

# ¿Qué ofrece Autentia Real Business Solutions S.L?

Somos su empresa de **Soporte a Desarrollo Informático**.  
 Ese apoyo que siempre quiso tener...

## 1. Desarrollo de componentes y proyectos a medida



## 2. Auditoría de código y recomendaciones de mejora

## 3. Arranque de proyectos basados en nuevas tecnologías

1. Definición de frameworks corporativos.
2. Transferencia de conocimiento de nuevas arquitecturas.
3. Soporte al arranque de proyectos.
4. Auditoría preventiva periódica de calidad.
5. Revisión previa a la certificación de proyectos.
6. Extensión de capacidad de equipos de calidad.
7. Identificación de problemas en producción.



## 4. Cursos de formación (impartidos por desarrolladores en activo)

Spring MVC, JSF-PrimeFaces /RichFaces,  
 HTML5, CSS3, JavaScript-jQuery

Gestor portales (Liferay)  
 Gestor de contenidos (Alfresco)  
 Aplicaciones híbridas

Tareas programadas (Quartz)  
 Gestor documental (Alfresco)  
 Inversión de control (Spring)

Control de autenticación y  
 acceso (Spring Security)  
 UDDI  
 Web Services  
 Rest Services  
 Social SSO  
 SSO (Cas)

JPA-Hibernate, MyBatis  
 Motor de búsqueda empresarial (Solr)  
 ETL (Talend)

Dirección de Proyectos Informáticos.  
 Metodologías ágiles  
 Patrones de diseño  
 TDD

BPM (jBPM o Bonita)  
 Generación de informes (JasperReport)  
 ESB (Open ESB)

## Últimas Noticias

- » Entrevista a Roberto Canales
- » Autentia en JavaHispano
- » Liberada TNTConcept 0.16.1
- » Cuarta charla Autentia + Agile Spain: Introducción a Scrum
- » Historia de la Informática. Capítulo 46 - 1963
- » ¡Adictos Renovado!
- » Una historia de guerra Ágil: SCRUM Y XP DESDE LAS TRINCHERAS, Cómo hacemos Scrum
- » Comentarios sobre Wikinomics de Don Tapscott
- » Gestión de Repositorios Maven

## +Noticias Destacadas

- » Liberada TNTConcept 0.16.1
- » Cuarta charla Autentia + Agile Spain: Introducción a Scrum
- » Entrevista a Roberto Canales
- » Si se pregunta ¿Qué ofrece este Web?

## +Comentarios Cómic

## +Enlaces

## Catálogo de servicios Autentia (PDF 6,2MB)



En formato comic...



- Web
  - [www.adictosaltrabajo.com](http://www.adictosaltrabajo.com)
- Buscar

## Tutorial desarrollado por



### Alejandro Pérez García

Alejandro es socio fundador de Autentia y nuestro experto en J2EE, Linux y optimización de aplicaciones empresariales.

Ingeniero en Informática y ScrumMaster

Si te gusta lo que ves, puedes contratarle para impartir **cursos presenciales** en tu empresa o para **ayudarte en proyectos** (Madrid). Puedes encontrarme en [Autentia](#)

## Catálogo de servicios de Autentia

Descargar (6,2 MB)

Descargar en versión comic (17 MB)

[AdictosAlTrabajo.com](#) es el Web de difusión de conocimiento de [Autentia](#).



