	L4, RadiOS SymbOS				L4, SymbOS																
Middleware-integración-comunicaciones	Middleware/ Integración	SOA /gSOAP)- CORBA (TAO,ACE) -DDS (O-Bus, OpenSplice, openDS) - OSGi (OSCAR, Knopflerfish, Eclipse OSGi runtime, Apache Felix), -Dbus-Nunit-Mercurial, Arctic Core (Autosar)				SOA	CORBA, DDS	CORBA, DDS	SOA									Arctic Core (Autosar), OSGi				
	Comunicaciones	UPnP (Ruddi, Apache jurn), UDDI (linux SQK, Media Tomb, Gmode Server)																				
	Hypervisores	Xtratum, Xen						Xen	Xen								Xtratum					
Librerías	Procesado de imagen	Cimage, Tina,																				
	Interfaces gráficos de usuario (GUI)	X -Xfree86-GNOME- QT																				
	Comunicaciones	Freedesktop (Gstreamer Telepathy)																				
	Interoperabilidad	Maemo- TAO/ACE- Net SNMP- OpenDDS																				
	Aplicaciones Web	Webkit																				
Otros	SICL																					
Nivel de aplicación	Funcionalidades específicas de dominio		MAKE, Roomba, Chumby	TV-B-G, Open IMS		LART, RFID Guardian, Maemo, Molin, WRT54GL				Linux PLC, SADL, Blitech, MORFEO Wasup			SunSPOT									

Figura 4.12: Mapa de soluciones globales de fuentes abiertas por sectores de actividad (I)

			Agroalimentario	Bienes de Consumo	Hogar	Construcción	Comunicaciones	Defensa y Seguridad	Energía	Control industrial	Infraestructuras	Logística	Medioambiente	Salud	Aeronáutica	Turismo	Espacio	Automoción	Ferrovial	Naval	Otros	
Herramientas	Programación	gcc																				
	Modelado	Papyrus, Topcased												Topcased (AADL)								
	Diseño Hardware	gEDA, ISE Web Pack, SPICECAD, Magic 7, KiCAD, POD, SVEditor, myHDL																				
	Eclipse	Eclipse																				
	Análisis de Código	Xradar, C Code Analyzer - CCA, C Sur, Splint, Frama-C																				
	Validación y certificación	CBMC,MAST, Cheddar, TIMES,OpenFTA,Qu cs, SimCAS, SystemC, Gazebo,Sugar MEMS Simulator- FreeFem																				
	Gestión de requisitos	MOSKitt, OSRMT, REM, RTH																				
	Despliegue	JBones, JMX, MC4J																				
Aplicaciones y entornos	Navegadores	FireFox -Chrome																				
	Soluciones Ofimáticas	OpenOffice-MediaWiki-Alfresco-OpenProj																				
	Entornos	PHP- Eclipse -Apache-jboss- CMAKE - MinGW-gcc-kdeveloper																				
	Bases de datos	MySQL																				
	Dibujo	DIA																				
	Error Tracking	Bugzilla- MantisBT - Gforge																				
	Simulación	Gazebo -Sugar MEMS Simulator																				
	Control de versiones	Tortoise SVN																				
	Gestión de documentación	Doxygen																				

Figura 4.13: Mapa de soluciones globales de fuentes abiertas por sectores de actividad (II)

4.2.3

>> Mapa de utilización de Software de Fuentes Abiertas

En esta sección se ha resumido en la Figura 4.14 el mapa de utilización de Software de Fuentes Abiertas de las empresas que han participado en la encuesta y han declarado utilizar Software de Fuentes Abiertas.

			Agroalimentario	Bienes de Consumo	Construcción	Comunicaciones	Defensa y Seguridad	Energía	Control Industrial	Infraestructuras	Logística	Medioambiente	Salud	Aeronáutica	Turismo	Espacio	Automoción	Ferrovuario	Naval	Otros
HW (chip)		ARM con kernel Linux - LEON3- NIOS-																		
HW (placa)																				
Firmware																				
SO	BSD y derivadas	FreeBSD																		
	Distribuciones Linux	Linux-Linux, uSEL, RTLinux, Ubuntu, Fedora, Debian, RedHat																		
		Tiempo Real	RTLinux- FreeRTOS- eCOS- MicroC OS II																	
	.NET	.Net,micro																		
Middleware-integración-comunicaciones	Comunicaciones	SOAP-CORBA- Dbus-Nunit-Mercurial																		
	Integración	Eclipse																		
Librerías	Procesado de imagen	image																		
	Interfases gráficos de usuario (GUI)	X-Xfree86- GNOME-QT																		
	Comunicaciones	Freedesktop (Gstreamer Telepathy)																		
	Interoperabilidad	Maemo- TAO/ACE- Net SNMP- OpenDDS																		
	Aplicaciones Web	Webkit																		
	Otros	SICL																		
Aplicaciones y entornos	Navegadores	FireFox - Chrome																		
	Soluciones Ofimáticas	OpenOffice- MediaWiki- Alfresco- OpenProj																		
	Entornos	PHP- Eclipse - Apache- Jboss- CMAKE - MinGW-gcc- kdevelop																		
	Bases de datos	MySQL																		
	Dibujo	DIA																		
	Error Tracking	Bugzilla- MantisBT - Gforge																		
	Simulación	GACEBO - Sugar MEMS Simulator																		
	Control de versiones	Tortoise SVN																		
	Gestión de documentación	Doxygen																		
	Licencias	GPL y derivadas	GPL-MGPL- LGPL																	
Apache		Apache																		
Creative Commons		CC																		
BSD y derivadas		BSD																		
Herramientas	Programación	gcc																		
	Modelado	TOPCASED																		
	Hardware	GEDA- TOPCASED- gEDA- SystemC - SCOPE- Freemin- xemacs - gcc																		
	Eclipse	Eclipse																		

Figura 4.14: Matriz de capacidades

La Figura 4.14 presenta de forma resumida todo el conjunto de soluciones software que las empresas han mencionado, indicando según el grado de color la influencia o representatividad de cada solución.

En los apartados posteriores se definirá en mayor detalle cada uno de los tipos de soluciones de fuentes abiertas utilizadas por las empresas objeto de estudio.

En el mapa se puede apreciar cómo en el inicio de la cadena de valor, en la fase referente al hardware, no se da un abanico efectivo de soluciones que se estén utilizando de manera amplia y consolidada. Si bien es cierto que en determinados sectores de actividad se aprecia una representación significativa, principalmente es el de Comunicaciones el que acapara la representatividad dentro de la tecnología. El sector de Comunicaciones es un sector que se puede considerar también transversal, lo que permite que su uso sea más generalizado que el resto de sectores.

A nivel Firmware, no se ha indicado por parte de las empresas la utilización de soluciones de fuentes abiertas. La delimitación entre software y firmware no se presenta clara en muchas ocasiones, con lo que es posible que soluciones de Software de Fuentes Abiertas sean en parte firmware, sin estar consideradas en esta categoría. Al ser una tecnología muy dependiente del hardware, en muchas empresas se implementan desarrollos propios.

Las tecnologías donde se aprecia que se utilizan soluciones de Software de Fuentes Abiertas en mayor medida son aquellas tecnologías que se pueden considerar aplicables directamente en muchos sectores de actividad. Los Sistemas operativos presentan soluciones de fuentes abiertas que están muy consolidadas y probadas en el entorno de escritorio, y es una de las tecnologías que ha dado el salto a los sistemas empotrados, aunque todavía con un gran camino por recorrer puesto que aunque se utilizan en varios sectores de actividad, no están generalizados en todos ellos. Las particularidades que cada sector de actividad específica para sus sistemas empotrados abre el abanico de soluciones necesarias. Los Sistemas

operativos en tiempo real, o para sistemas críticos, pueden avanzar bien sea en complejidad para aquellos sistemas empotrados que lo necesiten y posean recursos para soportarlos, o también pueden avanzar en la reducción de recursos necesarios para su implementación, con el objetivo de popularizarlos en sistemas empotrados que actualmente tienen el sistema operativo integrado en la propia aplicación. De momento, se comprueba que en este aspecto es Linux y sus posibles variantes quienes ganan la partida, ya sea en la reducción de complejidad como en su adaptación a sistemas muy complejos

Donde también se ve que hay un uso de soluciones de fuentes abiertas es en las Aplicaciones. En este campo, la heterogeneidad de las funcionalidades de los sistemas empotrados complica la aparición de soluciones globales que puedan ser utilizadas en sistemas empotrados. Aquellas soluciones que tienen su equivalente en los sistemas de escritorio son las que primero se posibilita su adaptación a sistemas empotrados. Esto no es óbice para que surjan también soluciones específicas para problemas concretos que se dan exclusivamente en sistemas empotrados y para un sector de actividad determinado.

Atendiendo a la cadena de valor, no hay una continuidad de soluciones a lo largo de la misma, que permita implementar un desarrollo continuo mediante fuentes abiertas en todas las etapas. Es posible, que para algún sector de actividad, con una arquitectura determinada, haya soluciones para todas las fases de la cadena de valor, pero no es una tónica generalizada para diferentes sectores.

4.2.4 >>Uso del Hardware de fuentes abiertas

A la hora de comenzar con el diseño de un sistema empotrado, el hardware del mismo tiene un papel importante, ya que, entre otras cosas, nos delimitará las capacidades de las que luego el software se podrá aprovechar.

El núcleo del sistema (microprocesador, memorias y similares) es importante

en el diseño, aunque no es la característica que destaca en un sistema empotrado y le da su personalidad. Lo que realmente identifica a un sistema empotrado es su periferia, es decir, las partes adicionales que realizan la tarea específica para la que se diseña.

La forma en que se conectan estos elementos al núcleo para realizar su control son las que nos indicarán como deberemos manejarlos a la hora de desarrollar el software. Y también otras cosas como coste o rendimiento.

Para llevar a cabo este desarrollo, la necesidad de una plataforma comercial y efectiva de diseño de sistemas empotrados es importante. Por ello, la utilización de hardware de fuentes abiertas para llevar a cabo la implementación de un sistema, aporta una gran versatilidad en los desarrollos.

A la hora de analizar el grado de implantación del hardware de fuentes abiertas en las empresas españolas, nos centramos en aquellas cuyas líneas tecnológicas de especificación sean las que se han definido ya en este estudio como Firmware, Hw (chip) y Hw (placa).

4.2.4.1 >> Herramientas de fuentes abiertas de Diseño Hardware

Como se puede observar en la Figura 4.15, el uso de herramientas de fuente abierta entre las empresas encuestadas, a la hora de llevar a cabo el diseño hardware de un sistema, es relativamente bajo (20%).

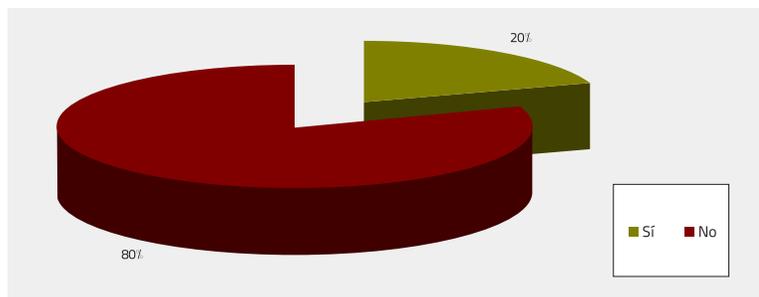


Figura 4.15: Empresas que utilizan herramientas Hardware de fuentes abiertas

En la actualidad existen multitud de iniciativas de desarrollo de herramientas de fuentes abiertas para el diseño hardware. Estas iniciativas parten del entorno de las universidades, y son éstas las que más aprovechan sus ventajas. El problema surge a la hora de potenciar su uso dentro de las empresas. Todavía existe el mito de que las herramientas de fuentes abiertas son poco potentes, lo que hace que la inmensa mayoría de las empresas que necesitan herramientas para el diseño de sus sistemas se decidan por utilizar herramientas cerradas.

Aunque la utilización de este tipo de herramientas es bajo, sorprende el hecho de que aquellas empresas que sí disponen de herramientas de fuentes abiertas para sus desarrollos, utilizan una amplia variedad de programas (compiladores, simuladores, etc.), escogiendo cada vez, aquel que cumpla en mayor medida las necesidades del momento. Es dentro de los simuladores donde los desarrolladores utilizan un conjunto mayor de herramientas que se adaptan positivamente a los objetivos que persiguen.

Como compilador más usado está GCC, el cual posee ciertas ventajas frente a otros compiladores: incluye más lenguajes y los implementa mejor, tiene más extensiones, funciona en más arquitecturas, etc. Otra de las herramientas más utilizadas es gEDA, paquete de herramientas de Automatización de Diseño Electrónico. Como herramienta específica se puede destacar TOPCASED, que es un conjunto de herramientas dedicadas al desarrollo de sistemas críticos y empotrados.

En este estudio, la mayoría de las empresas encuestadas achacan el bajo uso de herramientas abiertas a la poca madurez de éstas. Aunque existen herramientas de fuentes abiertas ampliamente desarrolladas y en constante actualización, las empresas las consideran poco potentes para la implementación del hardware tan específico que desarrollan algunas de ellas. Otro de los motivos que plantean las empresas para la no utilización de herramientas abiertas en el diseño de sistemas empotrados, viene motivado por la falta de experiencia del personal dedicado al diseño de hardware y, sobre todo, por no desaprovechar la formación que ya poseen en las herramientas cerradas.

4.2.4.2 >>Microprocesadores de fuentes abiertas

En cuanto a la utilización de microprocesadores de fuentes abiertas, el porcentaje de penetración en las empresas españolas relacionadas con los sistemas empotrados es del 20%, como se puede observar en la siguiente Figura 4.16.

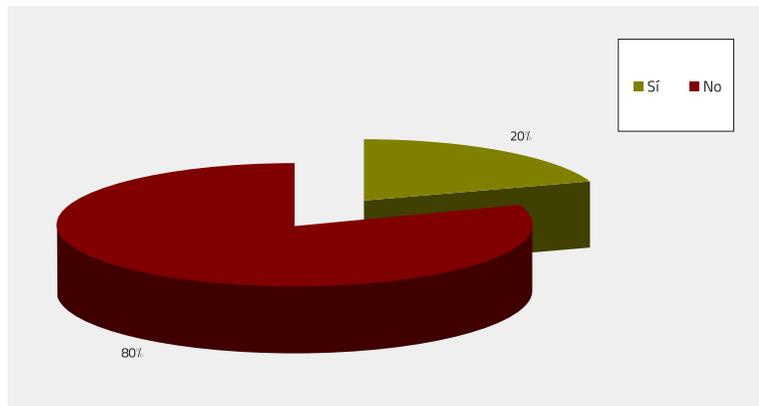


Figura 4.16: Empresas que utilizan Micros de fuentes abiertas

Cada vez más, la mejora en la potencia de los micros de fuentes abiertas, así como el aumento de sus funcionalidades, ha reforzado su utilización en el desarrollo de sistemas empotrados. En particular, la mayoría de las empresas encuestadas utilizan LEONx como un microprocesador clave para sus diseños, ya que su capacidad multiprocesadora aumenta considerablemente su rendimiento. Otro de los micros utilizados es PicoBlaze, que combina una funcionalidad básica fácilmente extensible, con una comunicación con el exterior a través de sus puertos de entrada/salida.

Al igual que ocurriría en el caso anterior, la potencialidad de los microprocesadores de fuentes abiertas y las ventajas que poseen no

están ampliamente extendidos entre los entornos empresariales.

El principal motivo por el cual las empresas no incluyen estos micros en el desarrollo de sus productos es porque no cubren, en una primera aproximación, las necesidades que requieren sus implementaciones. Existen empresas que han tratado de incorporarlos a sus diseños pero han sido desestimados por falta de prestaciones y de soporte.

4.2.4.3 >>Diseño de Hardware de fuentes abiertas

Antes de analizar la situación de la utilización de diseños hardware de fuentes abiertas en las empresas españolas proveedoras y/o integradoras de sistemas empotrados, expondremos las ideas más importantes de este tipo de hardware.

Cada día, el mercado demanda dispositivos electrónicos más complejos. Al igual que en los grandes proyectos del mundo software, el diseño de estos dispositivos sería inabordable sin la reutilización de código. Esto ha hecho aparecer multitud de compañías que ofrecen sus servicios de diseño y venden su código a otras para que éstas las integren en sus productos.

Un diseño hardware se dice que es de código abierto si su código HDL (Hardware Description Language) se distribuye con una licencia tal que:

- Dé permiso para distribuir, gratuitamente, el código fuente y cualquier dispositivo hardware basado en él.
- Dé permiso para modificarlo y distribuir los cambios bajo la misma licencia.

Esta definición se ciñe a hardware diseñado utilizando un HDL como

Verilog o VHDL. Sin embargo, el hardware abarca otras ramificaciones que el software no contempla. Una definición más amplia es la que ofrece OpenCollector (opencollector.org) que lo define como aquel que cumple las siguientes características:

- El interfaz del hardware es público, con lo cual el hardware puede ser usado libremente.
- El diseño del hardware es público, con lo cual otros pueden implementarlo y aprender de él.
- Las herramientas utilizadas para crear el diseño son libres, por lo tanto otros pueden utilizar y mejorar el diseño.

Esta definición se puede aplicar a distintos tipos de hardware, no sólo al descrito en un HDL, por ejemplo a una PCB (Printed Circuit Board) con componentes discretos, siempre y cuando se distribuya bajo esos términos.

No obstante las verdaderas ventajas del hardware de código abierto surgen cuando se utiliza siguiendo un modelo de desarrollo similar al del software del mismo nombre, esto es, cuando hablamos de código fuente que es distribuido bajo una licencia similar (o igual) a la GPL (General Public License).

Existen algunas comunidades que están interesadas en el desarrollo de hardware de fuentes abiertas. Una de ellas es OpenCores (www.opencores.org).

OpenCores nace con el objetivo de diseñar y publicar los desarrollos bajo una licencia para hardware basada en la LGPL (Lesser General Public Licence) para software, y está muy comprometido con el ideal de un hardware de fuentes abierta disponible libremente, utilizable libremente y reutilizable. Para conseguir esto, la comunidad propone:

- Desarrollar estándares para plataformas y cores open source.
- Crear herramientas y métodos para desarrollar plataformas y cores open source.
- Desarrollar plataformas y cores open source.
- Proporcionar documentación para esas plataformas y cores.

El hardware abierto es la solución a la mayoría de los problemas asociados a cores propietarios. Algunos de sus beneficios se listan a continuación:

Cada core contendrá una amplia base de usuario, lo cual asegura un mejor soporte, una mejor documentación y unos mejores ejemplos de implementación desde los que trabajar.

- El código está disponible, por lo que cualquier desarrollador puede averiguar lo que necesita saber sobre el core.
- No hay cargo por utilizar el core.
- Finalmente, cuando los cores y sus estándares se desarrollen, los cores llegarán a ser más compatibles que los cores propietarios.

Después de esta visión general sobre el diseño de hardware de fuentes abiertas, pasamos a analizar el uso que las empresas que trabajan, de una u otra forma, con sistemas empotrados, hacen de este hardware abierto.

En la Figura 4.17 se puede observar que la penetración de desarrollos hardware con diseño de fuentes abiertas en las empresas españolas es muy baja (13%).

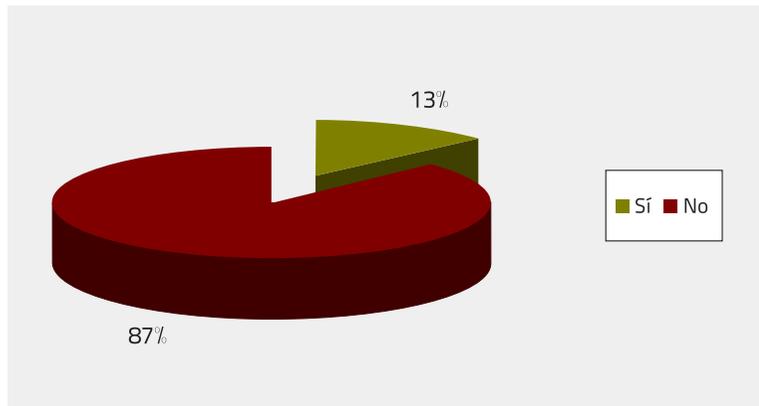


Figura 4.17: Empresas que utilizan diseño Hardware de fuentes abiertas

El principal motivo de esta baja utilización es que estos diseños hardware no encaja con los productos que vienen desarrollando en sus empresas. Además, existe el inconveniente de que son diseños todavía no muy extendidos entre la comunidad de empresas de sistemas empotrados, por lo que el conocimiento que poseen sobre las ventajas competitivas de los mismos es inexistente.

4.2.5

>>Uso del Software de Fuentes Abiertas: aplicaciones, sistemas operativos, librerías y herramientas

En este apartado se analiza el uso que hacen las empresas que han participado en la encuesta, del Software de Fuentes Abiertas según su tipología, es decir, bien sean aplicaciones, sistemas operativos, licencias, librerías o herramientas. El objetivo del presente apartado es mostrar la situación de la utilización de cada uno de los elementos a nivel empresarial.

Para facilitar la comprensión de este epígrafe, éste se ha subdividido en varios apartados con el objetivo de analizar las diversas herramientas,

aplicaciones o técnicas utilizadas. En el primer apartado, se muestra la visión del uso de aplicaciones de Software de Fuentes Abiertas desde una perspectiva general. En el segundo apartado, se ofrece una visión de los sistemas operativos de fuentes abiertas más utilizados. En el tercer apartado, se muestra un análisis sobre las librerías específicas más utilizadas, de acuerdo al área que cubren estas. Posteriormente, se incluye un análisis de los tipos de licencias de Software de Fuentes Abiertas más demandados. Seguidamente, un análisis de los entornos de programación que más se utilizan, entornos que en los últimos tiempos han tenido un auge muy importante. Y por último, se incluye información sobre las principales herramientas de integración que son de interés para las empresas.

Hay que resaltar el hecho, de que debido a lo específico del dominio de sistemas empotrados, no todas las empresas que han participado en la encuesta tienen la necesidad de contestar en todos estos apartados. Es por ello, que al principio de cada apartado se hará un pequeño sumario con los porcentajes de participación y respuesta. Es importante también resaltar que todas las aplicaciones, tecnologías, herramientas, etc están siendo actualmente usadas en el ámbito empresarial.

4.2.5.1 >>Sistemas Operativos

Los sistemas operativos son de gran importancia dentro del ciclo de desarrollo de los sistemas empotrados. En el desarrollo de este tipo de sistemas, en ocasiones, las restricciones hardware, temporales o de configuración, hacen necesario que los sistemas operativos provean de una flexibilidad extra para facilitar la implementación y desarrollo del sistema.

Precisamente, los sistemas operativos de fuentes abiertas ofrecen una gran maniobrabilidad y flexibilidad a la hora de adaptaciones específicas,

consideraciones de restricciones, optimizaciones, etc. Así, se han convertido en una opción y alternativa de gran importancia para las empresas ya que ofrecen una respuesta flexible que se ajusta y adapta a sus necesidades particulares en el ciclo de desarrollo del sistema. Además, en la actualidad, se han desarrollado sistemas operativos eficientes en términos de restricciones temporales, y de tiempo real, que ofrecen ventajas sobre versiones cerradas, mucho más genéricas.

De las empresas participantes en esta encuesta y que utilizan Software de Fuentes Abiertas, un 67% de ellas utilizan sistemas operativos de fuentes abiertas. En la siguiente gráfica (Figura 4.18) se muestra el porcentaje de uso de los diferentes sistemas operativos. Se puede observar que el 47% de las empresas utilizan Linux frente a las demás alternativas.

En los últimos tiempos, alternativas y distribuciones más específicas han hecho su aparición, ganando una importante cuota de mercado, como es el caso de Ubuntu (12%). Así mismo hay que destacar otras distribuciones de gran peso y tradición, como es el caso de Debian (7%), Fedora y RedHat que siguen siendo utilizadas en el 5% de las empresas, respectivamente. Una de las principales ventajas que se le atribuye a Ubuntu, y de ahí quizás el porcentaje de utilización, es la simplificación del interfaz para usuarios expertos y no expertos, facilitando su uso y configuración.

Tras estos grupos, aparecen una gran cantidad de sistemas operativos más específicos que tienen unas características especiales y necesarias dentro del ámbito de los sistemas empujados, como pueden ser las consideraciones de tiempo real, latencias, etc. Los más utilizados en este caso son: FreeRTOS, eCOS y RTLinux con un 5% de las empresas, respectivamente.

