


- » BOOST PERFORMANCE
- » REDUCE COST
- » INCREASE AGILITY
- » ENHANCE CRM
- » SHORTEN TIME TO MARKET
- » DRIVE INNOVATION
- » IMPROVE EFFICIENCY
- » INCREASE ADAPTIVITY
- » ENABLE BUSINESS
- » ENSURE REG



CONSULTING > SOLUTIONS > OUTSOURCING

# Framework de desarrollo de Código Abierto

(Utilización de framework de desarrollo de Código Abierto en el mundo empresarial)

Esther Casas Rescalvo  
Gerente Arquitectura Mundivia  
 [esther.casas@atosorigin.com](mailto:esther.casas@atosorigin.com)

# Indice



- » **¿Qué es un framework de desarrollo?**
- » **Directrices estratégicas de un framework de desarrollo**
- » **Estructura del framework de desarrollo.**
- » **Arquitectura.**
- » **Beneficios.**
- » **¿Cómo conseguir estos beneficios?**
- » **Ejemplos de uso en el mundo empresarial.**

## ¿Qué es un framework de desarrollo?



- » **Nuestra visión:**
  - » Un **Framework** (“**Marco de Trabajo**” o “**Marco de Desarrollo**”) es un **conjunto de librerías y componentes** de probada solvencia, junto con una **documentación y metodología de uso**, que permite diseñar, construir e implantar **aplicaciones corporativas de forma más uniforme, rápida, y con mayor calidad.**

## ¿Qué es un Framework de desarrollo?



- » Todo Framework consta de una serie de **elementos básicos**:
  - » **Artefactos software: componentes, herramientas y plataformas**
  - » **Documentación**
  - » **Material metodológico**
  - » **Material formativo y de difusión**
  - » **Aplicaciones de ejemplo**
  - » **Entornos de referencia (desarrollo y ejecución)**
  - » **Sistema de soporte**

## Directrices estratégicas de un Framework en un entorno corporativo.



- » **Estar alineado** con las directrices metodológicas y de estrategia tecnológica de la organización. Constituir un **factor de coordinación**.
- » Contar con **mecanismos de integración con los sistemas de negocio actuales** (Sistemas de Legado) y futuros con los que quede encuadrada la aplicación.
- » Enfocado a la **calidad de código**: reducir los costes de la “No-Calidad” mediante el establecimiento de una política de **Reuso Estratégico**.
- » Estar basada en **estándares abiertos y de amplia difusión** como los definidos en la plataforma J2EE y productos de código abierto asociados a ella, que eviten dependencias innecesarias de un proveedor concreto.
- » Ser **eminente práctica**, contrastando los diversos componentes con su utilización en el desarrollo real de aplicaciones.

## Directrices estratégicas de un Framework en un entorno corporativo.



- » Ser compatible con los **estándares metodológicos** imperantes en el desarrollo de aplicaciones reconocidos por la industria (**RUP, EUP, Métrica3...**)
- » **Simplificar el ciclo de vida de la explotación** de aplicaciones, proporcionando componentes que faciliten a los equipos de desarrollo la construcción de aplicaciones fácilmente explotables.
- » **Facilitar la construcción de aplicaciones seguras**, que sigan las directrices establecidas en los requisitos de seguridad de la organización.
- » Contar con los **mecanismos de soporte necesarios** para dar un alto nivel de apoyo a los equipos de desarrollo que lo utilicen.
- » Tener medios que faciliten **la formación y la difusión del conocimiento** asociado a estos componentes, y la captación de nuevos componentes y servicios de interés generados por las unidades de negocio.