[Catálogo de cursos](#)

Descargar este documento en formato PDF: [agileDraw.pdf](#)

Fecha de creación del tutorial: 2009-04-26

## AgileDraw una técnica rápida de modelado

Creación: 26-04-2009

## Índice de contenidos

1. Introducción
2. Entorno
3. ¿Qué significan los símbolos de AgileDraw?
4. Ejemplos
5. ¿Puedo usar más símbolos, a parte de los básicos vistos antes?
6. Buenas prácticas con AgileDraw
7. Conclusiones
8. Sobre el autor

## 1. Introducción

En este tutorial vamos a ver una introducción a [AgileDraw](#). AgileDraw, más que una herramienta, se trata de una técnica de modelado rápido. Es decir, AgileDraw persigue los principios de comunicación eficaz frente a herramientas, y comprensión frente a procesos. De esta forma AgileDraw nos proporciona cuatro elementos básicos para pintar nuestros modelos: círculos, cajas, líneas y texto. Con estos elementos nos centraremos más en el *qué* (lo que pretendemos comunicar), en vez de hacer hincapié en el *cómo* (que notación o herramienta utilizamos).

En la propia página de AgileDraw nos recomiendan que lo usemos cuando:

- Prefiramos simplicidad frente a complejidad.
- Documentación suficientemente buena frente a artefactos precisos.
- Compartir conocimiento frente a herramientas.
- Algo práctico frente a abstracciones idealizadas.
- Y por supuesto, conceptos frente a notación compleja.

Esto no quiere decir que AgileDraw pretenda sustituir el puesto al UML. Todo lo contrario, AgileDraw lo complementa perfectamente, tal como podemos ver en esta imagen sacada de su web:

## Últimos tutoriales

2009-04-26  
[AgileDraw](#)

2009-04-24  
[Spring AOP: Cacheando aplicaciones usando anotaciones y aspectos con Aspectj](#)

2009-04-20  
[Modelos de conocimiento con CmapTools](#)

2009-04-16  
[Informes Crosstab con iReport](#)

2009-04-16  
[Registro de un fichero de datos personales con el formulario NOTA](#)

2009-04-15  
[Estadísticas de www.adictosaltrabajo.com Abril 2009](#)

2009-04-15  
[Iniciación a OSWorkflow con Spring](#)

2009-04-14  
[Tests de Selenium con librerías de componentes JSF: Apache Tomahawk.](#)

2009-04-13  
[JTAPI. El API de Telefonía para Java](#)

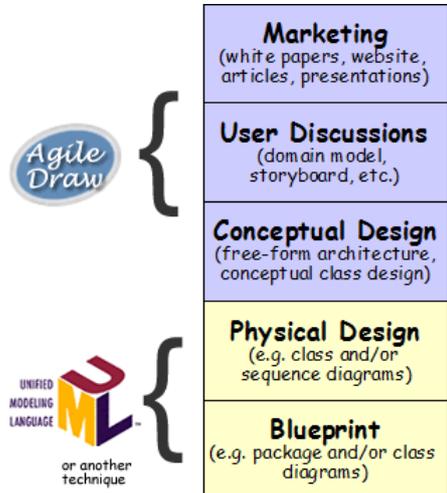
2009-04-13  
[Registro de Web Services con Apache JUDDI. Configuración y ejemplo](#)

2009-04-13  
[Cómo hacer UML con Eclipse y el plugin UML2](#)

2009-04-09  
[Spring WS: Servicios Web a través del correo electrónico](#)

2009-04-02  
[Creación de cursos con Moodle](#)

2009-03-31  
[Integrar Liferay Portal 5.2.1 con](#)



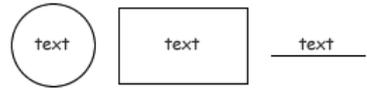
## 2. Entorno

El tutorial está escrito usando el siguiente entorno:

- Hardware: Portátil MacBook Pro 17' (2.93 GHz Intel Core 2 Duo, 4GB DDR3 SDRAM, 128GB Solid State Drive).
- NVIDIA GeForce 9400M + 9600M GT with 512MB
- Sistema Operativo: Mac OS X Leopard 10.5.6

## 3. ¿Qué significan los símbolos de AgileDraw?

En los sistemas informáticos, la información se identifica de dos maneras, en movimiento o almacenada. Basándose en esta premisa, AgileDraw utiliza los siguientes componentes:

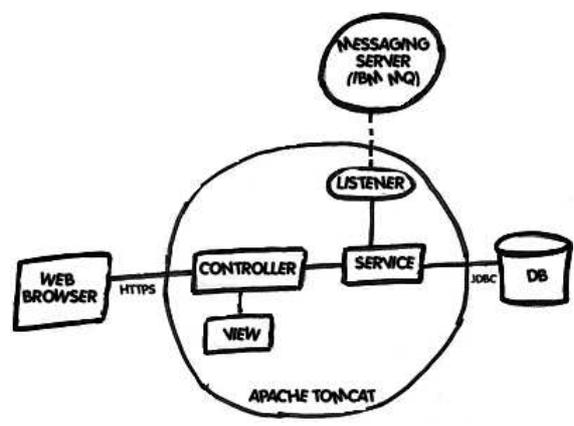


- Puntos:
  - **Círculos** que representan movimiento, flexibilidad. Por ejemplo "Listeners" como servidores de aplicaciones servidores de bases de datos, servidores de mensajes, ...
  - **Cajas** que representan cosas estáticas, faltas de movimiento, rígidas. Por ejemplo hardware, clases, entidades o tablas, interfaz de usuario, o en general cualquier cosa.
- Conectores: Las **líneas** representan la conexión entre los puntos. Las líneas pueden tener puntas de flecha para representar dirección. Podremos tener líneas de dos tipos:
  - Solidas que representan concreción, sincronismo, asociaciones fuertes. Por ejemplo comunicación síncrona, relaciones entre entidades, ...
  - Punteadas que representan abstracción, asíncrono, asociaciones débiles. Por ejemplo comunicaciones asíncronas, extensión de clases abstractas, ...
- **Texto**: Usaremos texto para describir los distintos puntos y conectores.

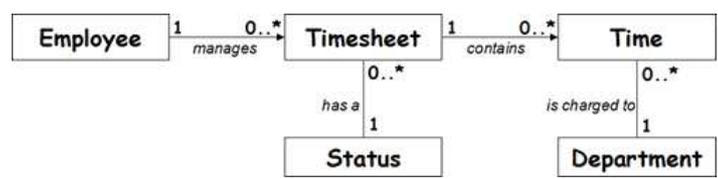
## 4. Ejemplos

Veamos algunos ejemplos sacados de la propia página de AgileDraw.

Una arquitectura a nivel general:



Un modelo de dominio:



Un diagrama de clases conceptual:

Pentaho BI 2.0.0 sobre MySQL 5.1

2009-03-31  
Spring WS: Construcción de Clientes de Servicios Web con Spring

2009-03-30  
Administración de sitios Moodle

2009-03-29  
Empaquetamiento de aplicaciones de escritorio (standalone) con Maven

2009-03-27  
Primeros pasos con Moodle

2009-03-26  
Introducción a JSF Java

2009-03-25  
A1 Website Analyzer

2009-03-24  
Cómo ver el correo de Gmail sin conexión a Internet

2009-03-20  
JasperReports Maven Plugin

2009-03-16  
Creación de contenidos SCORM: eXe

2009-03-15  
Spring WS: Creación de Servicios Web con Spring

2009-03-13  
Instalación Alfresco (Labs)

2009-02-26  
Maven JXR Plugin: publica el código fuente en el site

2009-03-15  
Generación de XML Schema (XSD) y DTD a partir de documentos XML

2009-03-04  
Persistencia con Spring

2009-02-26  
Vistas materializadas

2009-02-03  
Instalación de MySQL 5.1 en Windows

2009-03-03  
Instalación de Java Virtual Machine

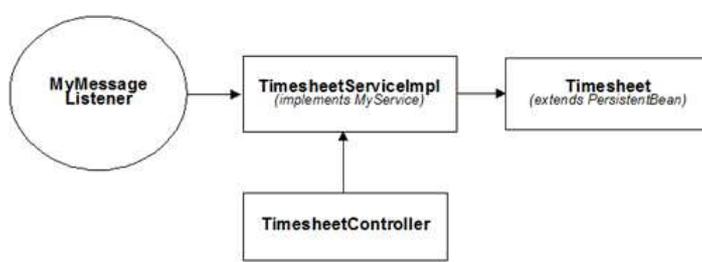
2009-03-03  
Primeros Pasos con Liferay 5.2.1