Otras opciones más específicas, debido a lo cual tienen una menor demanda, serían FreeBSD (2%), uSEL (2%), MicroC OS II (2%), y .Net microframework (2%). La principal característica que se puede destacar en algunos de ellos son las facilidades y prestaciones disponibles para el manejo de elementos a bajo nivel y su correspondencia con elementos hardware, la posibilidad de modificación del código existente o la posibilidad de añadir elementos específicos.

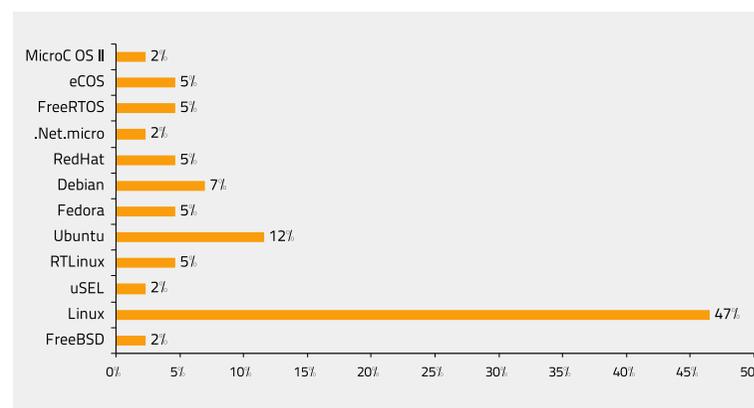


Figura 4.18: Sistemas Operativos

Desde el punto de vista de los sistemas empujados, la utilización de sistemas operativos de fuentes abiertas proporciona una ventaja competitiva clara en la producción y desarrollo en el ciclo de vida de los sistemas empujados. En este sentido, la inclusión de sistemas operativos de fuentes abiertas proporciona a los desarrolladores una gran flexibilidad, libertad y control sobre el desarrollo, características muy apreciadas debido a la gran variedad de elementos hardware que deben ser integrados y restricciones a niveles de tiempos, latencias, etc.

Del mismo modo, en los últimos tiempos ha aumentado la demanda de sistemas operativos de fuentes abiertas, especialmente sistemas de

tiempo real, donde se buscan unas características muy concretas, sobre todo en lo que se refiere a prestaciones para el manejo de elementos a bajo nivel y referente a temas de rendimiento. Este tipo de sistemas suelen proveer de unos rendimientos y escalabilidad mucho más competitivos que otros tipos de sistemas operativos más generales.

La gran ventaja que supone el uso de sistemas operativos de fuentes abiertas frente a las versiones cerradas es la facilidad de acceso al código, el verdadero control sobre las operaciones que se están realizando, y las posibilidades de empotrar el propio sistema operativo en el sistema empotrado, realizándolo de manera eficiente y efectiva. Cabe destacar también el hecho de que en la mayoría de las empresas entrevistadas, se utiliza más de un sistema operativo de fuentes abiertas, dependiendo de las necesidades específicas a cubrir.

4.2.5.2 >> Aplicaciones - herramientas

Las aplicaciones de Software de Fuentes Abiertas han experimentado un gran auge en los últimos años cubriendo un amplio espectro de las necesidades de la empresa a todos los niveles, desde soluciones generales, como pueden ser las soluciones ofimáticas, hasta soluciones más específicas relacionadas con entornos técnicos, como puede ser el caso de entornos integrados de programación, control de errores y versiones. En este sentido, las empresas de sistemas empotrados también se pueden beneficiar no sólo de las soluciones anteriormente comentadas, sino incluso de soluciones totalmente específicas como pueden ser los entornos de simulación, donde también existen las opciones de fuentes abiertas frente a las soluciones cerradas.

En esta sección se pretende dar una visión general sobre la utilización de aplicaciones, herramientas, y en general, cualquier desarrollo en fuentes abiertas que cubra algún aspecto del desarrollo operacional en

el mundo empresarial.

De todas las empresas participantes en el estudio y que utilizan Software de Fuentes Abiertas, un 60% de las empresas utiliza aplicaciones de fuentes abiertas. En la siguiente gráfica (Figura 4.19) se muestra la distribución de uso por tipo de aplicaciones, que ha sido agrupadas para facilitar su identificación y clasificadas en función de su generalidad o especificidad dentro de la operativa de la empresa. Se han considerado nueve grupos básicos:

- **Soluciones generales para la empresa:**
 - Navegadores
 - Soluciones Ofimáticas

- **Soluciones técnicas específicas:**
 - Programación, que incluye desde entornos específicos de programación, lenguajes de programación, etc.
 - Base de datos
 - Herramientas de dibujo
 - Aplicaciones para error tracking.
 - Aplicaciones para control de versiones.
 - Aplicaciones de simulación.
 - Aplicaciones para librerías específicas.

En líneas generales se puede observar que las soluciones ofimáticas han tenido un importante auge (31%), al igual que aquellas aplicaciones más técnicas como pueden ser las relativas a entornos de programación (32%), control de versiones (12%), etc. Cabe resaltar el hecho de que en los navegadores el porcentaje aún está lejos de alcanzar el éxito de las herramientas ofimáticas, con sólo un 7% de utilización. A continuación se va a realizar un análisis más detallado de cada uno de los grupos.

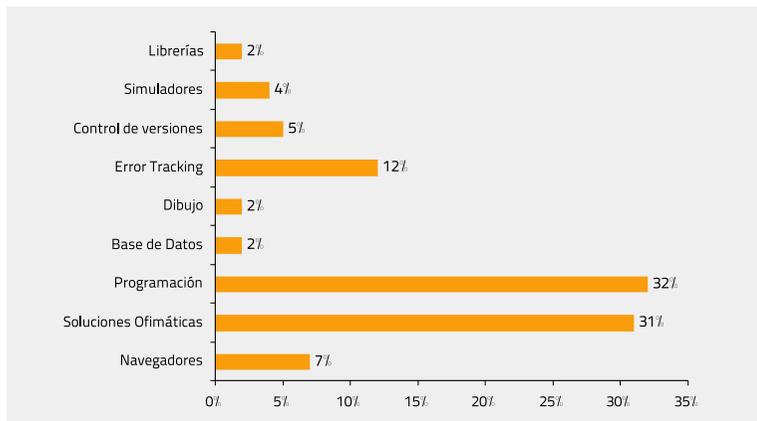


Figura 4.19: Aplicaciones

Las soluciones ofimáticas son utilizadas por un 31% de las empresas entrevistadas. Dentro de las aplicaciones ofimáticas, destaca el uso de OpenOffice (16%). Otras aplicaciones que se han incluido en este apartado, son Mediawiki (5%), Alfresco (5%) y OpenProj (5%) que cubren distintos aspectos de gestión documental y de proyectos, aunque su uso es más minoritario en comparación con OpenOffice, debido a que éste, en parte, es más específico.

En cuanto a soluciones generales, es de destacar el bajo uso de los navegadores en comparación con las soluciones ofimáticas, tan sólo el 7% de las empresas utilizan navegadores de fuentes abiertas. Dentro de esta categoría, las empresas utilizan principalmente Firefox (5%) y Chrome (2%).

Así mismo, hay que señalar la alta proporción de utilización de aplicaciones relacionadas con programación, ya sean entornos integrados, como lenguajes, y otro tipo de aplicaciones. Hay que destacar el uso de Eclipse y Apache, utilizado por un 12% y un 9%, de empresas respectivamente. Otras aplicaciones son utilizadas en menor medida (JBOS, Cmake, MinGW y PHP todos ellos utilizados por un 2% de las empresas).

Otra área de aplicaciones que está experimentando un gran auge en los últimos tiempos, son todas aquellas aplicaciones relacionadas con el "error tracking" y el control de versiones. Gracias a estas herramientas se reduce considerablemente el tiempo necesario para la depuración de programas y el valor añadido de mejora de calidad del producto entregado. Dentro del desarrollo de los sistemas empotrados se trata de un aspecto crítico en algunas de las fases del desarrollo. Los porcentajes de las empresas que utilizan este tipo de aplicaciones así lo muestra, con un 12% y un 5% respectivamente. Dentro de las herramientas de "error tracking", destacan el uso de Bugzilla y Gforge (usadas por un 5% de las empresas), y MantisBT (utilizada por un 2% de las empresas). Para el control de versiones, destaca el uso de Tortoise SVN, utilizada por un 5% de las empresas.

Es importante destacar la gran actividad y dinamismo existente en la comunidad de fuentes abiertas. En este sentido, es curioso observar el hecho de que existen soluciones muy específicas para problemas muy concretos, como es el caso de aplicaciones de fuentes abiertas para entornos de simulación dentro de los sistemas empotrados, como es el caso de Gacebo y SUGAR, utilizados por un 2% de las empresas entrevistadas. Este porcentaje refleja un uso mínimo, puesto que se trata de soluciones muy concretas que cubren necesidades muy específicas que no todas las empresas necesitan. En la misma línea, se puede encontrar el uso de librerías muy específicas, como es el caso de QT, utilizada asimismo por un 2% de las empresas. De hecho, en la gráfica se puede observar como el uso de librerías específicas al igual que aplicaciones de simulación, es del orden del 6% y del 2% respectivamente, porcentajes mínimos respecto al global de empresas.

En la gráfica puede observarse una importante distribución en el tipo de aplicaciones utilizadas, ya que el espectro de necesidades que cubren es muy variada. No obstante, cabe destacar el mayoritario uso de

OpenOffice como solución ofimática (16%) para la empresa. Esta aplicación es una solución y alternativa real y competitiva respecto a las versiones cerradas, tanto por reducción de costes, como por las alternativas de posibilidad de añadir extensiones específicas para cubrir todas las necesidades de las empresas. Otra ventaja de su uso es la compatibilidad con todos los sistemas operativos, así como mejoras en temas de rendimiento y del uso de los recursos de memoria. Además su interfaz es de uso sencillo para usuarios no expertos.

Por otro lado es interesante destacar dos proyectos que están en un gran auge actualmente, Eclipse y Apache. Eclipse se ha convertido en los últimos tiempos en uno de los principales referentes en los entornos de desarrollo integrados, con un 12% de utilización en las empresas. Su gran versatilidad, y posibilidades de integración con otras aplicaciones (como es el caso de JBOSS y TortoiseSVN) y extensiones propias, lo hacen idóneo para su adaptación e inclusión dentro del ciclo de desarrollo de un sistema empotrado.

En este sentido, el valor añadido del entorno Eclipse, no sólo es la posibilidad de los IDEs, sino la posibilidad de añadir una gran cantidad de aplicaciones, como es el caso del control de versiones, integración, multiplataforma, etc, de una manera totalmente transparente y con comunidades de soporte que aportan avances y mejoras cada poco tiempo.

Apache, por su parte con un 9%, se encuentra íntimamente ligado al desarrollo del popular servidor HTTP Apache. Sin embargo, hoy en día existen una gran cantidad de proyectos bajo la Apache Software Foundation que amplían el espectro de necesidades de mercado a cubrir. En ambos casos, la forma de trabajo y las posibilidades de integración de las herramientas bajo cualquier sistema hacen que tanto Apache como Eclipse sean aplicaciones de referencia. Además, poseen comunidades muy activas y grupos de trabajo oficiales, que día a día aumentan sus

posibilidades. Otro de los grandes puntos a favor de cualquiera de ellas es su alto grado de integración con muchas otras aplicaciones, abriendo mucho más el abanico de su utilización.

Debido a lo específico de los sistemas empotrados como área y de cada producto en particular, las necesidades varían en gran medida en las distintas empresas con lo que las herramientas específicas utilizadas por cada una de ellas varía significativamente. De todos modos es de destacar la existencia de productos de una alta especificidad, capaces de cubrir estas necesidades tan particulares, como puede ser el desarrollo de entornos de simulación, es el caso de Gacebo y Sugar MeMS simulator.

Las empresas valoran muy positivamente la integración y la unificación de los entornos de desarrollo, como es el caso de Eclipse. De la misma forma, se contemplaría como objetivo a alcanzar la integración de todo tipo de herramientas del ciclo de desarrollo en un único marco; editores de código fuente, depuradores, aplicaciones ofimáticas y extensiones como es el caso de la wiki, gestión documental, control de versiones, etc.

Es de destacar que en ocasiones las empresas comentan que la utilización de sistemas cerrados se debe en parte a que ya tienen formado a personal cualificado para el uso de herramientas cerradas, y su adaptación a una nueva aplicación supone tiempo y esfuerzo. Sin embargo, hay que resaltar que la mayoría de las empresas que utilizan Software de Fuentes Abiertas ven un claro potencial en este tipo de herramientas desde una perspectiva de reducción de costes de mantenimiento y como un valor estratégico para la empresa.

El enfoque presentado se sitúa en la línea de las aplicaciones y herramientas utilizadas en el desarrollo de sistemas empotrados. A nivel de aplicación dentro del sistema empotrado como se indica en la sección 4.1.6, es más complicado encontrar soluciones de fuentes abiertas; las empresas

participantes no han señalado la definición de aplicaciones de fuentes abiertas dentro de sus productos, aunque en el mercado es posible encontrar este tipo de soluciones:

- **Bienes de consumo**
 - MAKE
 - Rommba
 - Chumby
- **Hogar**
 - TV-B-G
 - Open IMS
- **Comunicaciones**
 - LART
 - RFID Guardian
 - Maemo
 - Molin
 - WRT54GL
- **Control Industrial**
 - Linux PLC
 - SADL
 - Blitech
- **Medioambiente**
 - SunSPOT
 - Salud
 - OpenEEG

4.2.5.3 >> Librerías

Debido a la especificidad de algunos dominios, han ido apareciendo soluciones a problemas determinados que han dado lugar al desarrollo de librerías específicas dentro del ámbito de las fuentes abiertas. La gran ventaja de estas librerías es las características y funcionalidades específicas, además de poseer un gran nivel de integración con otras aplicaciones.

Los sistemas emportrados no son una excepción en este sentido, y en la encuesta se ha podido observar que un 30% de las empresas hacen uso de librerías específicas dentro del ciclo de desarrollo de sus sistemas.

En el análisis de esta sección, se ha decidido agrupar las librerías en familias basadas en el tipo de soluciones que proveen. En este caso se han detectado las siguientes familias o grupos de librerías:

- Librerías de procesado de imagen.
- Librerías de interfaces gráficos de usuario (GUI).
- Librerías de comunicaciones, tiempo real, etc.
- Librerías de interoperabilidad entre aplicaciones.
- Librerías para aplicaciones web.
- Otro tipo de librerías.

En la siguiente gráfica (Figura 4.20) puede observarse los porcentajes de empresas que hacen uso de al menos alguna librería dentro de cada familia. Es de destacar, que el 13% de las empresas utilizan librerías encargadas del manejo de la interoperabilidad de aplicaciones, teniendo el resto de familias una distribución de uso bastante equitativa en las empresas.

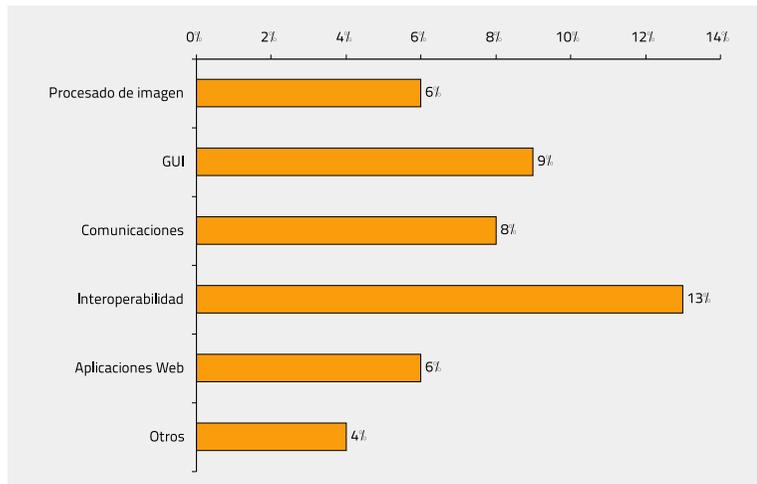


Figura 4.20: Librerías

Dentro del grupo de Interoperabilidad de aplicaciones, destaca el uso de la librería TAO/AC, siendo utilizada por un 7% de las empresas. Esta librería ofrece ventajas para la determinación del Quality of Service de aplicaciones, así como funcionalidades de tiempo real. De hecho, es similar a CORBA pero con restricciones de tiempo real. Dentro también de este grupo, existen otras alternativas que son igualmente utilizadas por las empresas, como es el caso de Maemo, Open DDS y NetSNMP, siendo usadas cada una de ellas en el 2% de las empresas participantes.

Los interfaces gráficos de usuario (GUI) han sido otro punto de trabajo muy importante, y donde se han hecho grandes esfuerzos. En la gráfica se puede observar que el 9% de las empresas utilizan alguna librería de este grupo. En este sentido, cabe destacar el uso de librerías como Xfree86 y X y las de Gnome (GTK y Clutter), muy conocidas y que siguen siendo utilizadas cada una de ellas por el 2% de las empresas. A pesar de su fama, existen otras alternativas que han ganado posiciones en los últimos tiempos gracias a su versatilidad y especificidad, como es el

caso de la librería QT que es utilizada en el 5% de las empresas.

Otro campo de significativo interés dentro de los sistemas empujados se encuentra englobado en los grupos de comunicaciones y aplicaciones web, que son utilizados en un 8% y 6% de las empresas, respectivamente. Dentro del grupo de las comunicaciones, existe una gran variedad de librerías para el manejo no sólo de comunicaciones, comunicaciones móviles, sino que se proveen de soluciones eficaces teniendo en cuenta restricciones de tiempo real. El uso de este tipo de librerías está más o menos distribuido de manera equitativa entre FreeDesktop, Moblin, LiMo, Iwlp, con un 2% de las empresas para cada uno de ellos. Por otra parte, en el desarrollo de aplicaciones web, destacan el uso de WebKit, Django y Struts, cada una de ellas utilizadas por un 2% de las empresas.

Desde el punto de vista del procesamiento de imagen (6% de las empresas las utilizan), las librerías más utilizadas son Cimage, OpenCV y OpenGL, con un 2% de las empresas, respectivamente. Tanto Cimage como OpenGL ofrecen soluciones para el procesamiento de imágenes en 2D y 3D, teniendo en cuenta cuestiones de rendimiento y eficiencia.

Otras librerías más específicas con un uso más minoritario se encuentran definidas dentro del grupo Otros, donde hay un 4% de empresas que utilizan algún tipo de librerías muy específicas para problemas muy concretos. En este grupo cabe destacar, SICL y ActivePerl que son utilizadas por un 2% de las empresas, respectivamente.

En resumen, se puede concluir que las librerías cubren una gran variedad de elementos y funcionalidades, hardware o software. En las respuestas de las empresas se ha observado que la mayoría utilizan diferentes tipos de librerías para cubrir los distintos aspectos del desarrollo del sistema empujado.

4.2.5.4 >> Licencias

Las licencias han aportado una manera eficaz de colaboración para el control de los derechos de autor y un control sobre la distribución, copia y modificación del software. De entre todas las empresas entrevistadas, sólo un 28% de ellas reconoce el uso de las licencias dentro de su proceso productivo.

En la siguiente gráfica (Figura 4.21) se muestran los distintos tipos de licencias que se han identificado y que son utilizadas en las empresas: GPL, LGOL, MGPL, Apache, CC1, BSD, y la modified BSD.

Es de destacar que la mayoría de empresas se decanta por el uso de la licencia GPL y modificaciones (aproximadamente un 39% de las empresas participantes), como es el caso de la LGPL y la MGPL. La GPL es utilizada por un 21% de las empresas, la LGPL por un 16% y la licencia MCPL por un 2% de las empresas. MPGL es una licencia de uso exclusivo para programas realizados en lenguaje de programación ADA, de ahí que su porcentaje de utilización sea tan bajo.

El resto de licencias son de uso más minoritario, destacando quizás entre todas ellas el uso de la licencia BSD (7% de las empresas lo utilizan) y la BSD modificada (5% de las empresas). Las licencias BSD, junto con la Apache (5% de empresas la utilizan) son menos restrictivas que las GPL, sin embargo, se encuentra mucho más extendido el uso de esta última.

Por otro lado la licencia CC (Creative Commons) no se trata en sí misma de una licencia software, sino de un modelo legal ayudado de herramientas informáticas para facilitar la distribución y uso de contenidos. De ahí que su uso no esté tan extendido (sólo es utilizado por un 2% de las empresas) como en el caso de otras licencias.

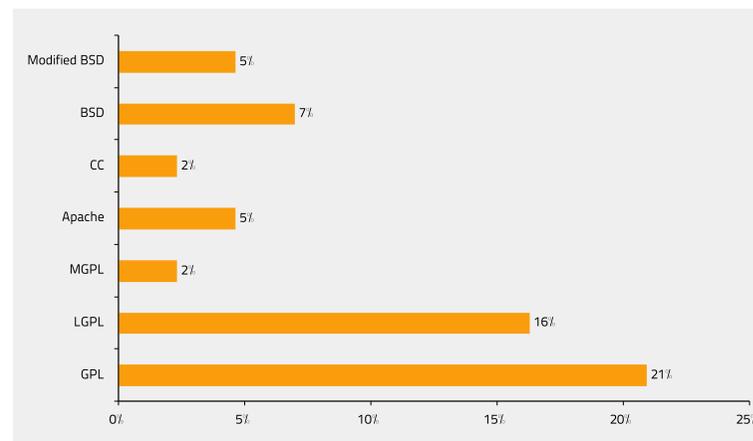


Figura 4.21: Licencias

4.2.5.5 >> Entornos de programación

Debido a la gran cantidad de tecnologías empleadas en el desarrollo, así como de recursos implicados, resulta de gran interés la utilización de entornos integrados que faciliten la comunicación, implementación y desarrollo de las aplicaciones.

En la encuesta se ha observado que este tipo de entornos está siendo utilizado en un 37% de las empresas usuarias de Software de Fuentes Abiertas. En la siguiente gráfica (Figura 4.22) se muestra la distribución de la utilización de los mismos dentro de las empresas.

Los entornos destacados son Eclipse y Gnome, siendo utilizados por un 23% y 19% de las empresas, respectivamente. En este sentido, Eclipse es uno de los frameworks más conocidos y dinámicos, gracias a la flexibilidad que proporcionan la gran cantidad de IDEs, plugins y funcionalidades disponibles, además de que el hecho de ser multiplataforma sea una gran ventaja a la hora de ser implantado en

una empresa. Otro punto a favor de Eclipse, es la gran cantidad de proyectos que actualmente alimentan y aumentan sus perspectivas y posibilidades.

Gnome (19%), aunque se suele limitar su uso a entornos GNU/Linux, ofrece a las empresas una alta confianza y posibilidades en su uso. Otras alternativas se encuentran más alejadas, tal y como se observa en la gráfica, es el caso de GCC, Kdevelopper, TopCased, J2EE o incluso Netbeans, directo competidor de Eclipse hace algún tiempo, cada uno de ellos utilizado por un 2% de empresas.

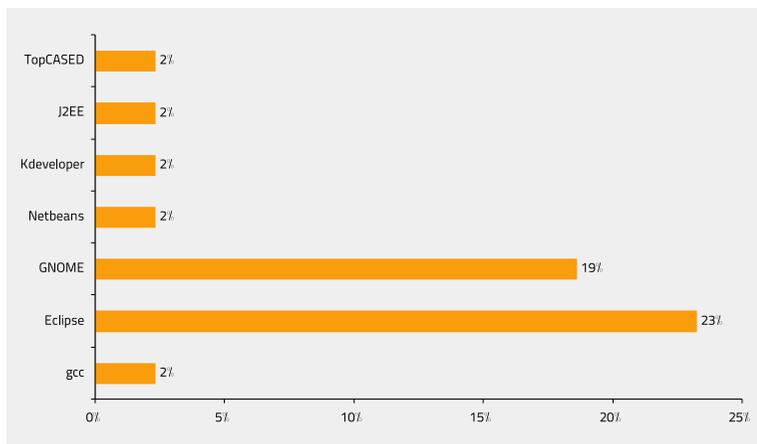


Figura 4.22: Entornos

4.2.5.6 >> Herramientas de integración

La integración es una de las partes principales dentro del desarrollo de sistemas empotrados, debido a la gran variedad de elementos que se tienen que implementar, comunicar e integrar. Existen diversas soluciones de fuentes abiertas que son utilizadas por las empresas para la realización de esta fase del ciclo de desarrollo del producto. A pesar de la gran

importancia de este apartado dentro del desarrollo, este tipo de herramientas de Software de Fuentes Abiertas sólo es utilizado por 23% de las empresas tenidas en cuenta en este capítulo. Una posible explicación es que la integración no es realizada por gran parte de estas empresas, ya que algunas de ellas se centran en otras fases del ciclo de desarrollo. Otro posible motivo es la posición de dominio que aún mantienen las herramientas cerradas en este ámbito.