# Estructura del Framework de desarrollo



	Descripción y principales tecnologías utilizadas
<b>Base</b>	<ul style="list-style-type: none"><li>» Es el "núcleo" de Framework. Está formada por los componentes que soportan el entorno de ejecución del framework y sigue un esquema de Arquitectura de N-capas basada en la plataforma JEE, con librerías que soportan cada una de las diferentes capas: Presentación-Control, Servicios-Negocio, Acceso a Datos-EIS, Integración, Seguridad, ...</li><li>» <b>Tecnologías:</b> <i>JSF, Struts, Spring, Hibernate, Ibatis, Spring Security, Axis2, ...</i></li></ul>
<b>Lib</b>	<ul style="list-style-type: none"><li>» Está conformada por un amplio conjunto de librerías y componentes utilitarios, con funciones bien delimitadas y concretas: comunicaciones, tratamiento ficheros, informes, etc. A diferencia del espacio Base, no son elementos que formen parte del núcleo general de un aplicativo, sino que realizan una actividad concreta.</li><li>» <b>Tecnologías:</b> <i>Muy variadas y específicas para cada funcionalidad. Ejemplo: Jasper Report, JavaMail,..</i></li></ul>
<b>Explo</b>	<ul style="list-style-type: none"><li>» Está compuesta por una serie de componentes orientados a facilitar la explotación de aplicaciones basadas en el Framework. Es el nexo de interacción con los servicios generales de Gestión y Explotación: registro de eventos, monitorización, resolución de problemas, ...</li><li>» <b>Tecnologías:</b> <i>Commons Logging, Log4J, JMX, Quartz, ANT, ...</i></li></ul>
<b>Infra</b>	<ul style="list-style-type: none"><li>» Define una serie de servicios de referencia, centrados en la infraestructura (máquinas virtuales, servidor de aplicaciones, servidor web, servidores de BBDD) sobre los que se ejecutan los aplicativos basados en el Framework. Es una capa adaptable a las necesidades de distintos clientes.</li><li>» <b>Tecnologías:</b> <i>Linux, Tomcat, JBoss, LDAP, varias RDBMS...</i></li></ul>

# Estructura del Framework de desarrollo



	Descripción y principales soluciones utilizadas
<b>Dev</b>	<ul style="list-style-type: none"><li>» Define el entorno de desarrollo básico (IDE) para la construcción de aplicaciones basadas en el Framework. Consta de las herramientas necesarias para dar soporte al ciclo de vida de desarrollo, en especial los aspectos de construcción y pruebas.</li><li>» <b>Tecnologías:</b> <i>Eclipse, NetBeans, JBossTools, ANT, JUnit, Hudson, Subversion, ...</i></li></ul>
<b>Gen</b>	<ul style="list-style-type: none"><li>» Construido sobre Dev, su función es acortar tiempos de desarrollo mediante procesos de generación de código en función de modelos de alto nivel, creando código que utilice los diversos servicios del resto de espacios, especialmente del espacio Base.</li><li>» <b>Tecnologías:</b> <i>Generadores de código propios.</i></li></ul>
<b>Forge</b>	<ul style="list-style-type: none"><li>» Este espacio define un canal web encaminado a proporcionar un servicio de soporte de alto nivel a los usuarios del Framework. Sirve para difundir el conocimiento y prestar soporte para la utilización correcta de los elementos del framework, así como obtener "feedback" sobre su utilización</li><li>» <b>Tecnologías:</b> <i>Wiki, Gestor de Incidencias, Foros, Descargas, Listas Correo, etc.</i></li></ul>
<b>Ejemplos</b>	<ul style="list-style-type: none"><li>» Espacio destinado a las aplicaciones ejemplo, desarrolladas con la arquitectura definida en el framework y siguiendo los procesos de generación de código, contemplando una funcionalidad limitada, pero lo suficientemente complejo para validar correctamente todos los elementos del Framework.</li></ul>