2009-02-27  
Edición de video MPEG2

2009-02-26  
Introducción teórica a XPath

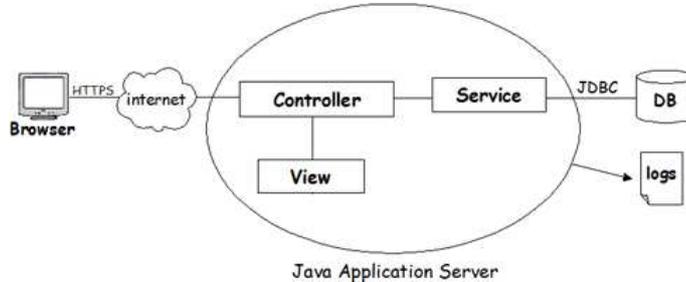
2009-02-26  
Integración Selenium / Maven 2 / Surefire / Cargo / Tomcat 6

2009-02-24  
Selenium Remote Control



## 5. ¿Puedo usar más símbolos, a parte de los básicos vistos antes?

Claramente SI. AgileDraw no nos limita los símbolos que debemos usar en nuestros modelos, de hecho nos anima a usar cualquier cosa que sea significativa y comúnmente aceptadas por todos los interesados. Veamos un ejemplo:



Podemos observar en el ejemplo como el PC del cliente se representa con una especie de monitor, Internet se representa con la típica "nube", la base de datos con un "tambor", y un fichero de log con el típico rectángulo con una esquina "doblada".

AgileDraw también nos propone el uso de colores en nuestros modelos. Por defecto son los siguientes códigos de color, aunque como siempre podemos usar cualquier convención que queramos, siempre y cuando sea comúnmente aceptada por todos los interesados.

Rojo	Acción, confianza, energía, poder, peligro, alerta.
Marrón	Orden, convención, conservador, estable.
Naranja	Resistencia, energía, creatividad, alegría, determinación, éxito, construcción.
Oro	Salud, prosperidad.
Amarillo	Sabiduría, felicidad, energía intelectual.
Verde	Vida, naturaleza, bienestar.
Azul	Verdad, distancia.
Púrpura	Realeza, misterio, imaginación.
Blanco	Pureza, limpieza.
Negro	Muerte, estabilidad.
Gris	Seguridad, madurez, neutralidad.

También podemos usar sombras, degradados, fotos, bocadillos, distintos tipos de fuentes, ... y en general cualquier cosa que haga más comprensible nuestro modelo.

## 6. Buenas prácticas con AgileDraw

En la página de AgileDraw podemos encontrar las siguientes buenas prácticas:

- Manténlo sencillo, limitando el número de distintos elementos en el mismo diagrama.
- Usa texto para describir las conexiones y los puntos. Puedes usar fuentes más pequeñas para elementos menos relevantes del diagrama.
- Los diagramas deberían ser auto explicativos. Para no usar documentación adicional, puedes usar notas o leyendas en el propio diagrama.
- Otras buenas prácticas cogidas de *Comparison of Diagramming Methods*, by Keith Stiern:
  - El contexto del diagrama debe ser claro.
  - Cada diagrama debe presentar una vista del sistema que sea independiente del resto de vistas.
  - El contexto del diagrama se debe comprender claramente por su título, un texto, una referencia a otro contexto, o cualquier combinación de estos. En definitiva, el diagrama tiene que dejar claro que es lo que está representando.
  - Debe tenerse en cuenta la fecha y la versión del documento. De esta manera no haremos cosas basadas en una versión antigua, ya que rápidamente se podrá identificar con que versión del documento están trabajando todos los interesados.
  - Si hay más de un tipo de diagrama para un sistema, cada tipo de diagrama debe ser Ortogonal al resto de tipos. Esto quiere decir que cada tipo de diagrama debe presentar aspectos del sistema desde perspectivas que son totalmente independientes