En la gráfica que se muestra a continuación (Figura 4.23) se muestran las herramientas de integración de fuentes abiertas utilizadas entre las empresas participantes en la encuesta.

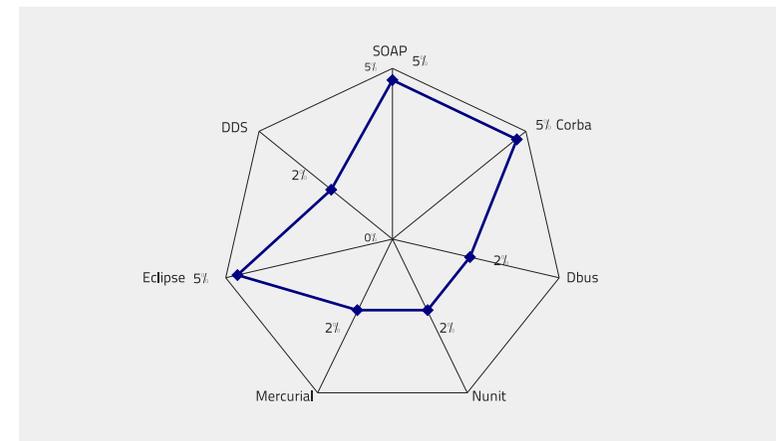


Figura 4.23: Herramientas de integración

Hay que destacar Corba, SOAP y Eclipse como las herramientas más utilizadas por las empresas, siendo utilizadas cada una de ellas por un 5% de las empresas. Otras herramientas que salen reflejadas en el estudio, aunque de uso más minoritario, son Dbus, Nuint, Mercurial y Dbus (implantadas cada una de ellas en el 2% de las empresas participantes).

4.3

>>Actividades de I+D relacionadas con Software de Fuentes Abiertas en el ámbito de sistemas empotrados

4.3.1

>>Actividades de colaboración I+D

Esta sección muestra las actividades de colaboración I+D que las empresas realizan relacionadas con el Software de Fuentes Abiertas.

A las empresas se les pidió que identificaran los proyectos de colaboración en I+D, relacionados con el Software de Fuentes Abiertas, en los que han participado, según seis categorías identificadas:

- Colaboración en proyectos I+D internacionales.
- Colaboración en proyectos I+D nacionales.
- Colaboración en proyectos con empresas a nivel internacional.
- Colaboración en proyectos con empresas a nivel nacional.
- Colaboración con Universidades.
- Colaboración con Centros Tecnológicos.

Las empresas han podido seleccionar todas las opciones que han considerado pertinentes.

Vemos como el 28% de las empresas que declaran utilizar Software de Fuentes Abiertas en alguna fase del desarrollo de su producto señalan que colaboran en proyectos con otras empresas a nivel nacional. Este

tipo de colaboración es la que más empresas declaran realizar. La colaboración en proyectos con empresas a nivel internacional y la colaboración con Universidades han sido indicadas por un 18% de empresas.

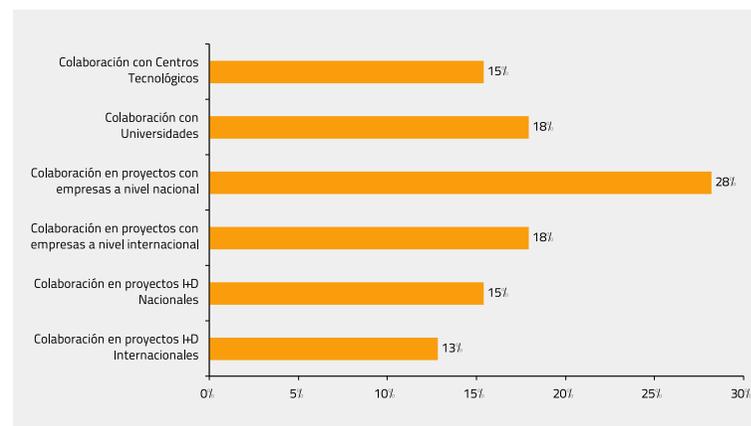


Figura 4.24: Actividades de colaboración I+D

Un 15% de las empresas han declarado que colaboran con Centros Tecnológicos y en proyectos de I+D nacionales. Finalmente, un 13% de empresas colaboran en proyectos I+D internacionales.

Los porcentajes obtenidos relativos a la colaboración I+D o colaboración entre empresas no son significativamente altos. La competitividad entre empresas de sistemas empotrados dentro del mismo sector de actividad y la necesidad de diferenciación respecto al competidor son una barrera más en la participación de actividades de colaboración.

En este aspecto, cabe reseñar cómo la situación dentro de la cadena de valor explica los porcentajes de colaboración entre empresas a nivel nacional. La participación en un mismo proyecto de empresas que a su vez son proveedoras y clientes de otros participantes facilita la colaboración y unión de objetivos, con una clara participación de cada una de las empresas

dentro del proyecto. La cercanía que implica que todas las empresas participantes del proyecto sean del mismo país es un factor que allana posibles dificultades. Muchas comunidades de Software de Fuentes Abiertas han surgido de países, incluso regiones y pequeñas comunidades.

La colaboración interempresarial a nivel nacional presenta este alto porcentaje, en gran parte debido al alto número de medianas y pequeñas empresas contabilizadas. El acceso a colaboraciones internacionales requiere un esfuerzo económico y de recursos humanos que muchas de estas pequeñas empresas se ven incapaces de afrontar. El acceso a proyectos compartidos con Universidades y Centros Tecnológicos permiten a las pequeñas empresas acceder a tecnologías incipientes, lo cual sería muy difícil si tuvieran que afrontar por sí solas este reto, ya que sería excesivamente oneroso.

Las soluciones de fuentes abiertas han estado auspiciadas en muchas ocasiones por Universidades o por Centros Tecnológicos, lo que puede posibilitar a las pequeñas empresas su acercamiento a este tipo de soluciones, muchas veces salvando barreras de desconocimiento iniciales.

Cabe mencionar en este apartado, como una posible línea de trabajo para allanar los obstáculos relativos a la participación en proyectos de I+D europeos con aportes de software de fuente abierta, que las empresas potencialmente candidatas reciban formación en los aspectos significativos de los acuerdos tipo DESCA (DEvelopment of a Simple Consortium Agreement) e IPCA (Integrated Project Consortium Agreement). Ambos tipos de acuerdo de consorcio incluyen mecanismos de protección de la propiedad intelectual, contemplando las modalidades de software de fuente abierta como elementos clave a la hora de fomentar la colaboración entre las empresas de un consorcio.

En particular el modelo IPCA es interesante, por dos motivos:

1- Es el modelo más habitual en el caso en el que los participantes sean empresas (el caso más interesante para las empresas de software de fuente abierta).

2- IPCA incluye cláusulas para utilizar el software de código abierto tanto como background (elementos a aportar al proyecto) como foreground (elementos resultantes del proyecto).

IPCA ha sido desarrollado por la Asociación Europea de la Industria de las Tecnologías de la Información y las Telecomunicaciones (EICTA). Este grupo, que representa a las industrias de tecnología digital, posee una larga tradición en la redacción de modelos de acuerdo de consorcio para sectores específicos. Para el 7PM, EICTA ha utilizado el modelo IPCA creado para el 6PM, asegurándose de que el modelo del 6PM cumplía con los objetivos y los requisitos jurídicos del 7PM. Dado que IPCA está siendo aplicado como el acuerdo estándar en la Joint Undertaking ARTEMIS de sistemas empotrados, en forma de APCA, se hace necesario que los agentes empresariales que buscan desarrollar sus capacidades en proyectos de I+D europeos conozcan las provisiones de IPCA/APCA en relación al software de fuente abierta.



Las soluciones de fuentes abiertas han estado auspiciadas en muchas ocasiones por Universidades o por Centros Tecnológicos, lo que puede posibilitar a las pequeñas empresas su acercamiento a este tipo de soluciones, muchas veces salvando barreras de desconocimiento iniciales.

4.4

>> Comercialización y distribución de sistemas empotrados basados en Software de Fuentes Abiertas

La clave dentro de la estrategia de comercialización y distribución de los sistemas empotrados está condicionada en gran parte por el valor añadido de los productos diseñados. El Software de Fuentes Abiertas está día a día ganando mejores puestos en valoración, no sólo por parte de las empresas, sino también por parte de los clientes, ya que está ofreciendo una amplia gama de herramientas adaptables a los productos que se desarrollan en las empresas.

4.4.1 >> Tipología de clientes

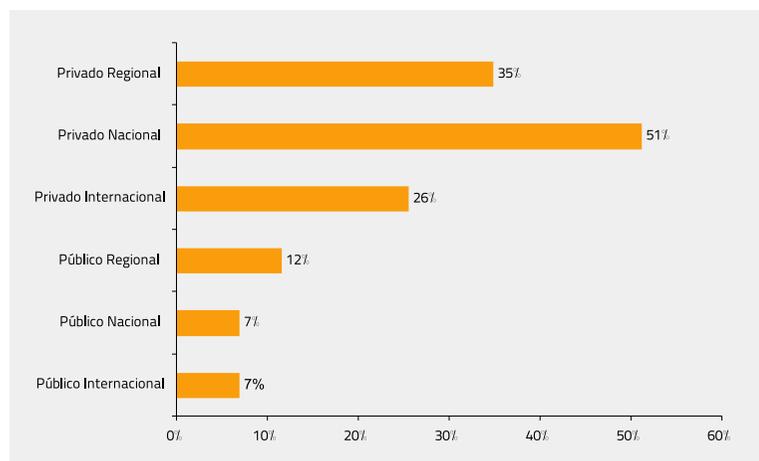


Figura 4.25: Tipología de clientes

En la Figura 4.25 se puede observar como la actividad comercial de empresas de sistemas empotrados basados en Software de Fuentes Abiertas está dirigida principalmente al sector privado. Un 51% de las empresas han señalado que desarrollan parte de su actividad comercial en el sector privado nacional, un 35% de las empresas en el sector privado regional y un 26% en el sector privado internacional.

Resulta indudable que, en los últimos años, el Software de Fuentes Abiertas ha adquirido un creciente protagonismo y desarrollo. En los distintos sectores, tanto públicos como privados, se está empezando a tomar conciencia de esto y están apostando por la implantación de este tipo de software. El problema lo encontramos a la hora de combinar sistemas empotrados y Software de Fuentes Abiertas.

Todavía, el cliente de este tipo de sistemas no está totalmente convencido sobre las herramientas abiertas existentes en el mercado, por lo que se decanta por la compra de sistemas empotrados basados en software cerrado, los cuales encajan mejor con sus necesidades.

En las Administraciones Públicas esto se acentúa, ya que la necesidad de un sistema ampliamente fiable prima sobre cualquier otra cosa. Además, los sectores públicos son reticentes a los cambios, más si los sistemas que poseen funcionan. Por este motivo, la evolución hacia este tipo de sistemas basados en Software de Fuentes Abiertas será más gradual en el sector público que en el privado.



La actividad comercial de empresas de sistemas empotrados basados en Software de Fuentes Abiertas está dirigida principalmente al sector privado.

4.4.2

>>Valoración del Software de Fuentes Abiertas como seña de identidad de la empresa

En la siguiente gráfica (Figura 4.26) se puede observar como las empresas valoran que el uso de Software de Fuentes Abiertas es un factor "muy importante" (27%) o "bastante importante" (30%) con respecto a la seña de identidad de su producto o de la propia empresa, es decir, que dar a conocer que sus productos están basados en Software de Fuentes Abiertas a sus clientes forma parte importante de su estrategia de comercialización. Aunque todavía, sigue habiendo un porcentaje relativamente alto (34%) que apenas lo valora como valor añadido al producto de cara a la estrategia comercial.

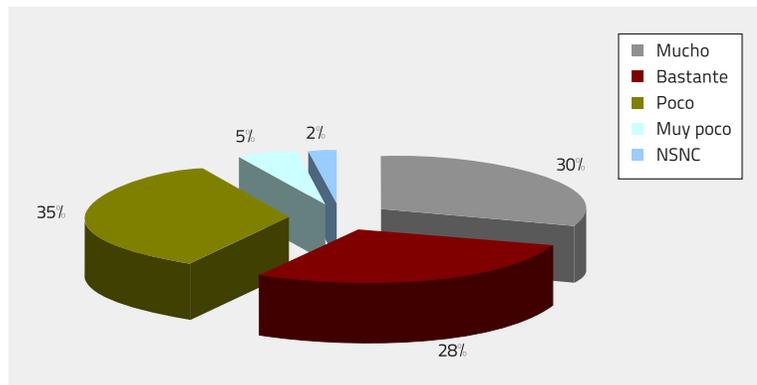


Figura 4.26: Valoración del Software de Fuentes Abiertas

El ahorro en el coste de fabricación está entre las principales ventajas derivadas del uso de Software de Fuentes Abiertas en los desarrollos de sistemas empotrados. Esto es un atractivo para las empresas a la hora de utilizar este tipo de herramientas en los desarrollos de sus productos. Aún teniendo en cuenta esto, el principal valor añadido que la empresa puede ofrecer a su cliente es la transparencia del producto, el cliente tiene acceso a cómo está hecho dicho producto. Esta claridad de cara al cliente, genera confianza, lo que implica un acceso al mercado más eficiente.

A pesar de que más de la mitad de las empresas consideran que el uso de Software de Fuentes Abiertas en sus productos es un valor añadido que ofrecen a sus clientes, encontramos que más de la mitad de las empresas no publicitan el hecho de usar Software de Fuentes Abiertas en la producción, de manera general (51%).

4.4.3

>>Publicidad de los productos

Las empresas españolas que desarrollan sistemas empotrados basados en Software de Fuentes Abiertas, suelen participar activamente en congresos, comunidades y blogs especializados. Esto fomenta el conocimiento de los productos que desarrollan pero no es suficiente. La necesidad de que el departamento comercial tenga una cualificación apropiada es primordial para hacer una publicidad adecuada del producto.

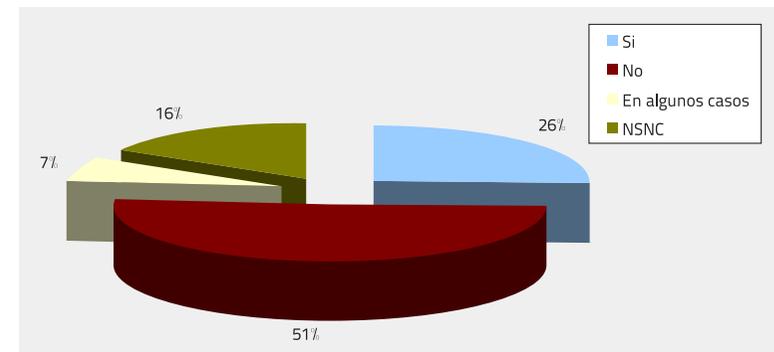


Figura 4.27: Publicidad



La mitad de las empresas consideran que el uso de Software de Fuentes Abiertas en sus productos es un valor añadido que ofrecen a sus clientes; no obstante, no publicitan dicho uso.

4.5

>> Recursos Humanos dedicados a sistemas empotrados basados en Software de Fuentes Abiertas

Esta sección va a analizar diferentes aspectos del Software de Fuentes Abiertas y los recursos humanos que se ven implicados en el mismo.

4.5.1 >> Formación

Esta sección analiza cómo se ha adquirido la formación del personal implicado en el desarrollo de Software de Fuentes Abiertas en el ámbito de sistemas empotrados, si ésta ha sido proporcionada por la empresa, o adquirida por el personal antes de su incorporación a la empresa, y/o una vez contratado por la misma. Las empresas podían elegir múltiples opciones de respuesta.

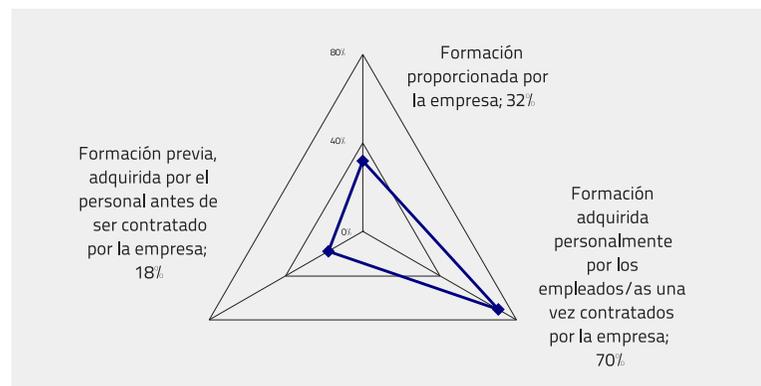


Figura 4.28: Formación de los recursos humanos

Como vemos en la Figura 4.28 un 70% de las empresas ha declarado que los recursos humanos que trabajan con Software de Fuentes Abiertas tenían formación en este campo adquirida personalmente; un 32% de las empresas indica que proporciona formación sobre Software de Fuentes Abiertas a sus empleados; finalmente, un 18% de las empresas ha requerido o a considerado conveniente dicha formación en el momento de la contratación de los recursos humanos.

A pesar del bajo porcentaje de empresas que ofrecen a su personal formación en Software de Fuentes Abiertas, hay que destacar el esfuerzo de las mismas en este tipo de formación.

Con el auge del Software de Fuentes Abiertas en aplicaciones de escritorio, existen muchas empresas dedicadas a la formación en este ámbito, pero existe la percepción por parte de las empresas de que esta formación todavía no ha alcanzado la especialización que los sistemas empotrados requieren según sus ámbitos de actividad comercial.

Se sigue confiando el desarrollo de estas soluciones en una formación adquirida mayoritariamente de manera autodidacta y que suele venir dada por la experiencia conseguida en desarrollos anteriores o por una motivación personal. Esto solamente facilita la reducción de tiempos de desarrollos cuando estas soluciones están extensamente implantadas en la empresa y el personal ha dedicado mucho tiempo a dicha autoformación.

El conocimiento y experiencia en Software de Fuentes Abiertas proporciona un valor añadido a la formación de los desarrolladores en este tipo de soluciones, lo que incita a personal sin experiencia previa a la búsqueda de formación en este campo. Sin embargo, el bajo porcentaje de empresas cuyos empleados han sido contratados teniendo ya una

formación en Software de Fuentes Abiertas, indica que no es una característica imprescindible. Se puede valorar en la contratación esta experiencia previa en una mayor medida cuando la empresa da sus primeros pasos en la adopción de Software de Fuentes Abiertas, de cara a no alargar en exceso los tiempos de desarrollo.

Comentarios interesantes que se han podido recoger por parte de las empresas apuntan a que la formación en Software de Fuentes Abiertas "se valora en la selección de personal, aunque no es decisiva y en caso de que no se tengan experiencia trabajando con este software, se les enseña bajo demanda."

4.5.2 >>Experiencia del personal dedicado a sistemas empotrados basados en Software de Fuentes Abiertas

Esta sección analiza la experiencia del personal que trabaja en las empresas en el desarrollo de Software de Fuentes Abiertas, para lo cual se ha pedido a las empresas que indiquen el tiempo medio de experiencia con Software de Fuentes Abiertas del personal involucrado.

Se han analizado las respuestas acotándolas en los siguientes rangos:

- Menos de 2 años
- Entre 2 y 5 años
- Entre 5 y 10 años
- Más de 10 años

Como se aprecia en la Figura 4.29 un 51% de las empresas afirma que los equipos dedicados al Software de Fuentes Abiertas cuentan con una experiencia media de entre 2 y 5 años. El 28% de empresas señalan una

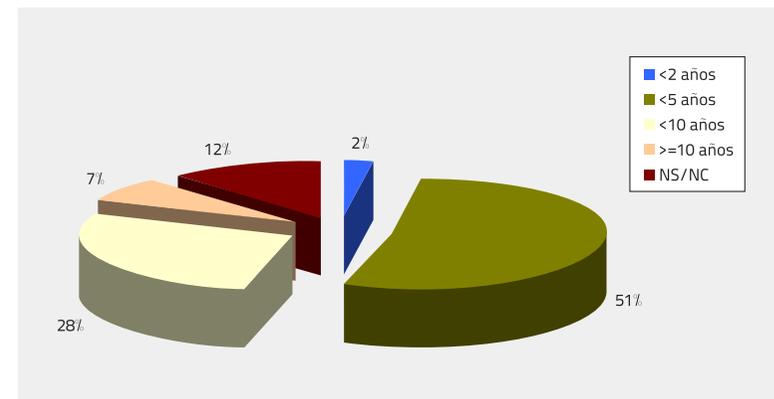


Figura 4.29: Experiencia del personal dedicado a sistemas empotrados basados en Software de Fuentes Abiertas

experiencia media de entre 5 y 10 años, siendo únicamente el 7% de empresas las que declaran que sus equipos cuentan con una experiencia superior a los 10 años.

Se aprecia claramente en estos resultados que la experiencia en Software de Fuentes Abiertas es un factor decisivo en su implantación. Solo un 2% de las empresas declaran una experiencia menor de 2 años. El mayor porcentaje se da entre los 2 y 5 años, en muchos casos el tiempo de experiencia básico en el desarrollo de nuevas soluciones y productos.

La participación en el estudio por parte de la empresa ha sido en gran medida a través de gerentes, directores técnicos y jefes de proyecto. La experiencia de los empleados situados en estos puestos de trabajo suele ser bastante alta, se les exige, entre otras características, un dominio de las tecnologías utilizadas. En muchos casos, cuando el personal empleado con experiencia en el desarrollo de Software de Fuentes Abiertas alcanza estos puestos con poder de decisión, la adopción de nuevas soluciones de fuentes abiertas se ve favorecida por

este mayor conocimiento por parte de los órganos decisorios en cuestiones técnicas de las empresas.

Algunas empresas indican que la experiencia señalada en la encuesta se refiere a la del personal más experto, y cómo "los nuevos ingenieros empiezan directamente utilizando herramientas de Software de Fuentes Abiertas" lo que permite una capacitación que se consigue por medio de la experiencia y el contacto con personal más experimentado.

Del 12% de empresas que han contestado NS/NC a esta pregunta, la mayoría indican que el personal "es muy variado" en cuanto a experiencia en este ámbito. Así, la unión de personal que cuenta con experiencia en el Software de Fuentes Abiertas, con personal menos experimentado permite la auto capacitación durante el desarrollo de los productos. Esta situación se da muy habitualmente ya que favorece la formación sin perjudicar en exceso el tiempo de desarrollo.

4.6

>> Colaboración con las comunidades de desarrollo

Una característica del Software de Fuentes Abiertas es la implicación de comunidades en la implementación y desarrollo de soluciones de fuentes abiertas, así como su verificación y publicidad. A partir de ellas surgen nuevos proyectos que abarcan enfoques nuevos a problemáticas que

no estaban cubiertas inicialmente, permitiendo así el desarrollo continuo.

En esta sección se analiza la colaboración de las empresas que realizan Software de Fuentes Abiertas con estas comunidades de desarrollo. Preguntadas las empresas por su relación con las comunidades, se han definido las siguientes posibles respuestas, pudiendo seleccionar varias de ellas.

- Se adoptan desarrollos realizados por comunidades.
- Se realizan contribuciones y desarrollos a comunidades.
- Hay una colaboración activa, con adopción de desarrollos, contribuciones y participación en la definición de objetivos, organización y apoyo a la misma.
- No hay implicación con comunidades.

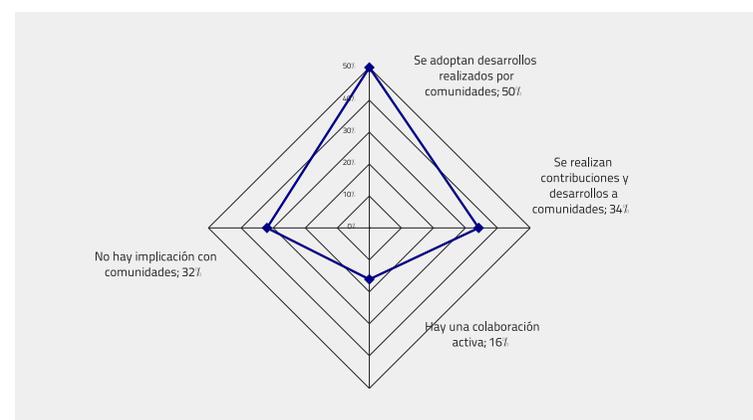


Figura 4.30: Colaboración con comunidades de desarrollo

En total un 50% de las empresas han declarado que adoptan desarrollos realizados por comunidades, un 34% aportan desarrollos a las comunidades, un 16% indica que mantienen una colaboración activa, con adopción de desarrollos, contribuciones y participación en la definición de objetivos, organización y apoyo a la misma. Finalmente, un 32% de las empresas no mantienen ninguna implicación con las comunidades de Software de Fuentes Abiertas.