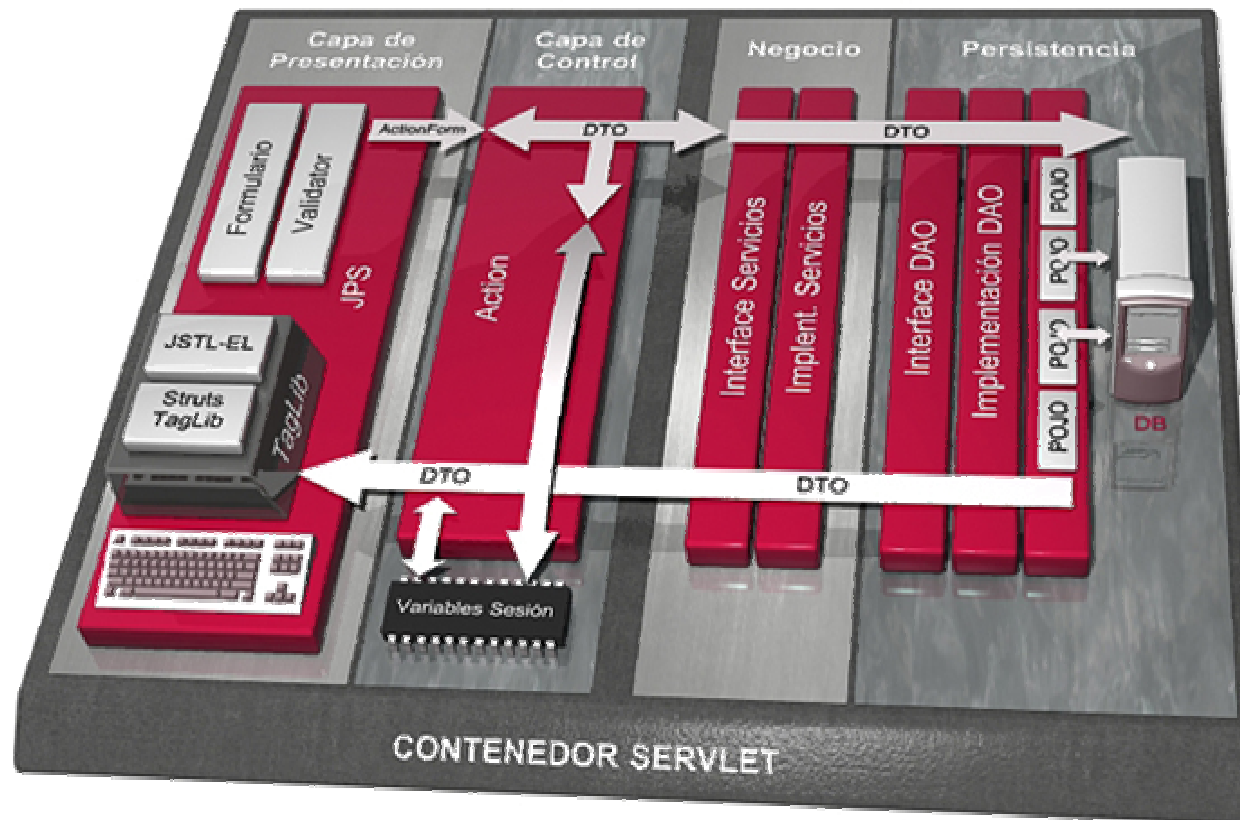
# Núcleo del Framework: Base



- » **Base** constituye el **núcleo central de la Arquitectura**, es el conjunto de paquetes y componentes principales usados para la construcción de aplicaciones.
- » Debe ser **escalable funcionalmente**: desde aplicaciones pequeñas y sencillas, a sistemas complejos, distribuidos y con criticidad.
- » Estructurada siguiendo un **Esquema de Capas**: Presentación-Control, Negocio-Servicios, Persistencia-Datos
- » Incluye los **esquemas básicos de integración** (acceso a EAI/ESB, Web Services, Sistemas de Legado) y los aspectos **transversales de seguridad** (Autorización, Autenticación, Registro).
- » Permite **varias alternativas de Arquitectura de Aplicaciones**, según necesidades.
  - » **Ligeras**: basadas en objetos java tradicionales y servidores de servlets.
  - » **Pesadas**: con uso de EJB y servidores de aplicaciones.

Capa	Tecnología
Presentación	» Struts » JSF
Negocio-Servicios	» Spring
Acceso a Datos	» Hibernate » Ibatis
Integración	» Axis » JMS » JCA
Seguridad	» ACEGI (Spring Security)

# Arquitectura propuesta: Aplicación básica



# Beneficios de un Framework de desarrollo



- » Reducción en el **coste de licencias**.
  - » La gran mayoría de los componentes son de Código Abierto.
  - » En escenario muy críticos y aún siendo componentes de Código Abierto la mayoría tienen posibilidad de soporte por terceros proveedores.
- » Reducción en los **tiempos de esfuerzo de desarrollo**.
  - » Ofreciendo generadores de código y plantillas arquitectónicas, desarrolladas con las últimas versiones de componentes y recogiendo todas las mejores prácticas de los mismos.
  - » Evitando el síndrome del “papel en blanco”: por dónde empiezo y qué utilizo.
  - » Conocimiento adquirido reutilizable en todos los proyectos.
  - » Simplificando los procesos de mantenimiento a largo plazo.
- » Reducción de los **tiempos de esfuerzo en operación**.
  - » Infraestructura uniforme que permite metodologías de operación estándares.
  - » Minimizar discrepancias entre equipos de desarrollo y de explotación.
  - » Simplificación en la resolución de problemas en producción.