2009-02-22  
Integración de Groovy, JRuby y BeanShell con Spring 2

2009-02-18  
Instalación de Pentaho BI Suite Community Edition 1.7.0

2009-02-18  
Replicar Web PHP en máquina local

2009-02-16  
Selenium Core : El motor de Selenium.

2009-02-16  
Integración de JasperReports con PHP

2009-02-09  
EJB 3.0 y pruebas unitarias con Maven, JUnit 4 y Embedded JBoss sobre Java 6

2009-02-09  
Web Service Security

2009-02-09  
Manual Avanzado de Firebug

2009-01-29  
Ejemplo con Mockito

2009-01-29  
Uso de Mock objects en pruebas con Mockito

2009-01-29  
StrutsTestCase

2009-01-28  
Eventos en Hibernate (parte III)

2009-01-28  
Eventos en Hibernate (parte II)

2009-01-27  
Eventos en Hibernate (parte I)

## Últimas ofertas de empleo

2009-04-24  
Comercial - Ventas - VALENCIA.

2009-03-26  
Comercial - Ventas - ALMERIA.

2009-03-12  
Comercial - Ventas - VALENCIA.

2009-03-12  
Comercial - Ventas - SEVILLA.

2009-02-21  
Otras - Estética/Peluquería - MADRID.

## Anuncios Google

## 7. Conclusiones

Como podéis ver AgileDraw es muy sencillo, ya que no tiene una notación extensa o un conjunto de reglas estrictas. Simplemente pretende mejorar la comunicación entre equipos, usando para ello herramientas sencillas (tan solo tenemos 4 elementos básicos: círculo, cuadrado, línea y texto).

Este tipo de técnicas son más que recomendables ya que ayudan a mejorar la comprensión del sistema y mejorar el entendimiento entre los distintos miembros del equipo.

Además AgileDraw no necesita de una potente aplicación de modelado. Lo más potente y flexible para estas cosas suele ser un simple papel y un lápiz, una pizarra, o similar.

## 8. Sobre el autor

Alejandro Pérez García, Ingeniero en Informática (especialidad de Ingeniería del Software) y ScrumMaster

Socio fundador de Autentia (Formación, Consultoría, Desarrollo de sistemas transaccionales)

<mailto:alejandrog@autentia.com>

Autentia Real Business Solutions S.L. - "Soporte a Desarrollo"

<http://www.autentia.com>

### ¿Qué te ha parecido el tutorial? Déjanos saber tu opinión y ivota!

Muy malo   Malo   Regular   Bueno   Muy bueno



Votar

### Anímate y coméntanos lo que pienses sobre este tutorial

Puedes opinar o comentar cualquier sugerencia que quieras comunicarnos sobre este tutorial; con tu ayuda, podemos ofrecerte un mejor servicio.

Nombre:

E-Mail:

Comentario:

Enviar comentario

[Texto Legal y condiciones de uso](#)

- Puedes inscribirte en nuestro servicio de notificaciones [haciendo clic aquí](#).
- Puedes firmar en nuestro libro de visitas [haciendo clic aquí](#).
- Puedes asociarte al grupo AdictosAlTrabajo en XING [haciendo clic aquí](#).
- Añadir a favoritos Technorati. 