Se puede apreciar en estos resultados cómo la interacción con comunidades se da principalmente en la adopción de desarrollos, que es el inicio básico para las empresas de la relación con la comunidad. El porcentaje baja sensiblemente en la aportación de desarrollos y la colaboración activa en la comunidad. Cabe destacar cómo las micro-empresas, mientras que afirman participar en las comunidades mediante la adopción y aportación de desarrollos, no participa de forma activa en ellas en la definición de objetivos, organización y apoyo a la misma. Esto puede ser debido a los bajos recursos que este tipo de empresa puede dedicar a este tipo de actividad, poco rentable económicamente frente al esfuerzo que requiere, al menos en un corto plazo.

Las PYMEs se definen básicamente como usuarias pasivas de las comunidades, centrandó toda su colaboración en la adopción de desarrollos. Como se ha recogido en algún comentario, como mucho "se reportan bugs encontrados".

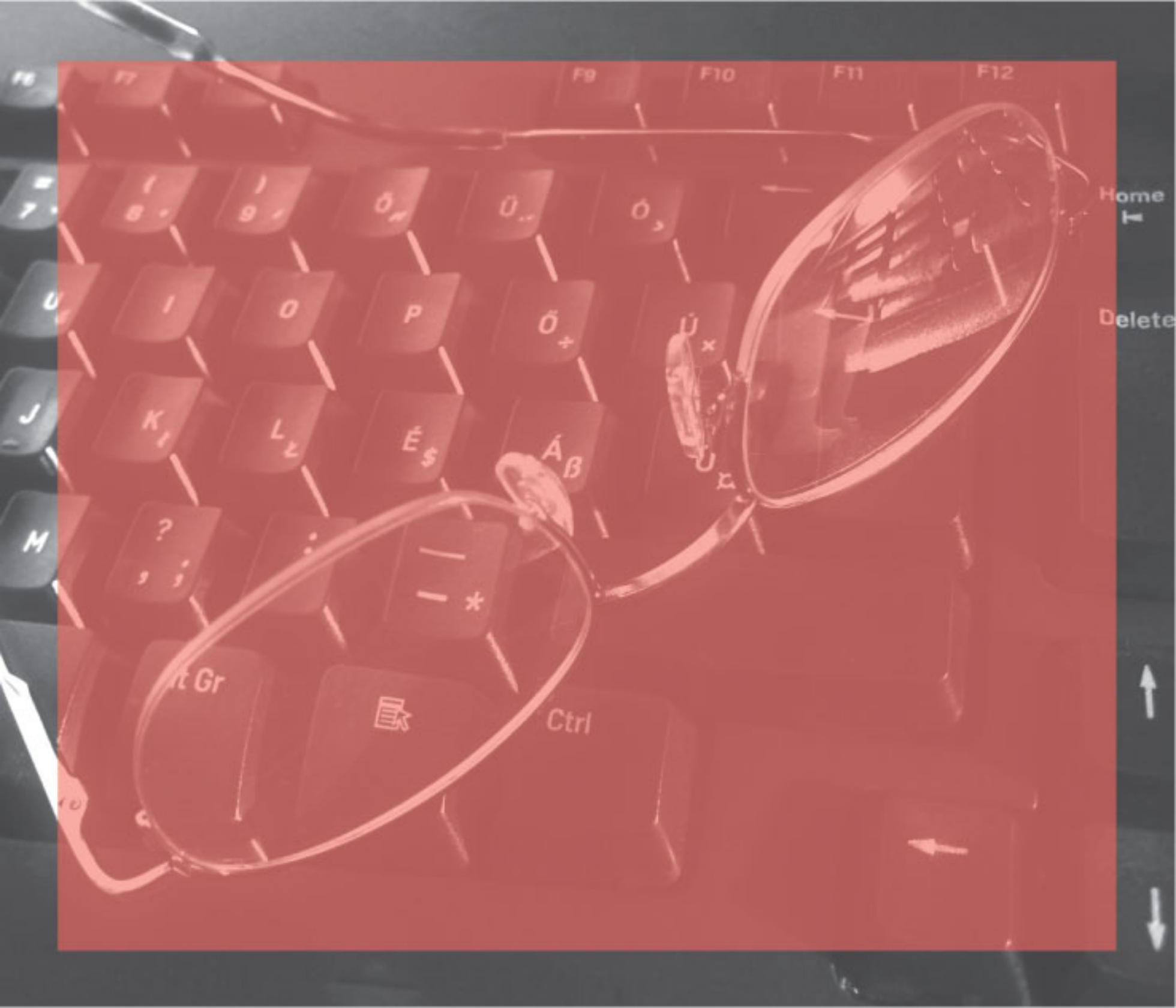
El alto porcentaje de grandes empresas que adoptan desarrollos de las comunidades, proporciona a éstas estabilidad y difusión de sus soluciones, dando una indicación de la validez de dichas soluciones. En muchas comunidades, es tan apreciado este hecho como la realización de contribuciones a la comunidad, puesto que ayuda a la eliminación de barreras culturales existentes en gran parte del tejido empresarial.



La interacción con comunidades se da principalmente en la adopción de desarrollos; el porcentaje baja sensiblemente en la aportación de desarrollos y la colaboración activa en la comunidad.

Las microempresas, mientras afirman participar en las comunidades mediante la adopción y aportación de desarrollos, no participan activamente en ellas en la definición de objetivos, organización y apoyo a la misma.

Esto puede deberse a los bajos recursos que este tipo de empresa puede dedicar a este tipo de actividad, poco rentable económicamente frente al esfuerzo que refiere, al menos en un corto plazo.



05

ANÁLISIS ESTRATÉGICO

En este capítulo se incluye un análisis estratégico, siguiendo el modelo DAFO (debilidades, amenazas, fortalezas y oportunidades), de la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados. Este análisis toma en consideración los datos concernientes a las dificultades que se han encontrado en la implantación de Software de Fuentes Abiertas dentro de las empresas encuestadas, los beneficios logrados en implantaciones pasadas, y las tendencias en su utilización dentro de las empresas españolas.

05

ANÁLISIS ESTRATÉGICO

En este capítulo se incluye un análisis estratégico, siguiendo el modelo DAFO (debilidades, amenazas, fortalezas y oportunidades), de la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados. Este análisis toma en consideración los datos concernientes a las dificultades que se han encontrado en la implantación de Software de Fuentes Abiertas dentro de las empresas encuestadas, los beneficios logrados en implantaciones pasadas, y las tendencias en su utilización dentro de las empresas españolas.

El análisis DAFO es una metodología de análisis cuyo principal objetivo es determinar las ventajas competitivas reales de una empresa sea cual sea el dominio de la misma. Este análisis permite entre otras cuestiones la definición de una estrategia empresarial competitiva enmarcada dentro del contexto determinado de la empresa, a nivel interno y a nivel externo. En la Figura 5.1 se muestra el esquema general del análisis. Como se puede observar, el análisis se realiza desde dos puntos de vista: interno y externo.

El análisis interno tiene como principal objetivo el estudio de la situación interna de la empresa. En primer lugar es necesario identificar y detectar todos los elementos internos positivos. Así los elementos positivos internos se enclavan dentro de las fortalezas. Del mismo modo, es necesario identificar aquellas carencias o debilidades, como puede ser falta o escasez de recursos, habilidades o incluso actitudes dentro de la empresa que pueden ser barreras para la mejora y buena marcha de la organización.

Dentro del ámbito de los sistemas empotrados, estas barreras pueden venir definidas por ejemplo a nivel organizativo o incluso de formación y habilidades de los propios recursos. Un análisis exhaustivo de la situación interna del objeto en estudio sobre estos dos aspectos proporciona una visión global de la capacidad estratégica y de evolución. Sin embargo, este análisis sólo proporciona una visión parcial de la situación real de la empresa. En un entorno competitivo, es de vital importancia el contexto de trabajo de la empresa en relación a su situación respecto a otras empresas y dentro del sector en el que se

encuadre. De este aspecto se encarga el análisis externo.

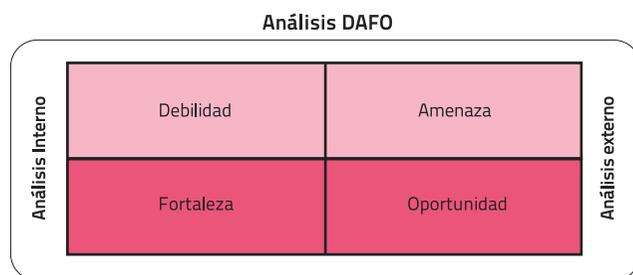


Figura 5.1: Estructura de un análisis DAFO

El análisis externo pretende mostrar el posicionamiento de la empresa frente a sus competidores y al mercado en general, también desde el punto de vista de posibles clientes. Este estudio pretende analizar por un lado todas aquellas situaciones que pueden influir de manera positiva y cuyo conocimiento puede ser aprovechado por la empresa a nivel estratégico. Esto es lo que se conoce como la identificación de oportunidades.

Por otro lado, es necesario también conocer todos aquellos elementos que pueden influir negativamente sobre la empresa, o dicho de otro modo, las debilidades. Este aspecto es crítico puesto que es necesario definir estrategias apropiadas para manejar estas situaciones para minimizar el impacto sobre el procedimiento general de la empresa. Dentro del ámbito de los sistemas empotrados, debido a la rápida evolución de sistemas, tecnologías y protocolos, este análisis se hace necesario en aras de mantener la competitividad de los productos y maximizar los beneficios.

Sin embargo, el análisis DAFO que se realiza en este apartado presenta unas características adicionales al no estar enfocado en una sola empresa. El objeto del análisis es el conjunto de empresas de sistemas empotrados que han participado en el estudio.

Este conjunto de empresas no forma un grupo homogéneo sino que además de la diversidad de sectores de actividad que abarcan, el uso de diferentes tecnologías les sitúa en diferentes posiciones cubriendo todo el rango de la cadena de valor. Es por ello que se pueden dar determinadas contradicciones entre el análisis interno y externo; al ser unas empresas proveedoras de otras, las debilidades que atañen a determinado grupo se convierten en oportunidades de otro, igualmente ocurre con las amenazas y las fortalezas. El análisis refleja cómo empresas que cubren tecnologías que abarcan toda la cadena de valor encuentran en las debilidades ajenas sus oportunidades.

Esta visión queda reflejada en la identificación de las debilidades, amenazas, fortalezas y oportunidades.



El conjunto de empresas de sistemas empotrados que han participado en el estudio no forma un grupo homogéneo sino que además de la diversidad de sectores de actividad que abarcan, el uso de diferentes tecnologías les sitúa en diferentes posiciones cubriendo todo el rango de la cadena de valor.

5.1

>> Debilidades

Las debilidades, dentro del análisis interno, pretenden determinar todas aquellas barreras y dificultades que se encuentran en la empresa. Como se ha comentado anteriormente, se han extraído una serie de debilidades que se pueden extrapolar a las empresas del ámbito de los sistemas empotrados.

A partir del análisis de los datos de la encuesta se han identificado las siguientes debilidades respecto a los sistemas empotrados y a la utilización de fuentes abiertas:

Cobertura de funcionalidades críticas de los sistemas empotrados.

Existen ciertas áreas aplicativas de los sistemas empotrados con requisitos críticos que exigen niveles de fiabilidad y de seguridad que no son adecuadamente cubiertos con el Software de Fuentes Abiertas, principalmente cuando se exige la certificación de los productos.

Documentación, soporte y mantenimiento. En algunos casos se puede encontrar herramientas software y hardware de fuente abierta sin garantías de soporte y mantenimiento. Esto se ha logrado mitigar en ciertos productos, como por ejemplo Linux y Eclipse, sobre los que ya se cuenta con servicios sólidos de soporte y mantenimiento.

Desconocimiento de las capacidades de las herramientas de fuentes abiertas. La decisión de usar un producto de fuentes abiertas a veces es una ardua labor para conseguir encontrar aquella herramienta que mejor satisfaga las necesidades de la empresa, y que incluso ofrezca suficientes garantías a nivel de mantenimiento, de documentación, etc.

Debilidades	Principales Argumentos
Cobertura de funcionalidades críticas de los sistemas empotrados	<p>Inmadurez en diversos aspectos del software para la certificación de los productos en aplicaciones críticas</p> <p>Poca fiabilidad y confianza. Es difícil encontrar SW fiable y testeado</p> <p>No cobertura de todas las necesidades de la empresa con soluciones Software de Fuentes Abiertas</p>
Documentación, soporte y mantenimiento	<p>Problemas de mantenimiento y soporte, sobre todo debido a la inestabilidad con determinadas versiones</p> <p>Falta de documentación (escasa o inexistente en algunos casos)</p> <p>Falta de personal formado adecuadamente</p>
Desconocimiento de las capacidades de las herramientas de fuentes abiertas	<p>Desconocimiento de las posibilidades que ofrecen las Software de Fuentes Abiertas</p> <p>La decisión entre usar productos Software de Fuentes Abiertas y productos de primera línea licenciados no es trivial.</p>
Desconocimiento de aspectos legales y alcance de licencias	<p>Complejidad en la verificación de los aspectos legales</p>
Fragmentación	<p>Se aprecia un uso cada vez mas extendido de soluciones basadas en fuentes abiertas, realizado de forma aislada, sin la cobertura de comunidades.</p>

Tabla 5.1: Debilidades del Software de Fuentes Abiertas en las empresas de sistemas empotrados.

Desconocimiento de aspectos legales y alcance de licencias. Los diferentes modelos de licenciamiento de los productos de fuentes abiertas, pueden implicar aspectos legales difíciles de interpretar y gestionar por empresas cuyo negocio no reside en el desarrollo software sino hardware, y con una herencia conservadora fundamentada en modelos de patentado y modelos de utilidad. En el sector de los sistemas empotrados abundan los recelos a licencias como GPL, basadas en el hecho de que esta licencia tiene un carácter "vírico" que puede dar lugar a futuras demandas al integrar dicho código en un sistema empotrado cerrado.

Esta limitación tiene dos aspectos. En principio, existe un desconocimiento técnico del alcance y capacidad de las herramientas Software de Fuentes Abiertas. Por otro lado, existe también una barrera ligada a la fragmentación. La fragmentación en España es extrema con múltiples pymes, pero escasez de grandes comunidades con extensiones internacionales.

La Tabla 5.1 resume los argumentos más importantes con las debilidades mencionadas en este apartado.

5.2

>> Amenazas

Las amenazas, dentro del análisis externo, pretenden determinar todos aquellos factores externos que pueden afectar de manera muy negativa a la empresa, pudiendo requerir estrategias de actuación determinada.

A través de los datos de la encuesta se han identificado las siguientes amenazas, respecto a los sistemas empotrados y a la utilización de fuentes abiertas:

Riesgo en la fiabilidad de los productos resultantes. De forma general se percibe cierto escepticismo en cuanto a la calidad, robustez y seguridad del Software de Fuentes Abiertas; mientras se considera adecuado para formación y desarrollo de prototipos, su utilización en entornos de producción y producto final se percibe como arriesgado. Esta preocupación es particularmente condicionante en dominios o sectores críticos, donde problemas de fiabilidad en el software, pueden poner en riesgo vidas humanas o suponer daños/pérdidas graves. Pese a los

numerosos casos de éxito conocidos y los esfuerzos en marcha en torno a la mejora de calidad y gestión de las comunidades de desarrollo, este mito constituye una importante barrera.

Problemas de integración. La necesidad de integrar las herramientas Software de Fuentes Abiertas con otro tipo de aplicaciones, incluyendo aplicaciones cerradas, en el ecosistema informático de las empresas, crea algunas barreras a la hora de optar por el Software de Fuentes Abiertas. En cierta manera, esto viene ligado a la limitación natural del software cerrado en la conectividad con formatos y protocolos abiertos.

Barrera cultural en su uso. Aunque es algo que está disminuyendo en los últimos años, existe cierta resistencia por parte de los usuarios finales de las aplicaciones empotradas cuando se trata de Software de Fuentes Abiertas.

Desconocimiento de las implicaciones legales por parte de usuarios. Existe un desconocimiento por parte de clientes y usuarios de las posibilidades legales y alcances que proporcionan las licencias de fuente abierta en el desarrollo y comercialización.

La Tabla 5.2 lista los argumentos más destacados relacionados a las amenazas mencionadas en este apartado.

Amenazas	Principales Argumentos
Riesgo en la fiabilidad de los productos resultantes	Aumento del tiempo de desarrollo debido a problemas no detectados o no soportados Mantenimiento lento y fiabilidad baja Soluciones ad-hoc
Problemas de integración	Redistribución, modificaciones propietarias que deben hacerse abiertas Problemas para la integración con aplicaciones legacy
Barrera cultural en su uso	Reticencias por parte del cliente
Desconocimiento de los beneficios del Software de Fuentes Abiertas	Desconocimiento de las posibilidades que ofrece el uso de fuentes abiertas Posibles reticencias por parte de determinados clientes por su desconocimiento del Software de Fuentes Abiertas y sus implicaciones
Desconocimiento de las implicaciones legales por parte de usuarios	Desconocimiento de los aspectos legales y el alcance de las licencias por parte de las empresas

Tabla 5.2: Amenazas del Software de Fuentes Abiertas en las empresas de sistemas empotrados.

5.3

>>Fortalezas

Las fortalezas dentro de la empresa, encuadradas dentro del análisis interno, pretenden determinar todos aquellos factores y elementos positivos dentro del proceso de la empresa que permiten convertirse en una ventaja competitiva. En referencia a los sistemas empotrados, sería todo aquel elemento que permita una ventaja productiva, estratégica de mercado, habilidades, sinergias, etc.

A partir de los datos de encuesta se han identificado las siguientes fortalezas, respecto a los sistemas empotrados y a la utilización de fuentes abiertas:

Buena cobertura de funcionalidades para recursos limitados.

Actualmente se cuenta con Software de Fuentes Abiertas maduro para aplicaciones con fuertes restricciones en la capacidad de los recursos hardware. En particular, existen sistemas operativos en versiones reducidas con características orientadas a los sistemas empotrados y que al mismo tiempo son suficientemente potentes para realizar tareas tales como comunicaciones por redes de datos, soporte gráfico, concurrencia con lanzamiento de threads, etc.

Dominio en el área de diseño de sistemas (HW/FW/SW).

En España existen capacidades para ser referente en el sector, principalmente en ámbitos académicos, universidades y centros, pero también en pymes orientadas a I+D. En la mayoría de casos, las soluciones comerciales existentes no proporcionan capacidad de producción y comercialización competitiva.

Cultura de colaboración. El Software de Fuentes Abiertas promueve la germinación y la formación de comunidades abiertas de colaboración alrededor de proyectos de desarrollo y mantenimiento de las aplicaciones y herramientas de Software de Fuentes Abiertas. Esta gran potencialidad del Software de Fuentes Abiertas, junto al acceso libre a la documentación, es especialmente determinante a la hora de optar por uno u otro producto en el marco de las empresas españolas.

Incremento de capacidades en pymes. Los recursos en las pymes son muy limitados. Las fuentes abiertas suponen una muy buena oportunidad de contribuir y a la vez recibir contribuciones, con lo que se incrementa la capacidad productiva y de resultados de su propio producto. Esto desemboca en una mejora de competitividad gracias al efecto de sinergia implícito en las estrategias basadas en fuentes abiertas.

Apertura operacional. El incremento en la variedad del Software de Fuentes Abiertas abre paso a una mayor capacidad de interoperabilidad y de poder de decisión para adoptar este tipo de software en las distintas fases del proceso de desarrollo y mantenimiento de sistemas empotrados.

Apertura institucional. El desarrollo basado en fuentes abiertas permite romper en las empresas una barrera importante, especialmente en lo que se refiere al intercambio de información y resultados. Asimismo, constituye un marco idóneo para introducir nuevas ideas, diseños y herramientas, mientras disminuye el riesgo de adopción.

Costes reducidos. La reducción en el coste por la utilización de software de fuente abierta se refleja no solamente en el costo efectivo de los productos utilizados, sino también por aspectos indirectos como los costos en la formación, la reducción de los tiempos de desarrollo debido a la gestión simplificada de acceso al Software de Fuentes Abiertas, o la independencia de proveedores únicos.

Implicación de grandes compañías en comunidades. Las grandes empresas poseen una alta capacidad de ejecución, y son un importante aliado en el desarrollo de la aplicación por el relevante feedback que se puede proporcionar. Por otro lado, cada día, grandes compañías están más interesadas e implicadas en el trabajo que se está desarrollando dentro de las comunidades de fuentes abiertas, en la búsqueda de nuevas oportunidades de negocio y apertura de posibilidades y de mercado con la integración con sus propios productos.

Diseños de dispositivos cada vez más complejos. Cada día, el mercado demanda dispositivos electrónicos más complejos. Al igual que en los grandes proyectos del mundo software, el diseño de estos dispositivos sería inabordable sin la reutilización de código. Esto ha hecho aparecer multitud de compañías que ofrecen sus servicios de diseño y venden su código a otras para que éstas las integren en sus productos.

La Tabla 5.3 resume los argumentos más importantes relacionados a las fortalezas mencionadas en este apartado.



Las fortalezas dentro de la empresa pretenden determinar todos aquellos factores y elementos positivos dentro del proceso de la empresa que permiten convertirse en una ventaja competitiva.

Fortalezas	Principales Argumentos
Buena cobertura de funcionalidades para recursos limitados	S.O. reducido y exigencia de software menores lo que hacen que sean idóneos en sistema empotrados. Soluciones más robustas. Satisfacción del cliente cuando era requisito suyo.
Dominio en el área de diseño de sistemas (HW /FW /SW)	Dominio de las tecnologías y métodos en el ámbito de los sistemas empotrados y diseño electrónico
Cultura de colaboración	Posibilidad de compartir código con otros grupos de desarrolladores Amplia documentación (en algunos proyectos Software de Fuentes Abiertas)
Incremento de capacidades en pequeña empresa	Incremento de la capacidad productiva y de resultados mediante contribuciones de comunidades
Apertura operacional	Dominio de la tecnología y del código fuente a todos los niveles Mayor disponibilidad de herramientas al poder tener acceso a un amplio abanico sin coste No depender de un proveedor único
Apertura institucional	Mejora en la gestión y coordinación de proyectos tanto científica como financiera Baja la barrera de entrada en el proceso de I+D
Costes reducidos	Reducción de costes Riesgos de adopción menores y tiempos de desarrollos más cortos Formación Mayor disponibilidad de herramientas Alternativas múltiples
Implicación de grandes compañías en comunidades	Alternativas múltiples

Tabla 5.3: Fortalezas del Software de Fuentes Abiertas en las empresas de sistemas empotrados.

5.4

>>Oportunidades

Las oportunidades son todas las situaciones o factores externos que pueden ser aprovechados por la empresa para un posicionamiento estratégico dentro de la competitividad del mercado.

A partir del análisis de los datos de la encuesta se han identificado las siguientes oportunidades, respecto a los sistemas empotrados y la utilización de fuentes abiertas:

Grandes perspectivas con soluciones maduras basadas en Software de Fuentes Abiertas. La mayor tendencia en el uso de aplicaciones y herramientas Software de Fuentes Abiertas se observa en comunidades maduras y establecidas como GCC o Eclipse. El nivel de implicación de compañías grandes, como IBM, Oracle, y HP entre otros, dentro de estos proyectos de Software de Fuentes Abiertas es un elemento determinante en el crecimiento de estas comunidades. Del mismo modo, los sistemas operativos de fuente abierta para sistemas empotrados tienen ya una sólida base en el mercado español.

Acceso a mercados. Existen aún muchos nichos de mercado en los sistemas empotrados, especialmente en el dominio de aplicaciones críticas que requieren un alto grado de fiabilidad y respuestas de tiempo real estricto. Aunque este dominio es muy dependiente de la información privativa de las empresas, los proyectos I+D representan una oportunidad inmejorable para crear comunidades especializadas en esta área y en la certificación de Software de Fuentes Abiertas.

Herramientas de Soporte. Constitución de comunidades de fuentes

abiertas, definidas con el objetivo de la provisión de métodos y herramientas de soporte durante el ciclo de vida de desarrollo e integración de los sistemas empotrados.

Desde el punto de vista del usuario, la principal oportunidad radica en:

Expansión del abanico de opciones. La existencia de diferentes soluciones de fuente abierta para cubrir las variadas funcionalidades y dominios requeridos por las empresas integradoras y usuarias en última instancia de los sistemas empotrados, evita la dependencia de proveedores y evaluación de nuevos proveedores.