## Beneficios de un Framework de desarrollo



- » La estandarización de un Framework de desarrollo **incrementa la calidad y reduce los costes en todo el ciclo de vida** del software corporativo:
  - » Permite el **diseño homogéneo** de las soluciones.
  - » Establece una **metodología de desarrollo**.
  - » Facilita y unifica la **formación de equipos** de trabajo.
  - » Estandariza los **servicios de contratistas externos**.
  - » Proporciona los mecanismos de **integración de aplicaciones**.
  - » Simplifica el **soporte a proyectos**.
  - » Agiliza la **puesta en explotación y la operación** de Aplicativos.
  - » Aporta mecanismos para la **resolución de incidencias**.
  - » Favorece la adaptación controlada a la **evolución tecnológica**.
  - » Posibilita el **alineamiento con el negocio**.

## Como conseguir estos beneficios



- » No considerar el framework de desarrollo una vez definido, como un mero “**empaquetamiento de librerías**” fijas e inmutables en el tiempo. El framework debe estar vivo.
- » Debe existir un **proceso de evolución definidos y controlado (“Hoja de Ruta”)**, acorde a:
  - » La evolución tecnológica del mercado.
  - » La estrategia y política de la organización.
  - » Las necesidades concretas de los proyectos, que no estén incorporadas en el Framework
- » La **formación y difusión** son elementos críticos.
  - » Catálogo de cursos fijos: Curso de desarrollo en el framework, Curso de diseño, Cursos concretos de tecnología.
  - » Divulgacion tecnológica de las evoluciones.
  - » No sólo para desarrolladores: debe existir material de difusión para perfiles no técnicos (directivos, jefes de proyecto)

## Como conseguir estos beneficios



- » Las organizaciones requieren **servicios de soporte de alta calidad** para sus aplicaciones críticas de negocio:
  - » Asesoramiento a los proyectos en sus fases iniciales para la definición de la Arquitectura.
  - » Construcción de pilotos y prototipos para la resolución de necesidades específicas de los proyectos.
  - » Incorporación de nuevos componentes para necesidades no cubiertas por el framework.
  - » Soporte a los equipos de desarrollo durante la construcción y el mantenimiento: resolución de dudas, corrección de errores, validación de código, ...
  - » Apoyo a los equipos de explotación durante la implantación y el despliegue.
  - » Aseguramiento de la calidad: auditorías de código, profiling, pruebas de rendimiento.
  - » Recogida de “lecciones aprendidas”, para enriquece el framework.

# Ejemplos de uso real: ATOS



**Filtro**

Usuario:

Nombre:  1º Apell.:  2º Apell.:

Grupo:  Horario:  Perfil:

Orden:  Desde:  Hasta:

html  pdf  xls  xml

Usuario	Grupo	Horario	Tipo	Fecha
pablo	IT	Mañana y Tarde	E	01-ene-1904 1:01:00
pablo	IT	Mañana y Tarde	E	01-ene-1905 1:01:00
pablo	IT	Mañana y Tarde	E	30-oct-1908 12:51:00
pablo	IT	Mañana y Tarde	E	31-oct-1909 13:59:00
pablo	IT	Mañana y Tarde	E	31-oct-1912 14:20:00

[Página anterior](#) [Página siguiente](#)

# Ejemplos de uso real: Endesa



# Ejemplos de uso real: Gobierno de Cantabria



## Conclusiones



- » **Un Framework de Desarrollo no debe quedarse sólo en un mero “empaquetamiento de código”**
- » **Las tareas de soporte, difusión, seguimiento y evolución son críticas para su correcto aprovechamiento por los proyectos**
- » **No debe “reinventarse la rueda”: existen elementos base de Código Abierto de altísima calidad**
- » **Solo debe desarrollarse a medida lo que sea diferencial para el proyecto o la organización**

**Gracias por su atención  
¿Preguntas?**