Esta obra está licenciada bajo [licencia Creative Commons de Reconocimiento-No comercial-Sin obras derivadas 2.5](#)

## Recuerda

Autentia te regala la mayoría del conocimiento aquí compartido ([Ver todos los tutoriales](#)). Somos expertos en: J2EE, Struts, JSF, C++, OOP, UML, UP, Patrones de diseño ... y muchas otras cosas.

**¿Nos vas a tener en cuenta cuando necesites consultoría o formación en tu empresa?, ¿Vas a ser tan generoso con nosotros como lo tratamos de ser con vosotros?**

**Somos pocos, somos buenos, estamos motivados y nos gusta lo que hacemos ...**

Autentia = Soporte a Desarrollo & Formación.

[info@autentia.com](mailto:info@autentia.com)

Gestión de contenidos

## Tutoriales recomendados

Nombre	Resumen	Fecha	Visitas	Valoración	Votos	Pdf
<a href="#">Modelos de conocimiento con CmapTools</a>	Roberto nos enseña en este tutorial el uso de la herramienta CmapTools	2009-04-20	470	Muy bueno	10	
<a href="#">Iniciación a OSWorkflow con Spring</a>	En este tutorial vamos a presentar uno de los motores de workflow, "OSWorkflow" y su integración con Spring.	2009-04-15	381	Muy bueno	2	
<a href="#">Cómo hacer UML con Eclipse y el plugin UML2</a>	En este tutorial vamos a ver como podemos hacer modelos de UML con el plugin de Eclipse UML2.	2009-04-13	910	Bueno	8	
<a href="#">Persistencia con Spring</a>	En el siguiente tutorial vamos a ver algunas de las aportaciones que nos ofrece Spring para mejorar la capa de persistencia de nuestras aplicaciones	2009-03-04	1498	Muy bueno	10	
<a href="#">Modelado BPMN con Bizagi Modeler</a>	En ésta ocasión, nuestro compañero Roberto os enseña una herramienta muy útil de modelado BPMN, llamada Bizagi Modeler	2008-12-10	1827	Muy bueno	40	
<a href="#">ETL con Talend</a>	Nuestro compañero Roberto, en otro más de sus numerosos tutoriales nos enseña el uso de Talend, una herramienta muy útil de integración de datos (ETL)	2008-12-03	856	Bueno	9	
<a href="#">Introducción a JTrac</a>	JTrac es una aplicación web de gestión y seguimiento de incidencias similar a Bugzilla, Eventum, Mantis, etc. Es una herramienta "open source" escrita en Java bajo licencia V2.0. de Apache	2008-11-17	1708	Muy bueno	6	
<a href="#">Planificación de tareas en Java mediante Quartz</a>	Introducción a Quartz, un excelente Framework para planificar tareas Java.	2008-10-31	2898	Bueno	6	
<a href="#">maven-license-plugin: cómo gestionar la licencia de nuestros ficheros fuentes</a>	maven-license-plugin es un plugin que se puede enganchar al ciclo de vida de Maven para comprobar y/o actualizar el texto de la licencia en todos nuestros ficheros fuente	2008-10-17	856	Muy bueno	1	
<a href="#">Introducción a UMLet</a>	En muchos de los proyectos, el UML se utiliza únicamente a efectos documentales y, por la falta de tiempo raramente se mantienen. En este tutorial vamos a probar una opción llamada UMLet, que os ayudara a crear diagramas UML	2008-09-30	1532	-	-	

### Nota:

Los tutoriales mostrados en este Web tienen como objetivo la difusión del conocimiento. Los contenidos y comentarios de los tutoriales son responsabilidad de sus respectivos autores. En algún caso se puede hacer referencia a marcas o nombres cuya propiedad y derechos es de sus respectivos dueños. Si algún afectado desea que incorporemos alguna reseña específica, no tiene más que solicitarlo. Si alguien encuentra algún problema con la información publicada en este Web, rogamos que informe al administrador [rcanales@adictosaltrabajo.com](mailto:rcanales@adictosaltrabajo.com) para su resolución.