Adopción incremental en las empresas. La adopción de Software de Fuentes Abiertas debe ser resultado de un proceso incremental y progresivo que tome en cuenta principalmente la interoperabilidad, madurez de la comunidad Software de Fuentes Abiertas asociada, y el beneficio efectivo en términos de tiempo de desarrollo. El proceso debe ser incremental tanto en las fases del desarrollo de los sistemas empotrados, como en las áreas aplicativas de la empresa.

La Tabla 5.4 resume los argumentos más importantes relacionados a las oportunidades mencionadas en este apartado.



Las oportunidades son todas las situaciones o factores externos que pueden ser aprovechados por la empresa para un posicionamiento estratégico dentro de la competitividad del mercado.

Oportunidades	Principales Argumentos
Altas Perspectivas con soluciones maduras basadas en Software de Fuentes Abiertas	La tendencia es al alza pero sobre todo en aquellas aplicaciones no críticas o en procesos muy maduros.
Acceso a mercados	Participación conjunta de proveedores tecnológicos y empresas provenientes de sectores cerrados en iniciativas basadas en fuente abierta.
Soporte	Soluciones (herramientas) cerradas fuera del alcance de PyMEs.
Expansión del abanico de opciones	Incrementar la selectividad en las adopciones. Se utilizará pero siempre para proyectos concretos donde sea la mejor opción técnica y económica.
Adopción incremental en las empresas	Incremento progresivo, pero con un ratio de incremento bajo. Adopción dentro de I+D de manera incremental. Si tiene esperanza que creciente en la medida que la madurez y la cuota de mercado del Software de Fuentes Abiertas también crece. Producción de servicios dirigidos a plataformas y sistemas operativos libres. Cada vez una mayor tendencia al uso de Software de Fuentes Abiertas dentro de todas las fases de un proyecto y departamento de la empresa.

Tabla 5.4: Oportunidades y perspectivas del Software de Fuentes Abiertas en las empresas de sistemas empotrados.

06

CONCLUSIONES Y RECOMENDACIONES

En este apartado se resumen las conclusiones más importantes del estudio de la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados.

06

CONCLUSIONES Y RECOMENDACIONES

6.1

>>Conclusiones

En este apartado se resumen las conclusiones más importantes del estudio de la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados.

- Los mayores usuarios de Software de Fuentes Abiertas en las empresas nacionales son las pymes y micro-empresas. Esto se debe en gran parte a la necesidad de internacionalización y a la vocación de innovación de este tipo de empresas. De igual modo, su implicación en proyectos I+D y la colaboración estrecha con universidades y centros de investigación son elementos relevantes a la hora de optar por soluciones Software de Fuentes Abiertas.
- El contexto abierto y creativo de la I+D en España y Europa contribuye de manera importante en la difusión y utilización de software de

fuentes abiertas. Existe un impulso significativo del software de fuentes abiertas por parte de los proyectos y unidades I+D de las empresas. Este aspecto, unido a la creciente importancia que se da a la I+D dentro de las empresas españolas, crea un ambiente propicio para el fortalecimiento de las comunidades de software de fuente abierta.

- Existen nichos aplicativos importantes del software de fuentes abiertas para sistemas empotrados. Las aplicaciones con exigencias críticas de fiabilidad representan un mercado difícil para el software de fuentes abiertas, lo cual requiere tareas de I+D en certificación de Software de Fuentes Abiertas y en la definición de estándares de interoperabilidad abiertos.

- La credibilidad del Software de Fuentes Abiertas está fuertemente ligada al grado de implicación de compañías fuertes en sistemas empotrados. En particular, dentro de los sistemas empotrados para aplicaciones empotradas, un aspecto relevante que ha fomentado el uso del Software de Fuentes Abiertas es la entrada de empresas consolidadas, tales como Wind River, Green Hills o MontaVista.
- La experiencia con herramientas de desarrollo de fuente abierta representa un modelo a seguir en otras áreas de interés para el desarrollo de sistemas empotrados. En especial, el aumento en el uso de Eclipse es resultado de su modelo de gestión basado en una fundación sin intereses de lucro y con contribuciones de compañías grandes.
- El hardware abierto es la solución a la mayoría de los problemas asociados a cores propietarios. Cada core abierto contendrá una amplia base de usuario, lo cual asegura un mejor soporte, una mejor documentación y unos mejores ejemplos de implementación desde los que trabajar. El código está disponible, por lo que cualquier desarrollador puede averiguar lo que necesita saber sobre el core. Finalmente, cuando los cores y sus estándares se desarrollen, los cores llegarán a ser más compatibles que los cores propietarios.



El contexto abierto de la I+D en España y Europa contribuye de manera importante en la difusión y utilización del Software de Fuentes Abiertas

6.2

>> Resumen DAFO

Debilidades	Amenazas
Funcionalidades Sistemas Empotrados no cubiertas Soporte y documentación Fragmentación Desconocimiento de posibles beneficios Desconocimiento de aspectos legales y alcance de licencias	Percepción de baja calidad Dificultad de integración con SW cerrado Barrera cultural en la utilización de Software de Fuentes Abiertas Desconocimiento del uso legal en el desarrollo y comercialización
Fortalezas	Oportunidades
Cobertura de funcionalidades para recursos limitados Dominio en el área de diseño de sistemas Cultura de colaboración Apertura operacional e institucional Costes reducidos Aumentar la implicación de grandes compañías en comunidades Incremento de capacidades en pequeña empresa Capacidad de incrementar la complejidad de los dispositivos	Perspectivas fuertes con Software de Fuentes Abiertas maduro Acceso a mercados Herramientas de soporte Expansión del abanico de opciones Adopción incremental en las empresas

Figura 6.1: Resumen DAFO

6.3

>> Recomendaciones y estrategias

La tabla presenta la matriz DAFO como fuente de estrategias y recomendaciones tras el análisis de los datos recogidos en el estudio sobre la situación del Software de Fuentes Abiertas en empresas proveedoras e integradoras de sistemas empotrados.

		Fortalezas						Debilidades					
		Cobertura de funcionalidades para recursos limitados	Domínio en el área de diseño de sistemas	Cultura de colaboración	Apertura operacional	Apertura institucional	Costes reducidos	Aumentar la implicación de grandes compañías en comunidades	Funcionalidades SE no cubiertas	Soporte y documentación	Fragmentación	Desconocimiento de posibles beneficios	Desconocimiento de aspectos legales y alcance de licencias
Oportunidades	Perspectivas fuertes con Software de Fuentes Abiertas maduros			Orange				Orange		Light Green	Orange		
	Acceso a mercados	Light Purple	Light Purple	Light Purple	Light Purple			Light Purple					
	Herramientas de soporte		Light Purple	Light Purple	Light Purple								
	Expansión del abanico de opciones	Orange	Orange				Orange						
	Adopción incremental en las empresas						Orange						
Amenazas	Percepción de baja calidad								Light Green				
	Dificultad de integración con SW cerrado				Light Green								
	Barrera cultural en la utilización de Software de Fuentes Abiertas			Blue		Blue			Light Green				Light Green

Tabla 6.1: Matriz DAFO

Estándares Abiertos

Los aspectos de estandarización en el ámbito de los sistemas empotrados, han sido identificados como clave en su adopción por el mercado y priorizados por la Unión Europea. En este sentido, se potencia el desarrollo y coordinación de estándares a diferentes niveles de sistemas: abordando desde el nivel arquitectónico (buses, definición de interfaces, gestión de IPs), middleware de interconexión o herramientas de soporte, a través de la adopción de formalismos y modelos consensuados.

En el documento de "ARTEMIS Standardisation strategy" se identifican las directrices en esta línea y de forma explícita las implicaciones en el Software de Fuentes Abiertas, diferentes proyectos FP7, como ProSE o ADAMS soportan esta estrategia en diferentes ámbitos y marcan recomendaciones específicas; así mismo, la relevancia en cuanto a regulación e interacción con entidades certificadoras, se ha convertido en requisito de elegibilidad para nuevas iniciativas.

A título de ejemplo, citar las siguientes iniciativas en curso: Middleware: AUTOSAR, OSGi, HIJA (High Integrity Java), RTSJ Java, Ravenscar, Modelos y herramientas: Estándares OMG (UML2, SysML, MARTE), Open ADR.

Los nuevos estándares abiertos propician la generación de nuevas oportunidades de negocio en un marco colaborativo entre iguales, donde soluciones cerradas e implementaciones abiertas parten de iguales condiciones.

Certificación y Capacitación

Es prioritario poder garantizar la calidad del software. Planteamientos como "que el cliente valida" deben ser erradicados para poder afrontar

muchas barreras y desconfianzas que encuentran las empresas de desarrollo a la hora de acceder a nuevos clientes. La fidelización de un cliente debe estar basada en la calidad del trabajo.

Actualmente CENATIC esta promoviendo la certificación de Software de Fuentes Abiertas a través de un estudio de viabilidad técnico-financiera para la certificación de Software de Fuentes Abiertas, donde se aborda la definición de un nuevo marco de certificación específico para las peculiaridades de este tipo de desarrollos.

En esta línea, desde VULCANO y otras comunidades como MORFEO o PEGASO, se ha planteado la constitución de Centros de Competencia que permitan la cualificación, capacitación y acreditación de empresas y personas implicadas en el proceso productivo. La apertura de estos mercados exigentes, requiere esfuerzos adicionales en cuanto a documentación, formalización del modo en que se realizan aportaciones dentro de las comunidades, responsabilidades más definidas. Los centros de competencia deberían coordinar y ser referencia en estas prácticas.

Otro aspecto a tratar en este ámbito es la formación a los diferentes agentes implicados, proveedores, integradores y usuarios de tecnología abierta acerca de los diferentes modelos de licenciamiento: modalidades, implicaciones de uso/transferencia. Pese a sus implicaciones legales, no existe formación adecuada a la hora de sopesar las diferentes opciones; deben ser equipos mixtos compuestos por responsables de producto y equipo de desarrollo.

Diseminación del planteamiento y soluciones basadas en fuentes abiertas

El ámbito de ingeniería y de los sistemas empotrados en particular,

resulta intrínsecamente cerrado, ya que tradicionalmente la explotación de los productos viene acompañada en la mayoría de casos de patentes reguladoras. Para lograr evitar estas barreras es crucial comunicar los beneficios de las soluciones de fuente abierta: reducción de costes, tiempo de puesta en mercado, capacidades y el concepto de comunidad, etc.

La celebración de eventos de diseminación a diferentes niveles (regional, nacional e internacional) ayuda en esta línea.

Comunidades fuertes

Uno de los factores valorado por los usuarios a la hora de implantar una tecnología es su madurez: calidad y soporte; el hecho de la existencia de comunidades por detrás de los productos, que garanticen su mantenimiento y mejora constituye un requisito primordial.

Por otra parte, la penetración en nuevos mercados es complicada para proveedores aislados; si se logra la suficiente coordinación en la cadena de valor, existe una gran oportunidad de romper esta fragmentación evitando una multiplicidad de soluciones "a medida" y agrupar capacidades para generar y comercializar soluciones competitivas. Si no se consigue, existe un riesgo importante de que el sector deje de ser competitivo y pierda considerablemente cuota de mercado.

En este escenario, es necesaria la participación de los diferentes agentes: grandes empresas, pymes y entidades usuarias, beneficiándose y aportando cada uno, según sus intereses y capacidades. En general, universidades y centros de investigación utilizando este modelo como forma de canalizar y lograr visibilidad de sus resultados de investigación, la pequeña/mediana empresa integrando y madurando estas soluciones

y; desde la perspectiva de las grandes empresas traccionando hacia mercados estratégicos, incorporando a coste/riesgo razonable mejoras en sus desarrollos y aumentando la competencia entre proveedores.

Proyectos estratégicos

Se ha visto cómo los departamentos de I+D de las empresas, constituyen el nexo entre la industria y el desarrollo basado en fuentes abiertas. A fin de generar, canalizar y transferir tecnología basada en fuentes abiertas y coordinar los esfuerzos para desarrollar la capacidad tecnológica en función de las necesidades industriales, es necesaria la promoción de proyectos estratégicos I+D que integren la cadena de valor completa. Estos proyectos deben ser el germen de nuevas comunidades innovadoras en tecnología.

Francia constituye una referencia en este sentido, donde las plataformas tecnológicas o "pôles de compétitivité", están coordinando nuevos proyectos de investigación inter-sectoriales centrados en provisión de herramientas de soporte a la ingeniería y nuevos diseños abiertos fundamentados en los principios de fuentes abiertas (p.e. Topcased, EDONA, Papyrus, etc).

6.3.1 >>Consideraciones estratégicas respecto a sistemas críticos empotrados

Los sistemas críticos empotrados se pueden considerar como un grupo específico dentro de lo que es el área de los sistemas empotrados ya que tienen unas particularidades especiales que les enmarcan dentro de unas necesidades que deben ser cubiertas, sobre todo en lo que se refiere a certificaciones. Es por ello que la inclusión o no de Software de

Fuentes Abiertas debe ser considerado bajo estos términos para cumplir con las necesidades críticas del dominio. En este apartado se pretende dar una visión general estratégica sobre este dominio particular.

Desde el punto de vista de la integración de componentes off-the-shell, existen una serie de riesgos asociados, siendo este punto independiente del hecho de que el software sea propietario o de fuentes abiertas. Entre los riesgos más comunes desde el punto de vista de la integración de componentes "off-the-shelf" se encuentran los siguientes:

- Saltos entre versiones de componentes, congeladas para una determinada release de producto y nuevas versiones.
- Desaparición de las compañías que comercializan u ofrecen el soporte.
- Riesgos técnicos asociados a fallos del producto.
- Riesgos añadidos asociados a términos de licenciamiento.

Estos riesgos pueden ser mitigados con la posibilidad de disponer el acceso al código fuente, como es el caso de Software de Fuentes Abiertas, frente a opciones cerradas. En este sentido el integrador de sistemas puede disponer de un control del riesgo efectivo, tras la identificación de los mismos, así como efectuar las correcciones necesarias, eliminando la dependencia sobre el manejo del código en terceros.

Desde la perspectiva técnica, y teniendo en cuenta la importancia de las certificaciones en este tipo de sistemas, el principal reto es la generación de "safety cases" asociados al sistema completo. Este tipo de "safety cases" deben ser aceptados por la autoridad de certificación

competente en el sector. En productos comerciales, también conocidos como COTS, el proveedor provee de la documentación correspondiente a su producto. A nivel de sistema, ésta es tarea del integrador. Para componentes basados en fuentes abiertas se plantea como solución un consorcio de integradores con interés en compartir esfuerzos y recursos en el proceso de certificación. Dentro de los muchos aspectos considerados en los procesos de certificación de producto, uno de los aspectos clave lo constituye el denominado "robustness testing". El principal objetivo de este test es el chequear el comportamiento del software ante fallos a diferentes niveles y condiciones de operación de estrés. Entre los principales objetivos de este test se encuentran:

- Comparativas de calidad entre productos candidatos,
 - Las pruebas de rendimiento constituyen un criterio de decisión en la selección de los productos candidatos.
- Caracterización de los modelos de fallo de determinados componentes
 - Condicionantes en el re-diseño de cara a la integración del sistema completo.

Dependiendo del nivel de criticidad, técnicas de verificación más formales son requeridas. Esto es debido a que la certificación tiene como objetivo último el asegurar el funcionamiento completo del sistema. Como ejemplo tenemos el caso del análisis estático, que facilita las pruebas de fuente o model-checking, que soportado por modelos, verifica el cumplimiento de propiedades del producto.

Se puede observar que en este tipo de sistemas críticos la certificación

es uno de los puntos claves. Es por ello que es necesario abordar de manera coordinada temas de certificación de producto, y no ya solamente de procesos como se plantea en diferentes iniciativas de certificación de Software de Fuentes Abiertas. Esto va a exigir esfuerzos adicionales en cuanto a documentación, formalización del modo en que se realizan aportaciones dentro de las comunidades, responsabilidades más definidas y principalmente, técnicas de testeo avanzado. Es por ello que una coordinación y referente de buenas prácticas debería ser propuesta desde los centros de competencia.

El Software de Fuentes Abiertas constituye una oportunidad estratégica para los integradores de sistemas críticos. Su utilización es viable, integrado con soluciones arquitectónicas apropiadas que simplifiquen la gestión de los aspectos críticos, y acciones para la certificación de sistema bajo dichos niveles de *safety*. Se percibe que el uso de Software de Fuentes Abiertas en ámbitos críticos requiere un planteamiento proactivo, en el que se fijen unas metas plausibles para cubrir las necesidades de este sector.

Siendo considerado un factor estratégico en la competitividad, la compatibilidad del Software de Fuentes Abiertas con normativas y procesos de certificación determina que deben plantearse estrategias globales, tanto en el uso como contribución al Software de Fuentes Abiertas, potenciadas y soportadas por el sector industrial.

A continuación se muestran algunas sugerencias para acercar el planteamiento basado en fuentes abiertas a los sectores críticos:

- **Definición de un escenario de certificación común.** Es necesario potenciar proyectos pilotos como demostradores, involucrando partners de diferentes sectores y visiones de la cadena de valor de

los grandes sistemas críticos (integradores, proveedores, entidades certificadoras, etc).

- **Difusión y capacitación en los principios arquitectónicos de los sistemas críticos.** Las arquitecturas hardware que soportan este tipo de sistemas tienen implicaciones específicas que deben tener en cuenta los proveedores de software: mecanismos de redundancia, particionado espacial y temporal, protocolos, etc. Existen diversas soluciones en el mercado, como es el caso de Integrated Modular Avionics (IMA), TTA/TTP que en su mayoría son abiertas, pero ni cuyas capacidades ni facilidades de protección son conocidas de cara a la integración con Software de Fuentes Abiertas.
- **Promoción de entornos de desarrollo basados en software de fuentes abiertas.** Este es un objetivo que se puede alcanzar a corto plazo con el respaldo de casos de éxito concretos en el uso del Software de Fuentes Abiertas en entornos de desarrollo de sistemas críticos. Los principales beneficios empresariales se enumeran a continuación:
 - Independencia de proveedores.
 - Facilidad de integración de avances tecnológicos desde los ámbitos académicos y de investigación.

Para las empresas europeas punteras, el software de fuentes abiertas representa una oportunidad para captar los mejores talentos en innovación y desarrollo de herramientas, y capacitación en los aspectos críticos de estos sectores. Desde el punto de vista de los investigadores, la especialización en Software de Fuentes Abiertas representa la

oportunidad de introducirse en áreas de negocio que de otra forma estarían herméticamente cerradas salvo para los empleados de determinadas corporaciones.

Todas las recomendaciones anteriormente presentadas necesitan el impulso de la industria, a la vez que un soporte político y financiero por parte de las organizaciones públicas.



Para las empresas europeas punteras, el Software de Fuentes Abiertas representa una oportunidad para captar los mejores talentos en innovación y desarrollo de herramientas, y capacitación en los aspectos críticos de estos sectores.

Desde el punto de vista de los investigadores, la especialización en Software de Fuentes Abiertas representa la oportunidad de introducirse en áreas de negocio que de otra forma estarían herméticamente cerradas salvo para los empleados de determinadas corporaciones.



07

EQUIPO REALIZADOR Y PANEL DE
EXPERTOS

07

EQUIPO REALIZADOR
Y PANEL DE EXPERTOS

7.1

>>Equipo realizador

7.1.1

>>CENATIC

- **Pop Ramsamy**
Dirección del proyecto
- **Ana Trejo**
Coordinación de la Investigación

7.1.2

>>Fundación European Software Institute (Tecnalia)

- **Carmen Alonso**
- **Naia Arana**
- **Sergio Campos**
- **Huascar Espinoza**
- **Joseba Laka**

7.1.3

>>Asociación de Investigación y Cooperación Industrial de Andalucía (AICIA)

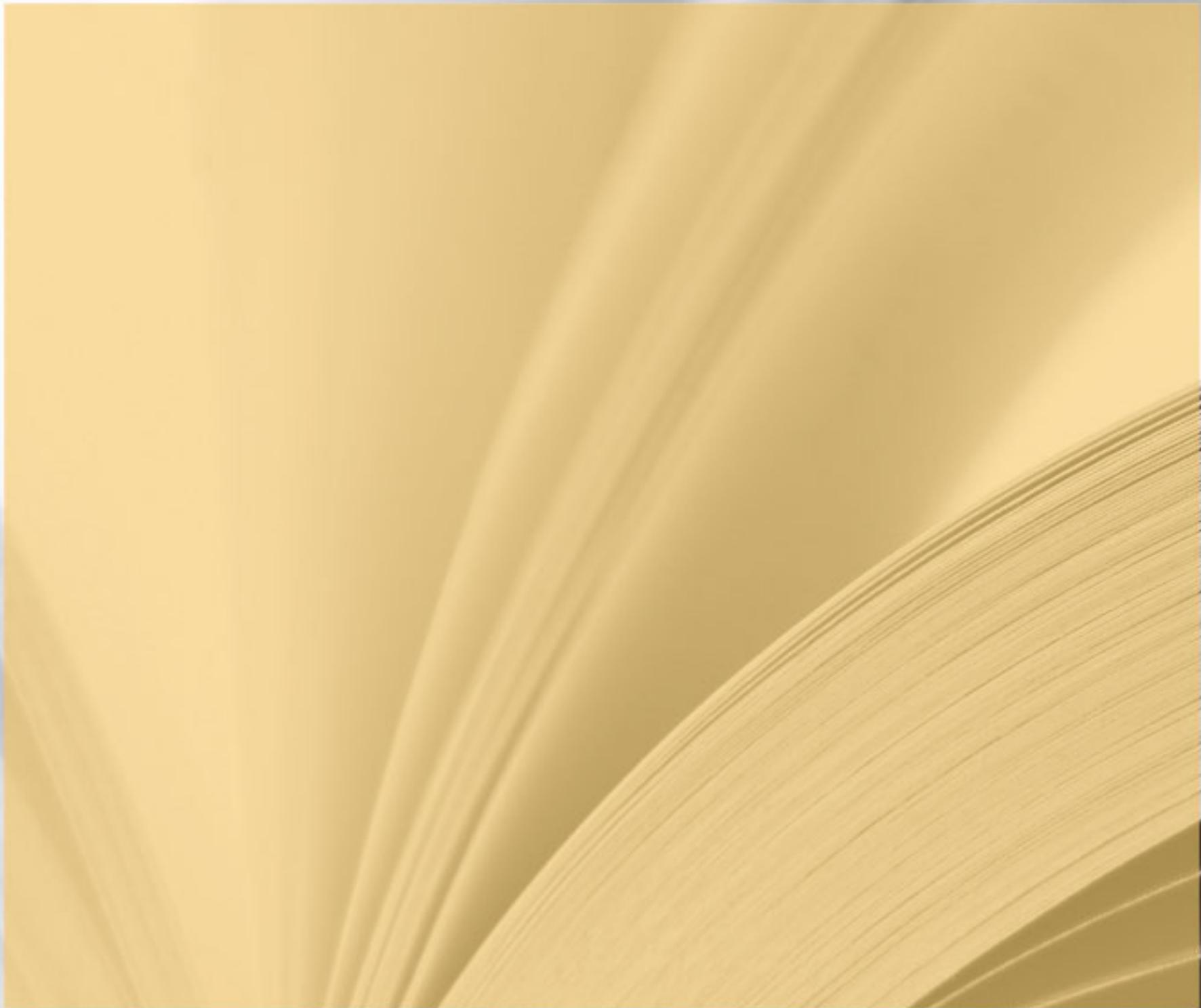
- **Rosa Rubio**
- **Antonio Torralba**

7.2

>>Panel de expertos

Agradecemos especialmente la colaboración y el esfuerzo de los siguientes expertos que han participado en diferentes partes del estudio a título personal, aportando su inestimable experiencia y conocimiento en el ámbito del Software de Fuentes Abiertas y Sistemas Empotrados:

- **José María Casanova**
Universidade da Coruña
- **Kirsten Haaland**
UNU-MERIT
- **Joaquín Lopez**
Aiddea Linux
- **Marcos Martínez**
DS2
- **Andrés Leonardo Martínez Ortiz**
Telefónica I+D
- **Carlos Pardo**
SIDSA
- **Alberto Ruiz de Olano**
Ikerlan
- **Juan José Sanchez**
IGALIA



08

BIBLIOGRAFÍA Y REFERENCIAS

08

BIBLIOGRAFÍA Y REFERENCIAS

[ADAMS] Action for the Dissemination and Adoption of the MARTE and related Standards for component based middleware.

Enlace web: <http://www.adams-project.org/>

[ARTEMIS INT364] Iniciativa Tecnológica sobre sistemas de computación empotrados / Creación de una empresa común. ARTEMIS 2007

[ARTEMIS SRA2006] ARTEMIS Strategic Research Agenda, first edition March 2006

[ATENEA] Métodos y herramientas V3. Proyecto ATENEA Arquitectura, Middleware y Herramientas. 2009

[AUTOSAR] AUTomotive Open System ARchitecture.

Enlace web: <http://www.autosar.org/>

[Bailey 2005] Embedded System: Desktop Integration. Wordware Publishing, Inc 2005. Oliver H. Bailey

[busybox] GNU/Linux empotrado. Implementación de las utilidades más comunes estándares de Linux.

Enlace web: <http://linuxemb.wikidot.com/busybox>

[CentOS] The Comunity Enterprise Operating System.

Enlace web: <http://www.centos.org/>

[Ciria Automoción] Jaime Ciria. UPC.

Enlace web:

http://docencia.ac.upc.edu/EPSC/PSE/documentos/Observatorio/07-08-Q2/SL_automocion.ppt, Jaime Ciria

[CMT2009] Estadísticas del sector, Segundo trimestre. Comisión del Mercado de Telecomunicaciones.

Enlace web: <http://www.cmt.es>

[eCOS 2009] An open source, royalty-free, real-time operating system intended for embedded applications.

Enlace web: <http://ecos.sourceware.org/>

[EDONA] The EDONA Project: Environments of Development Open to the Standards of the Car.

Enlace web:

<http://www.edona.fr/scripts/home/publigen/content/templates/show.asp?P=111&L=EN>

[ESCET] Grupo de Diseño HwSw URJC

<http://www.escet.urjc.es/jmartine>

[FreeRTOS] The FreeRTOS Project.

Enlace web: <http://www.freertos.org/>

[Gerzo 2006] Introduction to NanoBSD. Daniel Gerzo 2006.

Enlace web:

<http://www.freebsd.org/doc/en/articles/nanobsd/index.html>.

[GNOME MOB] The GNOME Mobile Platform.

Enlace web: <http://www.gnome.org/mobile/>

[GStreamer] A library for constructing graphs of media-handling components.

Enlace web: <http://gstreamer.freedesktop.org/>

[GTK] The GTK+ Project, a toolkit for creating graphical user interfaces.

Enlace web: www.gtk.org

[Heath 1997] Embedded System Design, Newnes 1997, Steve Heath

[HIJA] High-Integrity Java Project.

Enlace web: <http://www.hija.info/joomla/>

[INES 2009] Tecnologías software orientadas a servicios. Plataforma INES, Círculo de Innovación en TIC (CITIC) 2009.

[INFCEPAT2008] Software de Fuentes Abiertas para el desarrollo de la Administración Pública Española. CENATIC 2008.

[LiMo] The LiMo Foundation.

Enlace web: <http://www.limofoundation.org/>

[Lucero 2006] Linux en Sistemas empotrados. Alejandro Lucero.

Enlace web: http://www.os3sl.com/Documents/Seminario_UAM_II.pdf

[lwIP] A light-weight implementation of the TCP/IP protocol suite.

Swedish Institute of Computer Science. Adam Dunkels.

Enlace web: <http://www.sics.se/adam/lwip/>

[Lynch 2009] Tiny Core Linux 2.1. Desktop Linux Reviews. Jim Lynch 2009.

Enlace web: <http://desktoplinuxreviews.com/2009/07/01/tiny-core-linux-2-1>

[MAEMO] Maemo Community.

Enlace web: <http://maemo.org>

[MICROBOTICA] Microbótica.

Enlace web: <http://www.microbotica.com>

[MOBLIN] Moblin, a Linux Foundation Project.

Enlace web: <http://moblin.org/>

[MORFEO] Comunidad de Software Libre Morfeo.

Enlace web: <http://www.morfeo-project.org/>

[Nass 2008] An insider's view of the 2008 Embedded Market Study, Embedded.com The Official Site of the Embedded Development Community, Richard Nass

[Nokia QT] A cross-platform application and UI framework. Nokia.

Enlace web: <http://qt.nokia.com/products>

[OHA ANDROID] Android. The Open Handset Alliance.

Enlace web: <http://www.openhandsetalliance.com/>

[OMG CORBA] Common Object Request Broker Architecture. OMG.

Enlace web: <http://www.corba.org>

[OMG MARTE] Modeling and Analysis of Real-time and Embedded systems.

Enlace web: <http://www.omgmarte.org/>

[OMG SysML] The OMG systems Modeling Language.

Enlace web: <http://www.omgsysml.org/>

[OMG UML] The Unified Modeling Language.

Enlace web: <http://www.uml.org/>

[OpenBios] <http://www.openfirmware.info>

[OpenBSD] A free version of Unix. OSData.

Enlace web: <http://www.osdata.com/oses/openbsd.htm>

[OpenADR] The Open Automation of Demand Response.

Enlace web: <http://www.openadrcollaborative.org/>

[OpenSoC] OpenSoC,

Enlace web: <http://www.opensocdesign.com>

[OSADL] Open Source Automation Development Lab.

Enlace web: <http://www.osadl.org/>

[OSGI] Open Services Gateway Initiative

Enlace web: <http://www.osgi.org/Main/HomePage>

[PAPYRUS] Open Source Tool for Graphical UML2 Modelling.

Enlace web:

<http://www.papyrusuml.org/scripts/home/publigen/content/templates/show.asp?P=55&L=EN&ITEMID=2>

[PEGASO] Proyecto PEGASO, una Factoría Multidisciplinar de Conocimiento Libre. Enlace web:

<http://factoriapegaso.navegas.com/?section=proyecto&page=proyecto>

[PROSE] Promoting Standardisation for Embedded Systems.

Enlace web: <http://www.prose-project.eu/dotnetnuke/>

[Ravenscar] Perfil Ravenscar de ADA.

Enlace web: <http://www.ada-auth.org/cgi-bin/cvsweb.cgi/Als/Al-00249.TXT?rev=1.14>

[Ripoll RTLinux] Introducción a RTLinux. UPV. Ismael Ripoll.

Enlace web: <http://rtportal.upv.es/tutorial/01-intro/01-intro.html>

[RTLinuxFree] OpenRTLinux and Wind River Real-Time Core.

Enlace web: <http://www.rtlinuxfree.com/>

[RTSJ] The Real-Time Specification Java.

Enlace web: <http://www.rtsj.org/>

[Segovia-Abella] Il Libro blanco del software libre. Miguel Angel Segovia Romero, Alberto Abella García.

Enlace web: <http://libroblanco.com/cms/>

[TAO] A CORBA V3.0 compliant C++ ORB. Object Computing, Inc.
Enlace web: www.theaceorb.com/

[TOPCASED] The Open-Source Toolkit for Critical Systems.
Enlace web: <http://www.topcased.org/>

[TRUST4ALL] The Trust for all Project. Trustworthiness in embedded software.
Enlace web:
<http://www.hitech-projects.com/euprojects/trust4all/summary.htm>

[Walls 2006] Embedded Software: The Works, Elsevier 2006, Colin Walls

[XFree86] The XFree68 Project, Inc.
Enlace web: <http://www.xfree86.org/>

[XORG] The X.Org project: X Windows System.
Enlace web: <http://www.x.org/>

[VULCANO] Proyecto VULCANO.
Enlace web: <http://www.ines.org.es/vulcano/>



09

ANEXO. SOLUCIONES ESPECÍFICAS PARA SISTEMAS EMPOTRADOS

En este apartado se presentan a continuación, referencias y descripciones de las diferentes soluciones específicas para sistemas empotrados, que se han detectado durante la realización del informe.

09

ANEXO

En este apartado se presentan a continuación, referencias y descripciones de las diferentes soluciones específicas para sistemas empuotrados, que se han detectado durante la realización del informe.

9.1

>> Productos/Tipo software utilizados en infraestructura y componentes hardware base

9.1.1 >> IPCores

Dentro de una FPGA se puede incluir la funcionalidad de varios circuitos integrados. Esta funcionalidad puede ser desarrollada por el mismo equipo de trabajo o adquirida a través de un tercero. Debido a que estas

funcionalidades son como componentes electrónicos, pero sin su parte física, se los suele llamar componentes virtuales. En la industria se los conoce como bloques de propiedad intelectual o "IP cores".

9.1.1.1 >> Leon3

Leon3 es un procesador de 32 bits basado en la arquitectura SPARC V8 con soporte para configuraciones multiprocesos. El procesador es completamente sintetizable y pueden implementarse hasta 16 núcleos de CPU en configuraciones de multiprocesado asimétrico (AMP) o de multiprocesado síncrono (SMP). Una configuración típica con 4 procesadores es capaz de ofrecer hasta 1600 MIPS de rendimiento. El código fuente completo del núcleo del multiprocesador Leon3 está disponible bajo la licencia GNU/GPL para evaluación, investigación y propósitos educativos. También, existe una licencia de bajo coste para aplicaciones comerciales.

La solución multiprocesadora del Leon3 brinda un mayor rendimiento a bajas frecuencias que soluciones de procesado simple. Esto conlleva

significantes ahorros en coste y potencia, mientras se mantiene la compatibilidad con las herramientas de automatización de diseño electrónico existentes y los flujos de desarrollo. Este procesador simplifica el diseño de sistemas multiprocesadores complejos, por lo tanto se reduce el tiempo de llegada al mercado del producto y los costes de diseño.

9.1.1.2 >> Nios

Nios es un microprocesador RISC de propósito general. El procesador Nios soporta tanto arquitecturas de 32 bits como de 16 bits, ambas utilizan instrucciones de 16 bits.

Las implementaciones de Nios pueden incluir hasta 512 registros de propósito general internos. El compilador utiliza estos registros para acelerar las llamadas a las subrutinas y el acceso a las variables locales.

El conjunto de instrucciones de Nios incluye instrucciones de carga y almacenamiento que el compilador utiliza para acelerar el acceso a la estructura y a la pila.

Además, los usuarios pueden incorporar lógica propia directamente dentro de la unidad aritmético-lógica (ALU) de Nios.

9.1.1.3 >> PicoBlaze

El microcontrolador Picoblaze es un soft core de 8 bits, diseñado para ser empotrado en dispositivos FPGAs como SpartanTM-3, VirtexTM-II, y Virtex-IIProTM.

El núcleo de este microcontrolador es completamente empotrado dentro de una tarjeta FPGA y no requiere recursos externos. Es extremadamente sensible y su funcionalidad básica es fácilmente extendida y reforzada

por su comunicación con el exterior a través de sus puertos de entrada/salida.

9.1.2 >>Arquitectura y Plataformas hardware

9.1.2.1 >> Universal Plug&Play (UpnP)

Universal Plug&Play (UPnP) es una arquitectura abierta y distribuida que permite a las aplicaciones de los dispositivos conectados a una red intercambiar información y datos de forma sencilla y transparente para el usuario final. Esta arquitectura es independiente al fabricante, sistema operativo, lenguaje de programación, protocolo, etc.

Los beneficios de usar esta tecnología son:

- Independencia de medios y dispositivos
- Independencia de plataforma
- Tecnologías basadas en Internet
- Control UI
- Control programático
- Protocolos básicos comunes
- Extensible

9.1.2.2>> Plataformas

ASoC (Alsa System on a Chip). El subsistema ASoC proporciona un framework para audio en sistemas empotrados que corren en Linux.

Arduino. Arduino es una plataforma open-hardware basada en una sencilla placa con entradas y salidas (E/S), analógicas y digitales, y en un entorno de desarrollo que implementa el lenguaje Processing/Wiring.

Su corazón es el chip Atmega8, un chip sencillo y de bajo coste que permite el desarrollo de múltiples diseños.

Al ser open-hardware tanto su diseño como su distribución es libre. Es decir, puede utilizarse libremente para desarrollar cualquier tipo de proyecto sin tener que adquirir ningún tipo de licencia.

Arduino puede utilizarse en el desarrollo de objetos interactivos autónomos o puede conectarse a un PC a través del puerto serie utilizando lenguajes como Flash, Processing, MaxMSP, etc. Las posibilidades de realizar desarrollos basados en Arduino tienen como límite la imaginación.

Asimismo, su sencillez y su bajo coste, recomiendan su uso como elemento de aprendizaje e iniciación en el mundo de la electrónica digital.

Freeduino. Freeduino comienza como un proyecto colaborativo para publicar los archivos de producción compatibles con Arduino. Los archivos que resultaron de este proyecto permiten a los usuarios crear placas que son 100% compatibles, funcional, eléctrica y físicamente con el hardware Arduino.

Sun SPOT. Sun SPOT es un dispositivo Java empujado y programable con una gran flexibilidad. La unidad básica incluye acelerómetro, sensores de temperatura y de luz, transmisor radio, 8 LEDs multicolor, 2 botones de control, 5 pines de entrada/salida digital, 6 entradas analógicas, 4 salidas digitales, y una batería recargable.

Debido a su implementación Java, la programación del Sun SPOT es sorprendentemente fácil.

9.1.2.3>> Rifidi

Rifidi es una plataforma middleware completa de fuentes abiertas para la construcción de todo lo necesario para una aplicación RFID. Entre otras características, esta plataforma permite un prototipado rápido para una aplicación RFID.

9.1.3 >>Diseño Hardware

- **LART** (Linux Advanced Radio Terminal). LART es un potente y pequeño sistema empujado capaz de correr Linux. Su rendimiento está alrededor de 250 MIPS mientras que tiene un consumo de menos de 1 W de potencia. En una configuración estándar posee 32 MB de DRAM y 4 MB de Flash ROM, lo cual es suficiente para un kernel Linux y una imagen en la RAM considerable.
- **MIT Pengachu.** Dispositivo de mano basado en Linux, cuyo diseño hardware y software (y todos los archivos necesarios para construirlos) están disponibles abiertamente para todo el mundo.
- **RFID Guardian v3.** RFID Guardian es un dispositivo móvil que ofrece una gestión de la privacidad y la seguridad RFID para las personas. Monitoriza y regula el uso de RFID, de la parte del consumidor.

9.1.4 >>Herramientas

9.1.4.1 >> gEDA

El proyecto gEDA ha producido y continúa trabajando en un completo paquete con licencia GPL de herramientas de Automatización de Diseño Electrónico. Estas herramientas se utilizan para el diseño de circuitos eléctricos, captura de esquemáticos, simulación, prototipado, y fabricación.

Actualmente, el proyecto gEDA ofrece un amplio conjunto de aplicaciones software libres para diseño electrónico, incluyendo captura de esquemáticos, gestión de atributos, generación de la BOM (Bill of Materials), generación de la netlist en 20 formatos distintos, simulación analógica y digital, y layout PCB (Printed Circuit Board).

Este proyecto comenzó debido a la falta de herramientas de automatización de diseño electrónico libres para sistemas POSIX (Portable Operating System Interface) con el propósito principal de avanzar en el estado del hardware libre o el hardware de fuentes abiertas. El paquete se está desarrollando principalmente en la plataforma GNU/Linux, aunque hay esfuerzo en hacer que pueda correr en otras plataformas también.

9.1.4.2 >> ISE Web Pack (Xilinx)

La herramienta de diseño ISE (Integrated Software Enviroment) WebPACK es una completa solución libre para el diseño de FPGA de principio a fin. Está disponible para diferentes sistemas operativos.

ISE WePACK es la solución descargable ideal para el diseño de FPGA y CPLD que ofrece simulación y síntesis HDL, implementación, prueba de dispositivos, y programación JTAG.

ISE WebPACK ofrece un flujo de diseño completo que proporciona un acceso rápido a las características y a la funcionalidad del entorno de desarrollo sin coste alguno. Xilinx ha creado una solución que permite implementación eficaz, proporcionando una solución de diseño que es constantemente actualizada con una descarga libre de errores y un simple fichero de instalación.

9.1.4.3 >> SPICECAD

SPICECAD es un programa para la introducción interactiva de gráficos y la simulación de circuitos electrónicos.

SPICECAD contiene una interfaz para los simuladores SPICE3 y HSPICE, fácilmente intercambiables. Soporta todos los dispositivos y todos los tipos importantes de análisis. Además, permite técnicas de diseño jerarquizado con un número arbitrario de niveles jerárquicos. Se pueden incorporar las *netlists* creadas con otras herramientas. El usuario puede crear sus propios símbolos para los subcircuitos.

Además, existen otras herramientas open source para el diseño hardware que describimos brevemente a continuación:

- **Magic 7.** Herramienta de layout ASIC para diseño de chips.
- **KiCAD.** Software para el diseño de circuitos e impresión de placas.
- **POD.** Periféricos bajo demanda para diseño de FPGA.
- **SimCAS.** Simulador analógico de señal.
- **SVEditor.** Plugin de Eclipse para diseño con VHDL.
- **myHDL.** Editor hw con Python.
- **Qucs.** Simulador de circuitos.

9.1.4.4 >> SystemC

SystemC es un lenguaje de descripción de hardware y de sistemas formado por un conjunto de librerías y macros en C++ que hacen posible una simulación de procesos concurrentes. La sintaxis utilizadas es C++. Resulta muy útil para el modelado de sistemas a nivel de comportamiento. La metodología de diseño es comenzar con un modelo

de alto nivel escrito en C++ y aplicar un proceso iterativo consistente en transformar el código para usar sólo los elementos que tengan su equivalente en un lenguaje de descripción de hardware.

9.2

>> Productos/Tipo software utilizados en infraestructura y herramientas software base

9.2.1

>> Sistemas operativos

9.2.1.1 >> Linux

Linux, también conocido como GNU/Linux, es uno de los ejemplos más prominentes del Software de Fuentes Abiertas y del desarrollo del código abierto, cuyo código fuente está disponible públicamente, para que se pueda de manera libre usar, estudiar, redistribuir, comercializar y, modificar.

Linux puede instalarse en ordenadores de escritorio (PCs x86 y x86-64 así como Macintosh y PowerPC), ordenadores de bolsillo, teléfonos móviles, portátiles, dispositivos emporados, videoconsolas (Xbox, PlayStation 3, PlayStation Portable, Dreamcast, GP2X...) y otros, sin embargo su mayor desarrollo se ha llevado a cabo en los servidores y supercomputadores.

Las variantes de los sistemas Linux se denominan distribuciones y su objetivo es ofrecer una edición que cumpla con las necesidades de determinado grupo de usuarios. Dentro de las distribuciones, las hay que contienen software que no es libre, y otras que son de pago. Ejemplos de distribuciones de Linux son: Fedora, Debian, Ubuntu y RedHat.

RedHat Enterprise Linux

Se trata de una compañía encargada de la creación y mantenimiento de la distribución del sistema operativo Linux que lleva su mismo nombre y de Fedora. Es una de las distribuciones más populares de Linux. Además de estos, se encargan de la comercialización de diversos productos y servicios basados en software de código abierto. Algunas de las contribuciones más destacadas es por ejemplo el sistema de empaquetación RPM, utilidades de administración y configuración de equipos (sndconfig, mouseconfig), etc.

Fedora

Es una distribución Linux para propósitos generales basada en RPM.

Proyecto Debian

Se trata de una comunidad conformada por desarrolladores y usuarios que mantienen un sistema operativo GNU/Linux basado en Software de Fuentes Abiertas precompilado y empaquetado en un formato en múltiples arquitecturas de computador y varios núcleos. La filosofía de este proyecto es la separación en las versiones del Software de Fuentes Abiertas de aquel que no lo es. En este sentido, Debian apuesta por una filosofía de usuario en el que el desarrollo se lleva adelante por los

usuarios con independencia de las empresas que hay de soporte. Adaptaciones conocidas de Debian son: Ubuntu y FreeBSD entre otras.

Ubuntu

Distribución Linux basada en Debian que proporciona un sistema operativo actualizado para un perfil de usuario medio, plenamente enfocado a la facilidad de uso e instalación del sistema.

Linux Empotrado

Es la versión del sistema operativo Linux utilizado en sistemas empotrados como por ejemplo, teléfonos móviles, PDAs, reproductores multimedia y otros dispositivos electrónicos, equipamiento de redes, automatismos industriales, equipos de navegación y equipamiento médico entre otros.

Las versiones de Linux para sistemas empotrados están adecuadas para dispositivos con recursos muy limitados, se limita el almacenamiento de memoria y se utilizan memorias flash en lugar de discos duros. Estas versiones de Linux se suelen optimizar dependiendo del propósito o uso que tenga el dispositivo destino. Estas optimizaciones pueden incluir el número de controladores, aplicaciones software o modificar el núcleo de Linux. En muchos casos, estas modificaciones van dirigidas a la consecución de un sistema operativo en tiempo real.

Dependiendo del tipo de dispositivo la mayoría de aplicaciones que se pueden encontrar en la distribución Linux para PC se suelen eliminar en las distribuciones para sistemas empotrados o se reemplazan por un pequeño conjunto de utilidades de fuentes abiertas, como por ejemplo

busybox. Las librerías también se suelen reemplazar por versiones mucho más compactas.

A continuación se señalan algunas de las arquitecturas a las que Linux ha sido portado:

Arquitecturas Linux empotrado
ARM
Avr32
Blackfin
Cris
Frv
H8300
IP7000
M32r
M68k
mips
Mn10300
PowerPC
Procesadores Xtensa

Tabla 9.1: Arquitecturas Linux empotrado

Las ventajas más comunes que se suelen señalar en el uso de Linux en sistemas empotrados son la no tenencia de derechos de autor o tasas de licencia, un núcleo del sistema operativo muy estable, un soporte de una extensa comunidad y la posibilidad de modificar y redistribuir el código. Una instalación típica de una distribución de Linux empotrado puede ocupar un promedio de 2MB, un peso aceptable dependiendo del tipo de dispositivo empotrado.

Las desventajas se pueden indicar como la complejidad del acceso de memoria, un gran conocimiento del mismo para lograr una configuración de los recursos del núcleo de manera que esté optimizado.

RTLinux

Es un sistema operativo de tiempo real que ejecuta Linux como un *thread* (hilo de ejecución) de menos prioridad que las tareas de tiempo real. Con este diseño, las tareas de tiempo real y los manejadores de interrupciones nunca se ven retrasados por operaciones que no son de tiempo real.

La primera versión de RTLinux estaba diseñada para ejecutarse en la plataforma x86 y proporcionaba una pequeña API y un pequeño entorno de programación. La versión 2, que fue totalmente reescrita, fue diseñada para el soporte de multiprocesamiento simétrico (SMP) y para ser ejecutada en una amplia variedad de arquitecturas.

RTLinux proporciona la capacidad de ejecutar tareas de tiempo real y manejadores de interrupciones en la misma máquina que el Linux estándar. Estas tareas y los manejadores ejecutan cuando se necesitan en detrimento de lo que estuviera ejecutando Linux. El peor caso de tiempo es entre que se detecta la interrupción hardware y el procesador ejecuta la primera instrucción del manejador de la interrupción. Este tiempo es del orden de los 10 microsegundos en la plataforma x86.

Las características más representativas de RTLinux son:

- Sistema operativo de tiempo real estricto.
- Extensiones para entorno multiprocesador SMP (x86).

- API "próxima" al de POSIX threads. Planificador expulsivo por prioridades fijas, señales, sistema de archivos POSIX (open, close, etc.) semáforos y variables condición.
- Depuración de código mediante GDB (GNU Debugger).
- Soporte para arquitecturas x86 y PPC.
- Acceso directo al hardware (puertos e interrupciones).
- Comunicación con procesos linux mediante memoria compartida y "tubos".
- Estructura modular para crear sistemas pequeños.
- Eficiente gestión de tiempos. En el peor caso se dispone de una resolución próxima al microsegundo (para un i486).
- Facilidades para incorporar nuevos componentes: relojes, dispositivos de E/S y planificadores.

9.2.1.2 >> Toppers

Se trata de un proyecto en el que se desarrollan un conjunto de sistemas en tiempo real de fuentes abiertas, entre otras aplicaciones. El kernel Toppers/JSP es un kernel en tiempo real que es conforme al profile estándar de la especificación de ?ITRON4.0 que tiene un enfoque para uso en industria y académico. Diversos proyectos se están realizando actualmente para añadir y extender funcionalidades como puede ser la creación y borrado de objetos dinámicos.

Se distribuye bajo una licencia propia que es similar a la licencia BSD.

9.2.1.3 >> eCOS

eCOS es un sistema operativo desarrollado en el ámbito de herramientas open source GNU. Este sistema operativo no tiene partes propietarias y proporciona la libertad de examinar y modificar el código según demanda. También permite el desarrollo y distribución de aplicaciones basados en el mismo.

Una clara innovación de este sistema operativo es que permite al desarrollador de aplicaciones imponer requisitos a los componentes en tiempo de ejecución y de esta forma implementar aplicaciones específicas conjuntamente con el sistema operativo. Esta característica permite personalizarlo para una gran variedad de sistemas empotrados.

El sistema de configuración presenta el sistema operativo como un componente dentro de la arquitectura, lo que establece un mecanismo estandarizado para extender sus funcionalidades y generar aplicaciones mediante una gran variedad de componentes configurables en tiempo de ejecución.

eCOS está diseñado para portarse a un gran número de arquitecturas y plataformas, incluyendo arquitecturas de 16, 32 y 64 bits, MPUs, MCUs y DSPs. El núcleo, librerías y componentes de eCOS están construidos sobre una capa de abstracción del hardware, que una vez implementada permitirá la ejecución del sistema operativo y aplicaciones basadas en él.

Actualmente las arquitecturas que eCOS soporta son:

Arquitecturas eCOS
68K/ColdFire
ARM (incluyendo ARM7TDMI, ARM9TDMI, Cortex-M, StrongARM, XScale)
CalmRISC16 y CalmRISC32 (solamente RedBoot)
Fujitsu FR-V
Fujitsu FR30
Hitachi H8/300
Intel x86
Matsushita AM3x
MIPS
NEC V8xx
PowerPC
SPARC
SuperH

Tabla 9.2: Arquitecturas eCOS

eCOS ha sido diseñado para soportar aplicaciones con requisitos de tiempo real, posee características de mínimos tiempos de latencia en interrupciones, primitivas de sincronización, políticas de planificación o gestión de tareas, mecanismos de gestión de interrupciones, etc. Este sistema operativo también proporciona las típicas funcionalidades que un sistema operativo para un sistema empotrado necesita: controladores de dispositivos, gestión de la memoria, gestión de excepciones, librerías matemáticas. Para facilitar el desarrollo de aplicaciones en sistemas empotrados se puede encontrar software que permite configurar el sistema operativo, así como compiladores, ensambladores, linkers, depuradores y simuladores.

A continuación se señalan las funcionalidades principales que ofrece eCOS:

- Capa de abstracción hardware (Hardware Abstraction Layer - HAL)
- Real-time kernel
 - Gestión de interrupciones
 - Gestión de excepciones
 - Diferentes gestores de tareas
 - Soporte para hilos
 - Conjunto de primitivas de sincronización
 - Relojes, contadores y alarmas
 - Diferentes asignadores de memoria

- Soporte para la depuración y simulación

- API compatible con μ ITRON 3.0
- API compatible con POSIX
- Librerías ISO C y matemáticas
- Controladores para puerto serie, ethernet, SPI, I2C, buffers de tramas, CAN, ADC, wallclock y watchdog
- Soporte para USB slave
- TCP/IP networking stacks
- Soporte para depuración GDB

La distribución eCOS puede ser utilizada en conjunto con las herramientas GNU que están disponibles en la red.

9.2.1.4 >> FreeBSD

FreeBSD es un sistema operativo libre para computadoras basado en las CPU de arquitectura Intel, incluyendo procesadores 386, 486 (versiones SX y DX), y Pentium. También funciona en procesadores compatibles con Intel como AMD y Cyrix. Actualmente también es posible utilizarlo hasta en once arquitecturas distintas como Alpha, AMD64, IA-64, MIPS, PowerPC y UltraSPARC.

FreeBSD es un sistema operativo multiusuario, capaz de efectuar multitarea con apropiación y multiproceso en plataformas compatibles con múltiples procesadores; el funcionamiento de FreeBSD está inspirado,

como ya se dijo, en la variante 4.4 BSD-Lite de UNIX. Aunque FreeBSD no puede ser propiamente llamado UNIX, al no haber adquirido la debida licencia de The Open Group, FreeBSD sí está hecho para ser compatible con la norma POSIX, al igual que varios otros sistemas "clones de UNIX".

El sistema FreeBSD incluye el núcleo, la estructura de ficheros del sistema, bibliotecas de la API de C, y algunas utilidades básicas. La versión 6.1 trajo importantes mejoras como mayor apoyo para dispositivos Bluetooth y controladores para tarjetas de sonido y red.

La versión 7.0, lanzada el 27 de febrero del año 2008, incluye compatibilidad con el sistema de archivos ZFS de Sun y a la arquitectura ARM, entre otras novedades.

TinyBSD es un conjunto de herramientas y scripts de shell, una parte de FreeBSD el sistema base, diseñado para hacer que el desarrollo de sistemas empotrados basado en FreeBSD lo más fácil posible. El TinyBSD script puede ser usado en FreeBSD 5.x, 6.x y 7.x y 8-CURRENT para crear una mini versión de FreeBSD.

La instalacion genera una imagen del sistema FreeBSD empotrado que es de unos 20 MB de tamaño y tiene un planteamiento muy genérico aunque puede ser reducida hasta 8MB.

NanoBSD es una herramienta actualmente desarrollada por Poul-Henning Kamp. Crea una imagen del sistema FreeBSD para aplicaciones empotradas lista para su uso en una tarjeta Compact Flash (u otro medio de almacenamiento masivo).

Puede utilizarse para generar imágenes instalables especiales; está diseñado pensando en una instalación y mantenimiento fáciles en

sistemas empotrados.

Las características de NanoBSD incluyen:

- Los ports y paquetes funcionan como en FreeBSD: Cualquier aplicación puede instalarse y usarse en una imagen de NanoBSD de la misma forma que en FreeBSD.
- No se pierde funcionalidad respecto a FreeBSD a menos que se haya eliminado explícitamente esa característica o características especiales al crear la imagen de NanoBSD.
- Todo está en modo de sólo lectura durante el funcionamiento del sistema: Un apagado brusco es totalmente seguro
- Fácil de generar y personalizar

9.2.1.5 >> CentOS

CentOS (Community ENTERprise Operating System) es un clon a nivel binario de la distribución Linux Red Hat Enterprise Linux RHEL, compilado por voluntarios a partir del código fuente liberado por Red Hat.

Red Hat Enterprise Linux se compone de Software de Fuentes Abiertas y código abierto, pero se publica en formato binario usable (CD-ROM o DVD-ROM) solamente a suscriptores pagados. Como es requerido, Red Hat libera todo el código fuente del producto de forma pública bajo los términos de la Licencia pública general de GNU y otras licencias. Los desarrolladores de CentOS usan ese código fuente para crear un producto final que es muy similar al Red Hat Enterprise Linux y está libremente disponible para ser bajado y usado por el público, pero no es mantenido

ni asistido por Red Hat. Existen otras distribuciones también derivadas de los fuentes de Red Hat.

CentOS usa yum para bajar e instalar las actualizaciones, herramienta también utilizada por Fedora

Las arquitecturas que son compatibles con este sistema empuotrado son:

Arquitecturas CentOS
i386
x86_64
AMD64
s390, s390x
Alpha
ppc32 (versión beta de CentOS)
Sparc (versión beta de CentOS)

Tabla 9.3: Arquitecturas CentOS

9.2.1.6 >> FreeRTOS

FreeRTOS es un sistema operativo para sistemas empuotrados distribuido bajo licencia GPL con una posible excepción que permite mantener cerrado código propietario mientras el kernel permanezca libre, de esta forma se facilita el uso de este sistema operativo en aplicaciones comerciales.

FreeRTOS se ha diseñado para ser pequeño y simple, con un núcleo que se puede resumir en tres o cuatro ficheros C para hacerlo comprensible, portable y mantenible. A pesar de estar escrito en C en su mayoría, hay partes que están implementadas en ensamblador, la mayoría dentro de la gestión de tareas. Las distribuciones se suelen acompañar de configuraciones y demostraciones permitiendo un rápido diseño de aplicaciones.

Las principales características son:

- Pequeño y simple, adecuado para desarrolladores noveles en sistemas operativos.
- Núcleo configurable en modo preventive, cooperative e híbrido.
- Versión SIL3 del sistema operativo certificado para el uso en aplicaciones críticas.
- Soporte oficial para 23 arquitecturas.
- FreeRTOS-MPU soporta Cortex M3 Memory Protection Unit (MPU).
- Código extremadamente portable al estar escrito en C.
- Soporta tareas y co-rutinas.
- Funcionalidad de trazas de ejecución.
- Opciones de detección de desbordamiento de pila.
- Sin restricción de número de tareas que se pueden crear.

- Sin restricción del número de prioridades.
- Sin restricción en la asignación de prioridades de tareas.
- Colas, semáforos, multiplexadores para comunicación, sincronización entre tareas o entre tareas e interrupciones.
- Multiplexadores con herencia de tareas.
- Herramientas libres de desarrollo (Cortex-M3, ARM7, MSP430, H8/S, AMD, AVR, x86 y 8051).
- Código libre empotrado.
- Libre de tasas.
- Aplicaciones y demostraciones para diferentes arquitecturas facilitando la curva de aprendizaje.

9.2.1.7 >> MicroC/OS II

MicroC/OS II es un kernel de sistema operativo, real y multitarea creado bajo la premisa "low-cost", para microprocesadores, escrito principalmente en C y diseñado específicamente para sistemas empotrados.

La designación II implica que es la segunda generación de un kernel publicado en un artículo de la revista Embedded Systems Programming y el libro μ C/OS.

Es un sistema operativo libre para la enseñanza y actividades no comerciales, aunque se puede licenciar para un producto determinado

o para una familia de productos. Está gestionado por Micrium, que proporciona otros productos software-middleware como uC/OS-View, uC/CAN, uC/TCP-IP, uC/FS, uC/GUI, uC/MOD-BUS, uC/LCD, uC/USB y una gran cantidad de aplicaciones basadas en uC/TCP-IP.

MicroC/OS II es un sistema operativo de tiempo real que ha sido creado bajo unos estrictos procesos de desarrollo, bajo estándares de código, con revisiones de código y una clara y concisa documentación. Estas características permiten su utilización para el cumplimiento de estándares "safety-critical" y requisitos en sectores muy estrictos como el militar, aeroespacial, medicina, transporte y automoción.

Este sistema operativo permite "round-robin scheduling", tiempos de deshabilitación de interrupciones cercanas a cero, ilimitado número de tareas, chequeo de errores, etc.

En la actualidad se está utilizando en los siguientes sectores de actividad:

- Aplicaciones
- Aeronáutica
- Salud
- Comunicaciones
- Bienes de consumo
- Control industriales
- Automoción
- Otros

En relación con este sistema operativo se pueden encontrar las siguientes herramientas:

- Entornos de desarrollo integrados (IDE)
- Compiladores y depuradores C/C++
- Kits de desarrollo
- Pruebas para chequeo de hardware
- Herramientas para máquinas de estados

9.2.1.8 >> Micro Núcleos

Micronúcleo o microkernel es un tipo de núcleo de sistema operativo que ofrece un conjunto de primitivas mínimas para implementar servicios básicos. Algunos ejemplos se muestran a continuación:

L4 es, colectivamente, una familia de programas de computadora relacionados. Se trata de una serie de micro núcleos que están llegando a ser muy conocidos en la industria de computadoras por su excelente funcionamiento y pequeño tamaño. Núcleo altamente optimizado para Intel i386, diseñado e implementado por Jochen Liedtke.

RadiOS es un micro núcleo libre, un clon de QNX Neutrino, escrito completamente en lenguaje assembly.

SymbOS es un sistema operativo multitarea para ordenadores de 8 bits basados en el Zilog Z80. Contrariamente a anteriores sistemas operativos de 8 bits, se basa en un micro núcleo, que realiza multitarea preferente (preemptiva), orientada a prioridades y administra una memoria RAM de hasta 576 Kb.

9.2.1.9 >> Contiki OS

Contiki SO es un sistema operativo de fuentes abiertas muy portable, y multitarea para sistemas de memoria restringida y pequeña. Soporta diversos formatos de computador (antiguos y modernos) como por ejemplo MSP430, x86, AVR, entre otros.

9.2.1.10 >> RTAI (Real Time Application Interface)

RTAI es una implementación de Linux para tiempo real basada en un principio en RTLinux. Se basa en el núcleo de Linux, proporcionando un pequeño núcleo de tiempo real bajo el estándar de Linux. Además proporciona una amplia selección de mecanismos de comunicación entre procesos y servicios de tiempo real. Por otro lado se facilitan aplicaciones para el desarrollo de aplicaciones de tiempo real.

9.2.2 >>Middleware

9.2.2.1 >> TAO/ACE

ACE ORB (TAO - The Ace Orb) es una implementación C++ de tiempo real de CORBA accesible libremente, open source. Está basada en ACE (Adaptive Communication Environment) y su objetivo es proporcionar una calidad de servicio (QoS) eficiente y predecible de principio a fin.

TAO aplica las mejores prácticas de software y modelos para automatizar la calidad de servicio en rendimiento y tiempo real a aplicaciones distribuidas. Se utiliza en desarrollos de aplicaciones emporadas distribuidas con unos requisitos de rendimientos muy altos.

TAO proporciona un marco de comunicaciones de propósito general que utiliza conceptos familiares a la industria del software que no realiza software de tiempo real.

9.2.2.2 >> gSOAP

El conjunto de herramientas gSOAP es un desarrollo abierto C y C++ para SAP/WSDL, servicios web XML y aplicaciones C/C++ de conexión de datos XML. gSOAP genera código para conexiones de datos XML y utiliza traductores para procesar datos XML. La generación automática de código ahorra sensiblemente tiempo de desarrollo de servicios SOAP/XML y el uso de conexiones de datos XML simplifica el uso de XML en aplicaciones mediante el mapeo automático de XML a tipos de datos C/C++.

Este conjunto de herramientas ha sido utilizado para desarrollar aplicaciones en sistemas empotrados, móviles, equipos de telecomunicaciones, routers, sistemas bancarios, etc.

9.2.2.3 >> CORBA

CORBA es el acrónimo de Common Object Request Broker Architecture, una arquitectura e infraestructura abierta e independiente de los vendedores que las aplicaciones software usan para trabajar conjuntamente sobre redes. Cualquier programa basado en CORBA que utilice el protocolo estándar IIOP, prácticamente en cualquier sistema operativo, lenguaje de programación y red puede interactuar con otro programa basado en CORBA que se esté ejecutando en otro sistema operativo, lenguaje de programación o red.

CORBA es muy útil en muchas situaciones porque permite la integración

de dispositivos de diferentes vendedores de muy diferente capacidad de procesamiento, lo que permite la interacción entre sistemas empotrados y PCs. Es un middleware utilizado por infinidad de compañías, proporciona fiabilidad, capacidad de crecimiento y tolerancia a fallos. Existen versiones de CORBA particularizadas para sistemas de tiempo real y sistemas empotrados de muy poca capacidad de procesamiento y de memoria.

9.2.2.4 >> D-Bus

D-BUS es un sistema de comunicación entre procesos que ayuda a coordinar el ciclo de vida de los procesos simplificándolo en una instancia de una aplicación o un demonio y lanzando aplicaciones y demonios bajo demanda según se necesiten los servicios. Está desarrollado como parte del proyecto freedesktop.org.

D-Bus es un sistema de baja latencia y baja carga de tráfico fácil de utilizar en la comunicación entre procesos.

- Baja latencia porque está diseñado para evitar retornos y permitir operaciones asíncronas, de un modo parecido al protocolo X
- Baja carga de tráfico porque utiliza un protocolo binario y no necesita convertir a formato texto.
- Fácil de utilizar porque trabaja en términos de mensajes en lugar de tramas de bytes y gestiona automáticamente muchos problemas IPC.

D-Bus está diseñado para dos usos específicos

- Un bus de sistema para notificaciones del sistema a sesiones de

usuario y para permitir al sistema realizar peticiones de datos a las sesiones de usuario

- Un bus de sistema implementar entornos de escritorio como GNOME y KDE

D-Bus proporciona un demonio de sistema para eventos del tipo nuevo hardware encontrado o cambio en la cola de impresión, y un demonio de establecimiento de sesión por usuario. El bus de mensajes está construido en un marco de tratamiento de mensajes uno a uno que puede ser utilizado por dos aplicaciones que se comunican directamente sin utilizar el demonio de bus de mensajes.

La implementación de la API de bajo nivel y el protocolo han sido probados durante siete años en el mundo real y se encuentran en una fase totalmente estable.

9.2.2.5 >> OSGi

La OSGi Alliance (<http://www.osgi.org/>) es una organización de estándares abiertos que ha especificado una plataforma basada en Java, que puede ser gestionada de manera remota. El núcleo de las especificaciones es un 'framework' que define un modelo de ciclo de vida de aplicaciones y un repositorio de servicios. Esta es la base sobre la que se han definido una gran cantidad de servicios OSGi: Log, gestión de configuración, preferencias, servicio http (ejecuta servlets), 'parseo' de XML, acceso a dispositivos, administración de paquetes, administración de permisos, administración de usuarios, conector I/O, Jini, exportación a UPnP, seguimiento de aplicaciones, distribuciones firmadas, servicios declarativos, gestión de energía, gestión de dispositivos, políticas de seguridad, diagnóstico, etc.

La plataforma implementa un modelo de componentes dinámico y completo, algo que faltaba en los entornos Java/VM para aplicaciones autónomas ('standalone'). Las aplicaciones o componentes (desplegados en forma de distribuciones) pueden ser instalados remotamente, ejecutados, parados, actualizados y desinstalados sin la necesidad de reiniciar el sistema. La gestión de paquetes y clases Java está especificada con mucho detalle. Esta gestión del ciclo de vida se realiza mediante unas APIs que permiten la descarga (de forma remota) de políticas de gestión. El repositorio permite que las distribuciones detecten nuevos servicios, o la desconexión de los mismos, y se adapten consecuentemente.

El foco original eran las pasarelas de servicios, pero la aplicabilidad de la plataforma ha demostrado ser mucho mayor. Las especificaciones OSGi ahora se utilizan en múltiples aplicaciones, desde teléfonos móviles al IDE de código abierto Eclipse. Otras áreas de aplicación incluyen coches, automatización industrial, automatización de edificios, PDAs, computación Grid, entretenimiento (p. ej. Philips iPronto), gestión de flotas y servidores de aplicación. En conclusión, OSGi es la plataforma ideal de despliegue y alojamiento para cualquier entorno donde se quiera llevar a cabo la búsqueda de servicios, así como ejecución y composición de los mismos.

Bajo la filosofía OSGi se han desarrollado diversos proyectos, que se enumeran a continuación:

- OCAR (<http://oscar.objectweb.org>)
- Knoflefish (<http://www.knopflerfish.org>)
- Eclipse OSGi run time (<http://www.eclipse.org>)
- Apache Felix (<http://felix.apache.org/site/index.html>)

9.2.2.6 >> OpenSplice DDS

OpenSplice DDS es una implementación de fuentes abiertas de OMG para servicios de distribución de datos para especificaciones de sistemas de tiempo real (DDS). Esta implementación ofrece una infraestructura de altas prestaciones en tiempo real para publicar y suscribir mensajes. Existe una comunidad del mismo nombre. Esta implementación se encuentra bajo la licencia LGPL.

9.2.3 >>Entornos de desarrollo

9.2.3.1 >> Eclipse

Eclipse es una herramienta universal que permite integrar diferentes aplicaciones para construir un entorno integrado de desarrollo (IDE). Es un proyecto de desarrollo de software open-source, que está dividido en tres partes: el Eclipse Project, el Eclipse Tools Project, y el Eclipse Technology Project.

El Eclipse Project está subdividido a su vez en tres sub-proyectos que son: la propia plataforma, JDT (Java Development Tool) y PDE (Plug-in Development Environment).

Mediante Eclipse se pueden crear diversas aplicaciones como pueden ser sitios Web, programas Java, C++ y Enterprise Java Beans.

La aplicación principal de Eclipse Project es JDT, herramienta para crear aplicaciones en Java. Además, pueden integrarse a Eclipse otras aplicaciones en forma de plug-ins, que son reconocidos automáticamente

por Eclipse al iniciar el mismo.

Como Eclipse está escrito en Java, para su funcionamiento se debe tener instalado el JRE (Java Runtime Environment). Eclipse detecta automáticamente la ubicación de la JRE instalada.

La funcionalidad que proporciona Eclipse proviene de dos fuentes: de un pequeño núcleo conocido como el Platform Runtime o de plug-ins. Existe un conjunto de plug-ins que ya vienen con la plataforma.

Un **plug-in** es la unidad mínima de la plataforma que puede ser desarrollada independientemente. Se pueden encontrar herramientas pequeñas desarrolladas en un sólo plug-in o herramientas mucho más complejas que se componen de un conjunto de plug-ins que se comunican entre sí

Eclipse proporciona la capacidad de arrancar el programa con un conjunto de plug-ins predeterminado, permitiendo acceder a distintas aplicaciones sin necesidad de levantar todas a la vez al momento de ejecutarlo. Al comenzar Eclipse, el Runtime descubre cuáles son los plug-ins que se encuentran disponibles, lee su información desde sus propios archivos de manifiesto, y construye lo que se denomina plug-in-registry.

Otros subproyectos de interés:

- EMF: Gestión de (meta)Modelos. <http://www.eclipse.org/emf/>
- OCL, Validation, Query, Transaction: Extensiones/Complementos de EMF ("EMF Tools"). <http://www.eclipse.org/emft/>

- GEF: Entorno de creación de elementos gráficos.
<http://www.eclipse.org/gef/>
- GMF: Extensión de EMF/GEF para la creación de editores gráficos.
<http://www.eclipse.org/gmf/>
- MOFScript: Lenguaje para la transformación de modelos a código.
<http://www.eclipse.org/gmt/mofscript/>
- Moskitt (Modeling Software KIT) (MOSKitt) es una herramienta open source con diversos módulos para el manejo de requisitos: captura e identificación, atributos, control de cambios, definición de criterios de clasificación, parametrización de atributos y representación.

9.2.3.2 >> Kdevelop

Entorno de desarrollo integrado open source que da soporte a varios lenguajes de programación y es extensible para que soporte otros lenguajes de programación puedan ser añadidos por parte de otras comunidades.

9.2.4 >>Herramientas

9.2.4.1 >> Papyrus

Papyrus es una herramienta para el modelado UML2. Se trata de una herramienta open-source basada en Eclipse. La herramienta ofrece un elevado grado de compatibilidad con los estándares de modelado del

OMG y una muy buena integración con otras herramientas orientadas a modelos dentro de Eclipse.

Papyrus soporta varios de los más importantes perfiles UML estandarizados como SysML, CCM e incluso MARTE. La herramienta ha sido desarrollada con una arquitectura extensible y es sencillo añadir nuevos diagramas, perfiles UML o generadores de código.

Papyrus también permite a los usuarios generar código Java, C o C++ automáticamente a partir de los modelos empleando la tecnología de otro plug-in, ACCELEO. Además Papyrus soporta también el lenguaje de transformaciones Atlas (ATL).

En la actualidad el plug-in de Papyrus soporta los diagramas de clases, composición, casos de uso, máquinas de estados, actividad y secuencia. Además Papyrus incluye algunos de los más importantes perfiles de UML para el modelado hardware y software listos para utilizar, entre otros MARTE, SysML y CCM.

Una de sus principales características es la posibilidad de desarrollar nuevos perfiles UML que además podrán ser exportados y utilizados en otras herramientas compatibles con los modelos UML2.

9.2.4.2 >> UML TOPCASED

TOPCASED es un grupo de herramientas open-source para la ingeniería de sistemas y software compatibles con los requisitos de aplicaciones empotradas.

TOPCASED es una metaherramienta ya que es posible desarrollar editores gráficos y transformaciones de modelos a partir de ella. Varios

editores han sido creados ya en el proyecto TOPCASED: un editor de Ecore, un editor de UML, un editor de SysML, un editor gráfico de AADL y un editor gráfico de SAM.

El grupo de herramientas TOPCASED incluye un editor compatible con UML2 y los ficheros UML2. Este editor soporta los diagramas UML de clases, despliegue, secuencia, máquinas de estados y casos de uso.

El editor de diagramas es sencillo de utilizar y permite acceder a las propiedades de los elementos UML, así como a las constraints y los estereotipos.

Por último el editor de UML soporta perfiles, tanto generar nuevos como utilizar perfiles existentes; sin embargo no existe disponible ningún repositorio de perfiles listos para utilizar. Es posible emplear un plugin para utilizar el repositorio de perfiles UML de Papyrus.

9.2.4.3 >> Cmake

Sistema para automatizar la generación, testeo y empaquetado de software. CMake es una solución open-source inter-operable en diferentes plataformas que permite controlar el proceso de compilación usando ficheros independientes al compilador. También permite generar makefiles que pueden ser usados en entornos de compilación existentes.

9.2.4.4 >> MinGW

MinGW significa "GNU mínima para Windows" y es una solución open source para el desarrollo de aplicaciones nativas de Microsoft Windows que incorpora un puerto para GCC y GNU Binutils.

9.2.4.5 >> Bugzilla

Bugzilla es un sistema de gestión de defectos o de gestión de bugs. Este tipo de herramientas permiten que los desarrolladores individuales o los grupos de desarrollo para mantener un seguimiento de defectos y bugs en su producto de forma efectiva.

La herramienta está basada en un marco de herramientas open-source: Apache, PHP, MySQL. La interacción con los usuarios se realiza a través de navegadores Web, evitando de esta forma la instalación de clientes en los equipos de los usuarios.

Las características más importantes de Bugzilla son las siguientes:

- Estructura de base de datos optimizada.
- Seguridad para asegurar la confidencialidad.
- Una herramienta de queries capaz de recordar las búsquedas de los usuarios.
- Integración con tecnologías de correo electrónico.
- Perfiles de usuarios editables.
- Un sistema de privilegios comprensible.

Empleando las capacidades del marco de herramientas es posible gestionar tanto bugs como tareas dentro de los proyectos. Además, dado que Bugzilla utiliza un interfaz de usuario Web, es sencillo integrarlo con otras herramientas de los usuarios finales.

9.3

>> Productos/Tipo software específicos del dominio

9.3.1

>>Procesado de imágenes

9.3.1.1 >> OpenGL

OpenGL (Open Graphics Library) es una especificación estándar que define una API multilenguaje y multiplataforma para escribir aplicaciones que produzcan gráficos 2D y 3D.

La interfaz consiste en más de 250 funciones diferentes que pueden usarse para dibujar escenas tridimensionales complejas a partir de primitivas geométricas simples, tales como puntos, líneas y triángulos.

Fue desarrollada originalmente por Silicon Graphics Inc. (SGI) en 1992[1] y se usa ampliamente en CAD, realidad virtual, representación científica, visualización de información y simulación de vuelo. También se usa en desarrollo de videojuegos, donde compite con Direct3D en plataformas Microsoft Windows.

9.3.1.2 >> Cimg

La librería Cimg es una herramienta de fuentes abiertas para procesado de imagen, implementado en C++. Las principales características son:

- Utilidades genéricas para el manejo de imágenes en el código C++
- Portabilidad a diferentes sistemas operativos y con soporte para diversos compiladores C++
- Simple y ligero
- Extensibilidad ya que permite el uso de de librerías y herramientas externas

9.3.2

>>Interfaces gráficas

9.3.2.1 >> QT

Qt es una biblioteca multiplataforma para desarrollar interfaces gráficas de usuario y también para el desarrollo de programas sin interfaz gráfica como herramientas de la consola y servidores.

Qt es utilizada principalmente en KDE, Google Earth, Skype, Qt Extended, Adobe Photoshop Album, VirtualBox y Opie. Es producido por la división de software Qt de Nokia, que entró en vigor después de la adquisición por parte de Nokia de la empresa noruega Trolltech, el productor original de Qt, el 17 de junio de 2008.

Qt es utilizada en KDE, un entorno de escritorio para sistemas como GNU/Linux o FreeBSD, entre otros. Qt utiliza el lenguaje de programación C++ de forma nativa, adicionalmente puede ser utilizado en otros lenguajes de programación a través de bindings.

Qt se encuentra disponible para sistemas tipo unix con el servidor gráfico X Window System (Linux, BSDs, Unix), para Apple Mac OS X, para sistemas Microsoft Windows, para Linux empotrado (para sistemas integrados como PDA, Smartphone, etc.) y para dispositivos que utilizan Windows CE. La última incorporación de esta librería ha sido el sistema operativo Maemo 5.

Funciona en todas las principales plataformas, y tiene un amplio apoyo. Algunas de sus características son:

- Métodos para acceder a bases de datos mediante SQL
- Uso de XML
- Gestión de hilos
- Soporte de red
- API multiplataforma unificada para la manipulación de archivos
- Manejo de ficheros
- Manejo de estructuras de datos tradicionales.

Distribuida bajo los términos de GNU Lesser General Public License (y otras), Qt es Software de Fuentes Abiertas y de código abierto.

9.3.2.2 >> X, Xfree86

X Window System fue desarrollado a mediados de los años 1980 en el MIT (Instituto Tecnológico de Massachusetts) para dotar de una interfaz gráfica a los sistemas Unix. Este protocolo permite la interacción gráfica en red entre un usuario y una o más computadoras haciendo

transparente la red para éste. Generalmente se refiere a la versión 11 de este protocolo, X11, el que está en uso actualmente. X es el encargado de mostrar la información gráfica de forma totalmente independiente del sistema operativo.

Debido al esquema cliente-servidor que posee, se puede decir que X se comporta como una terminal gráfica virtual.

El hecho que exista un estándar definido para X permite que se desarrollen servidores X para distintos sistemas operativos y plataformas, lo que hace que el código sea muy portable

X no es un gestor de ventanas, necesita de uno para controlar el manejo de ventanas. Esto trae la ventaja de que permite al usuario instalar uno o más administradores de ventanas de su preferencia. También trae la ventaja de que hace de X estrictamente un sistema gráfico, de tal modo que un cliente X podría estar enviando un gráfico a una pantalla, a una impresora o a cualquier otro hardware sin darse cuenta, flexibilizando la salida gráfica.

XFree86 es una implementación del sistema X Window System. Fue escrita originalmente para sistemas operativos UNIX funcionando en ordenadores compatibles IBM PC. En la actualidad está disponible para muchos otros sistemas y plataformas.

XFree86 es open source y Software de Fuentes Abiertas, publicado bajo la licencia XFree86 1.1.

El proyecto XFree86 es desarrollado por el XFree86 Project, Inc.; siendo su desarrollador líder David Dawes.

XFree86 provee una interfaz gráfica cliente/servidor entre el hardware (sistemas gráficos y dispositivos de entrada, como el mouse o el teclado) y un entorno de escritorio que provee un sistema de ventanas así como una interfaz estandarizada de aplicación.

XFree86 es independiente de la plataforma, extensible y puede utilizarse en red. Funciona actualmente en una gran variedad de sistemas UNIX como BSDs (FreeBSD, NetBSD, OpenBSD, Mac OS X vía Darwin, etc), Solaris, SGI IRIX, Linux, así como en OS/2 y Cygwin (para Windows).

Un cambio de licencia producido en Febrero de 2004 a partir de la versión 4.4.0 (anteriormente se distribuía bajo la licencia MIT) provocó la creación de la bifurcación X.Org Server, apoyada por empresas y desarrolladores descontentos con presuntas incompatibilidades con la popular licencia GPL. Esto ha provocado una caída en la popularidad de XFree86, siendo reemplazado por X.Org en prácticamente todas las distribuciones de GNU/Linux y en algunos sistemas BSD.

Aún así, XFree86 es utilizado y distribuido en muchos otros sistemas operativos, como por ejemplo NetBSD, Mac OS X (como entorno alternativo) y FreeBSD (como alternativa a X.Org).

9.3.2.3 >> GNOME Mobile, GTK, Clutter

GNOME es un entorno de escritorio e infraestructura de desarrollo para sistemas operativos Unix y derivados Unix como GNU/Linux, BSD o Solaris; compuesto enteramente de Software de Fuentes Abiertas.

El proyecto fue iniciado por los programadores mexicanos Miguel de Icaza y Federico Mena y forma parte oficial del proyecto GNU. Nació como una alternativa a KDE bajo el nombre de GNU Network Object

Model Environment. Actualmente se encuentra disponible en 48 idiomas, entre ellos el español.

El Proyecto GNOME, según sus creadores, provee un gestor de ventanas "intuitivo y atractivo" y una plataforma de desarrollo para crear aplicaciones que se integran con el escritorio. El Proyecto pone un gran énfasis en la simplicidad, usabilidad y eficiencia. Otros objetivos del proyecto son:

- La libertad para crear un entorno de escritorio que siempre tendrá el código fuente disponible para reutilizarse bajo una licencia de Software de Fuentes Abiertas.
- El aseguramiento de la accesibilidad, de modo que pueda ser utilizado por cualquiera, sin importar sus conocimientos técnicos y discapacidad física.
- Hacer que esté disponible en muchos idiomas. En el momento está siendo traducido a más de 100 idiomas.
- Un ciclo regular de liberaciones y una estructura de comunidad disciplinada

GNOME Mobile es un avance en el uso, desarrollo y comercialización de los componentes GNOME en móviles y plataformas de sistemas empotrados. GNOME Mobile engloba industria, consultores expertos, desarrolladores y organizaciones industriales.

Sus objetivos son:

- Incrementar la comunidad y visibilidad comercial de la tecnología

GNOME Mobile

- Coordinar los esfuerzos dentro de GTK+ y la plataforma GNOME, así como avances en documentación y herramientas de desarrollo.
- Facilitar la colaboración para el desarrollo entre las organizaciones participantes
- Impulsar las oportunidades de estandarización de la plataforma
- Asegurar la libertad del software y su disponibilidad en todo el mundo

La plataforma GNOME Mobile es un subconjunto de la plataforma GNOME. A continuación se mencionan las tecnologías que engloba la plataforma:

- Bluez: Bluetooth
- E-D-S: calendario y contactos
- Telepathy: mensajería instantánea, VoIP y videoconferencia
- Avahi: búsqueda de servicios
- GStreamer: multimedia
- Matchbox: gestión de ventanas
- GConf: Sistema de almacenamiento de preferencias de aplicaciones
- SQLite: bases de datos
- GVFS y GIO: sistema de ficheros

- GTK+: Interfaz de usuarios
- GLib y DBus: infraestructura del núcleo

Proyectos como OLPC XO, FIC neo 1983, Nokia N800 y Vernier LabQuest utilizan esta plataforma.

GTK+ o **The GIMP Toolkit** es un conjunto de bibliotecas multiplataforma para desarrollar interfaces gráficas de usuario (GUI), principalmente para los entornos gráficos GNOME, XFCE y ROX aunque también se puede usar en el escritorio de Windows, MacOS y otros.

Inicialmente fueron creadas para desarrollar el programa de edición de imagen GIMP, sin embargo actualmente se usan mucho por muchos otros programas en los sistemas GNU/Linux. Junto a Qt es uno de las bibliotecas más populares para X Window System.

GTK+ se ha diseñado para permitir programar con lenguajes como C, C++, C#, Java, Ruby, Perl, PHP o Python.

Licenciado bajo los términos de LGPL, GTK+ es Software de Fuentes Abiertas y es parte del proyecto GNU.

Clutter es una librería gráfica libre para creación de interfaces de usuario rápidos, ricos visualmente y portables. Se basa en OpenGL y puede ser compilada para diferentes plataformas (X11, Darwin y Win32).

Clutter soporta reproducción multimedia mediante GStreamer y utilización de gráficos 2D mediante Cairo.

Clutter fue creada por OpenedHand Ltd., ahora parte de Intel y tiene licencia LGPL.

9.3.3 >>Comunicaciones

9.3.3.1 >> lwIP

lwIP es una implementación muy ligera del protocolo TCP/IP. Fue originalmente creada por Adam Dunkels, del Swedish Institute of Computer Science, pero en la actualidad está siendo desarrollada por un equipo de desarrolladores distribuidos por todo el mundo, todos ellos dirigidos por Leon Woestenberg.

Desde su aparición, lwIP ha generado un gran interés y en la actualidad está siendo utilizada en una gran cantidad de productos comerciales. Ha sido portada a múltiples plataformas y sistemas operativos, incluso puede correr sin un sistema operativo.

El objetivo de esta implementación es reducir el uso de la RAM a la vez que se mantiene una funcionalidad completa del protocolo TCP. Esta característica hace que lwIP sea apropiada para el uso en sistemas emporrados que pueden incorporar una librería con una capacidad de decenas de kB de RAM libre y unos 40kB de ROM.

Las principales características de esta librería son:

- IP (Internet Protocol), incluyendo reenvío de paquetes a través de múltiples interfaces de red.
- ICMP (Internet Control Message Protocol) para mantenimiento de red y depuración
- UDP (User Datagram Protocol)
- TCP (Transmission Control Protocol) con control de congestión, estimación with congestion control, estimación de RTT (Round-Trip

Time) y rápida recuperación y retransmisión de paquetes

- Posibilidad de una API para Berkeley socket

lwIP está disponible bajo licencia BSD en código C.

9.3.3.2 >> Telepathy

El proyecto Telepathy crea una plataforma para diferentes tipos de comunicaciones en tiempo real. Telepathy utiliza un sistema de mensajes D-Bus para proporcionar una interfaz sencilla para aplicaciones clientes, permitiendo que estas adquieran las características de Telepathy que a continuación se mencionan:

- Tiempo real: soporta mensajería instantánea (uno a uno o en grupos), llamadas de voz y videoconferencia.
- Unificación: diferentes programas pueden aprovechar estas comunicaciones, la plataforma Telepathy permite a dichos programas trabajar conjuntamente.
- Plataforma: permite el manejo de diferentes aspectos de la comunicación entre diferentes partes del sistema simplificando de esta forma cada una de las partes.

9.3.4 >>Móviles

9.3.4.1 >> Maemo

Maemo es una plataforma software desarrollada por Nokia para teléfonos móviles inteligentes y Tablet PC basada en Linux, concretamente la distribución Debian (lo que permite integrar innumerables paquetes

software de esta distribución). Así mismo, integra muchas de las librerías y plataformas GUI de GNOME.

La plataforma integra el sistema operativo Maemo y el kit de desarrollo Maemo SDK. Algunas de sus características son:

- Internet
- Aplicaciones ofimáticas
- Múltiples escritorios personalizables
- Multitarea
- Mensajería instantánea, VoIP y diferentes conectividades
- Geolocalización
- Adobe Flash
- Facilidad de actualización
- Multimedia

El proyecto estrella que se basa en esta plataforma es el teléfono móvil inteligente Nokia N900.

9.3.4.2 >> Moblin

Moblin es un proyecto de código abierto para el desarrollo de software orientado a dispositivos móviles enfocados principalmente en la conectividad a Internet (MIDs), así como netbooks y nettops.

El proyecto fue creado por Intel en 2007 e impulsado con el lanzamiento de la familia de procesadores Intel Atom en el 2008, cuando ya se abandonó la versión beta. Posteriormente se fueron agregando más empresas hasta incluir el proyecto bajo el amparo de la Linux Foundation.

Moblin presenta una interfaz de usuario que oculta completamente el sistema subyacente. La idea no es nueva y ya lo había hecho Nokia con sus Internet Table, los antecesores de los actuales netbooks. Se pretende que el usuario no tenga que lidiar con las complejidades de un sistema completo y se enfoque en lo que necesita hacer, y no en cómo lo va a hacer, algo comúnmente conocido como diseño orientado a las tareas (task based design).

Los componentes principales de la plataforma son:

- Moblin Image Creator (MIC): permite a los desarrolladores el crear un sistema Linux adaptado para un dispositivo. Utilizando MIC, un desarrollador de una plataforma puede elegir qué componentes de Moblin quieren integrar en su dispositivo, construir el sistema, copiar todos los archivos necesarios a un pen drive USB y cargarlo en el dispositivo destino.
- Kernel: parches específicos de la plataforma al kernel de Linux y muchos otros drivers para dispositivos.
- UI Framework: interfaz gráfica y el framework subyacente basado en GTK, que utiliza el framework de aplicaciones Hildon.
- Política de gestión de energía: extiende y mejora las capacidades de gestión de energía existentes en Linux.
- Browser: el navegador Moblin es un navegador web basado en tecnologías Mozilla con un interfaz manejado con el dedo y con integración con el UI del MID. El navegador Moblin soporta plugins como el Adobe Flash.
- Multimedia: sonido, vídeo y visualización de imágenes incluyendo los frameworks multimedia Helix o GStreamer.

- Linux Connection Manager: Conexiones a Internet que pueden ser ampliadas a través de plugins que soporten múltiples tecnologías inalámbricas o de cable.

9.3.4.3 >> LiMo

La fundación LiMo es un consorcio industrial para crear el primer sistema operativo basado en Linux realmente independiente del hardware y abierto. Respaldo por líderes industriales del sector móvil, se pretende que LiMo unifique esfuerzos en torno a Mobile Linux.

La misión de la fundación LiMo es crear una plataforma software basada en Linux para el uso de toda la industria relacionada con dispositivos móviles mediante un proceso de contribuciones transparentes y equilibradas que permitan un ecosistema de productos, aplicaciones y servicios por parte de fabricantes de dispositivos, operadores vendedores de servicios e integradores.

LiMo se proclama como un sistema operativo Middleware con una base tecnológica que permite una eficiencia en el coste del desarrollo de diferentes equipos y servicios. La plataforma consiste en contribuciones de los miembros de la fundación y la adopción de tecnologías libres con costes y resultados compartidos por todos los miembros. La no definición de patentes permite contribuciones sin tasas de licencias. El conjunto de la plataforma tiene una licencia FPL.

La fundación LiMo seleccionó Linux como núcleo de la plataforma debido a su funcionalidad, posibilidad de crecimiento y gran aceptación dentro de la industria de sistemas empotrados móviles. Linux permite también el objetivo de ampliar el objetivo de dispositivos de la plataforma a otros dispositivos electrónicos de consumo y automoción así como incorporar a la extensa comunidad de desarrolladores Linux al área móvil.

Las aplicaciones pueden acceder los componentes middleware a través de una API, para aumentar la flexibilidad del diseño, los componentes se pueden implementar en C o C++

Los componentes principales son:

- Marco gestor de aplicaciones
- Registro
- Gestión de conflictos
- Transmisión de eventos / IPC
- Marco de seguridad
- Marco de telefonía
- Marco de red
- Marco de mensajería
- Marco Multimedia
- Marco DRM (Digital Rights Management)
- Base de datos
- Otros marcos

9.3.4.4 >> Android

Los dispositivos móviles han experimentado un importante auge en los últimos años. Es por ello que se ha empezado a realizar un gran esfuerzo en la comunidad de fuentes abiertas para el desarrollo de sistemas y aplicaciones adaptados a este tipo de dispositivos y a sus características. En este contexto, ha aparecido Android. Se trata de un sistema operativo para dispositivos móviles y computadores basado en un kernel de Linux,

desarrollado por Google y actualmente por Open Handset Alliance. Este proyecto nace con la idea de aunar esfuerzos en varios niveles: hardware, software y de telecomunicaciones, tratando de establecer un estándar para este tipo de dispositivos móviles. La filosofía de fuentes abiertas permite un trabajo colaborativo para el desarrollo, y del mismo modo se han establecido una serie de políticas para gestionar y coordinar este trabajo. La mayor parte del código fuente está publicado bajo licencia Apache.

Una de las grandes empresas que ha apostado por Android, es el fabricante HTC que actualmente provee de una gran cantidad de dispositivos. Las principales características de Android se pueden resumir en los siguientes puntos[Android-Developpers]:

- Framework de aplicaciones: permite reutilización y reemplazo de componentes.
- Máquina virtual Dalvik: optimizada para dispositivos móviles.
- Navegador integrado: basado en el motor de código abierto WebKit.
- Gráficos optimizados, con una biblioteca de gráficos 2D; gráficos 3D basado en la especificación OpenGL ES 1.0 (aceleración por hardware opcional).
- SQLite para almacenamiento de datos estructurados.
- Soporte para medios con formatos comunes de audio, vídeo e imágenes planas (MPEG4, H.264, MP3, OGG, AAC, AMR, JPG, PNG, GIF)
- Telefonía GSM (dependiente del hardware)
- Bluetooth, EDGE, 3G, y WiFi (dependiente del hardware)
- Cámara, GPS, brújula, y acelerómetro (dependiente del hardware)
- Rich development environment, que incluye un emulador de dispositivo, herramientas para depurar, perfiles de memoria y rendimiento, y un plugin para el IDE Eclipse.

9.3.4.5 >> OpenMoko

Openmoko es un proyecto de código abierto para crear el primer sistema operativo libre para teléfonos móviles a nivel mundial. Openmoko actualmente está vendiendo el teléfono Neo FreeRunner a usuarios avanzados con conocimientos técnicos y empezará la venta al público tan pronto con el software este lo suficientemente desarrollado.

El sistema operativo Openmoko, que incluye un servidor X completo, permite a los usuarios y a los desarrolladores transformar el hardware de los móviles en productos únicos. Nuestra licencia proporciona a los desarrolladores y a los usuarios libertad para realizar ligeros cambios estéticos o transformarlo radicalmente; cambiar el fondo de pantalla o reprogramar desde cero el móvil entero. La licencia te garantiza la libertad de por ejemplo, transformar el teléfono en un dispositivo médico o en un punto de venta o simplemente instalar tus juegos favoritos.

9.3.5 >>Multimedia

9.3.5.1 >> Gstreamer

GStreamer es un framework multimedia libre multiplataforma escrito en el Lenguaje de programación C, usando la biblioteca GObject.

GStreamer permite crear aplicaciones multimedia, como vídeo, sonido, codificación, etc. Por ejemplo, con GStreamer se puede reproducir música o realizar tareas más complejas como mezclar audio y vídeo.

GStreamer recibe hospedaje en freedesktop.org, lo que de algún modo permite una mayor interoperabilidad entre distintos sistemas libres. Inicialmente fue adoptado por el proyecto GNOME, a partir de su versión 2.2, luego siguió su evolución y cada vez más aplicaciones lo utilizan.

GStreamer, ha sido portado a variados sistemas operativos (Linux, MacOS, Windows, Solaris), procesadores (x86, SPARC, PPC, ARM) y compiladores (GCC, MS Visual Developer). Prueba de ello son otros dispositivos como el Nokia 770 y Nokia 800 (basados en la plataforma de desarrollo Maemo), los que hacen uso de GStreamer.

La función del núcleo de GStreamer es proveer un framework para plugins, flujo de datos y manejo/negociación de distintos tipos de medios. También provee una API para escribir aplicaciones.



La industria de los sistemas empotrados y en general de los sistemas con inteligencia integrada depende cada vez más del software. El impacto en la productividad y eficiencia de los productos finales depende en gran medida del tipo de software implicado en las diferentes fases del ciclo de vida, constituyendo parte del producto en forma de avanzadas plataformas software o sistemas operativos en tiempo real o un medio de soporte, mediante herramientas básicas o avanzadas en forma de IDE, integrando capacidades de co-diseño, análisis, verificación o simulación.

Centrándonos en nuestro entorno cercano, la industria española desarrolladora e integradora de sistemas empotrados se encuentra actualmente alineada con sus equivalentes europeos e internacionales en diferentes ámbitos y sectores, adoleciendo al mismo tiempo de los mismos problemas y dificultades. El mercado de los sistemas empotrados se ha caracterizado tradicionalmente por modelos restrictivos de patentado como elemento diferenciador entre empresas; no obstante, el competitivo entorno internacional, apunta a la implantación del Software de Fuentes Abiertas como una tendencia prometedora en la productividad de las empresas, adquiriendo de forma creciente visibilidad de cara al usuario final. Desde PROMETEO se percibe cómo el Software de Fuentes Abiertas se posiciona de forma global como un medio y fin cada vez más extendido en la industria.

La existencia de estudios que abordan tendencias en los sistemas empotrados es clara y visible, tanto nacional como internacional, no así el impacto del Software de Fuentes Abiertas en este ámbito. Este panorama presenta la necesidad de abordar un análisis que permita mostrar la utilización del Software de Fuentes Abiertas dentro de este mercado tan competitivo, abarcando la totalidad de la cadena de valor y los diferentes sectores en los que se puede enmarcar.

Con el objetivo de conocer la casuística existente, CENATIC, atendiendo a su carácter investigador y difusor ha promovido la realización del presente Informe sobre la situación del Software de Fuentes Abiertas en las empresas españolas proveedoras e integradoras de sistemas empotrados.

PROMETEO, en la figura de su Comité Ejecutivo, quiere declarar su apoyo a las nuevas tecnologías orientadas a la productividad del sector, como las ofrecidas por el modelo de Software de Fuentes Abiertas. Es por ello que PROMETEO se congratula en secundar este estudio que establece un punto de partida a la hora de descubrir e indicar nuevas líneas estratégicas que permitan un liderazgo de nuestro tejido empresarial.

Comité Ejecutivo de PROMETEO

Francisco Ramos, Jesus Bermejo. Grupo Telvent.

Sergio Bandinelli. Tecnalia Corporación Tecnológica.

José Luis Burón. Acciona.

Juan Carlos Dueñas. Universidad Politécnica de Madrid.

Jesús Ángel Sánchez. Indra.

Iñaki Larrañaga. Mondragón.

Observatorio Nacional del
Software de Fuentes Abiertas



Patronato de CENATIC:

