

# 1 APÉNDICE CON LOS DIAGRAMAS DE CLASES DEL PFC

## 1.1 JOSEJAMILENA::PFC::ANALIZADOR

### 1.1.1 ANALIZADOR::ABOUTBOX

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:10. Modificado el 16/07/2009 11:32:10.

Autor: Jose Antonio Jamilena Daza

Esta clase muestra la ventana de “Acerca de...” de la interfaz gráfica.

#### 1.1.1.1 Atributos de analizador::AboutBox

Atributos	Tipo	Notas
agradecimientoScroll	privado : javax.swing.JScro llPane	

agradecimientoText	privado : javax.swing.JText Area	
alumno	privado : javax.swing.JLab el	
closeButton	privado : javax.swing.JButt on	
profesor	privado : javax.swing.JLab el	

#### 1.1.1.2 Métodos de analizador::AboutBox

Método	Tipo	Notas
AboutBox (java.awt.Frame)	público:	
closeAboutBox ()	público: void	
initComponents ()	privado: void	

### 1.1.2 ANALIZADOR::APP

Tipo: Clase Pública

Implementa: SingleFrameApplication.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:10. Modificado el 16/07/2009 11:32:10.

Autor: Jose Antonio Jamilena Daza

La clase principal de la aplicación.

#### 1.1.2.1 Atributos de analizador::App

Atributos	Tipo	Notas
conn	público y de clase: Connection	Conexión JDBC.
ficheroEstadisticas	privado static : String	Fichero de BD de estadísticas.

#### 1.1.2.2 Métodos de analizador::App

Método	Tipo	Notas
startup ()	protegido: void	

configureWindow (java.awt.Window)	protegido: void	
getApplication ()	público y de clase: App	
getFicheroEstadisticas ()	público: String	
setFicheroEstadisticas (String)	público: void	
iniciarConexion ()	público y de clase: boolean	
main (String[])	público y de clase: void	

### *1.1.3 ANALIZADOR::FILECHOOSER*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:10. Modificado el 16/07/2009 11:32:10.

Autor: Jose Antonio Jamilena Daza

Esta clase implementa el explorador de ficheros para apertura de un fichero de estadísticas.

#### 1.1.3.1 Atributos de analizador::FileChooser

Atributos	Tipo	Notas
fileChooser	privado : javax.swing.JFileChooser	

#### 1.1.3.2 Métodos de analizador::FileChooser

Método	Tipo	Notas
FileChooser (java.awt.Frame, boolean)	público:	
initComponents ()	privado: void	
fileChooserActionPerformed (java.awt.event.ActionEvent)	privado: void	

#### 1.1.4 ANALIZADOR::*FILECHOOSERCSV*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:10. Modificado el 16/07/2009 11:32:10.

Autor: Jose Antonio Jamilena Daza

Esta clase implementa el explorador de ficheros para la exportación como fichero de valores separado por comas del fichero de estadísticas actual.

##### 1.1.4.1 Atributos de analizador::*FileChooserCSV*

Atributos	Tipo	Notas
fileChooser	privado : javax.swing.JFileChooser	

##### 1.1.4.2 Métodos de analizador::*FileChooserCSV*

Método	Tipo	Notas
FileChooserCSV (java.awt.Frame, boolean)	público:	

initComponents ()	privado: void	
fileChooserActionPerformed (java.awt.event.ActionEvent)	privado: void	
main (String[])	público y de clase: void	

### *1.1.5 ANALIZADOR::GRAFICOPORCLIENTE*

Tipo: Clase Pública

Implementa: JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:11. Modificado el 16/07/2009 11:32:11.

Autor: Jose Antonio Jamilena Daza

Genera gráficas por cliente.

#### **1.1.5.1 Métodos de analizador::GraficoPorCliente**

Método	Tipo	Notas
--------	------	-------

GraficoPorCliente (String, String)	público:	
---------------------------------------	----------	--

#### *1.1.6 ANALIZADOR::GRAFICOPORSCRIPT*

Tipo: Clase Pública

Implementa: JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:11. Modificado el 16/07/2009 11:32:11.

Autor: Jose Antonio Jamilena Daza

Genera gráficas por script.

##### **1.1.6.1 Métodos de analizador::GraficoPorScript**

Método	Tipo	Notas
GraficoPorScript (String, String)	público:	



### *1.1.7 ANALIZADOR::GRAFICOPORSGBD*

Tipo: Clase Pública

Implementa: JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:11. Modificado el 16/07/2009 11:32:11.

Autor: Jose Antonio Jamilena Daza

Genera gráficas por sistema gestor de bases de datos.

#### **1.1.7.1 Métodos de analizador::GraficoPorSGBD**

Método	Tipo	Notas
GraficoPorSGBD (String, String)	público:	

### *1.1.8 ANALIZADOR::MSGBOX*

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:11. Modificado el 16/07/2009 11:32:11.

Autor: Jose Antonio Jamilena Daza

Genera mensajes emergentes.

#### **1.1.8.1 Métodos de analizador::MsgBox**

Método	Tipo	Notas
MsgBox ()	privado:	
MsgBox (String, String, String)	público:	

#### *1.1.9 ANALIZADOR::NETBOX*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:11. Modificado el 16/07/2009 11:32:11.

Autor: Jose Antonio Jamilena Daza

Abre la ventana para pedir los datos de conexión a servicio por red, para poder obtener el fichero de estadísticas de dicho servicio.

#### 1.1.9.1 Atributos de analizador::NetBox

Atributos	Tipo	Notas
closeButton	privado : javax.swing.JBut ton	
hostLabel	privado : javax.swing.JLa bel	
hostText	privado : javax.swing.JTe xtField	
icono	privado : javax.swing.JLa bel	
jButton1	privado : javax.swing.JBut ton	
portLabel	privado : javax.swing.JLa bel	
portText	privado : javax.swing.JTe xtField	

#### 1.1.9.2 Métodos de analizador::NetBox

Método	Tipo	Notas
NetBox (java.awt.Frame)	público:	
initComponents ()	privado: void	
closeNetBox ()	público: void	
obtenerEstadisticas ()	público: void	

#### 1.1.10 ANALIZADOR::SELECCIONARCLIENTE

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:12. Modificado el 16/07/2009 11:32:12.

Autor: Jose Antonio Jamilena Daza

Selecciona un host cliente.

##### 1.1.10.1 Atributos de analizador::SeleccionarCliente

Atributos	Tipo	Notas
-----------	------	-------

lista	privado : List<String>	
icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	
siguienteButton	privado : javax.swing.JBut ton	

#### 1.1.10.2 Métodos de analizador::SeleccionarCliente

Método	Tipo	Notas
SeleccionarCliente (java.awt.Frame, List<String>)	público:	
siguienteSeleccionarScript Box ()	público: void	
initComponents ()	privado: void	

### *1.1.11 ANALIZADOR::SELECCIONARCLIENTEScript*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:12. Modificado el 16/07/2009 11:32:12.

Autor: Jose Antonio Jamilena Daza

Selecciona un script, habiendo sido previamente seleccionado un cliente

#### **1.1.11.1 Atributos de analizador::SeleccionarClienteScript**

Atributos	Tipo	Notas
lista	privado : List<String>	
hostCliente	privado : String	
cargarGrafica	privado : javax.swing.JBut ton	
closeButton	privado : javax.swing.JBut ton	

icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	

#### 1.1.11.2 Métodos de analizador::SeleccionarClienteScript

Método	Tipo	Notas
SeleccionarClienteScript (java.awt.Frame, List<String>, String)	público:	
closeScriptBox ()	público: void	
initComponents ()	privado: void	
cargarGrafica ()	público: void	

#### 1.1.12 ANALIZADOR::SELECCIONARSCRIPT

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:12. Modificado el 16/07/2009 11:32:12.

Autor: Jose Antonio Jamilena Daza

#### 1.1.12.1 Atributos de analizador::SeleccionarScript

Atributos	Tipo	Notas
lista	privado : List<String>	
icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	
siguienteButton	privado : javax.swing.JBut ton	

#### 1.1.12.2 Métodos de analizador::SeleccionarScript

Método	Tipo	Notas
SeleccionarScript (java.awt.Frame, List<String>)	público:	



siguienteSeleccionarScript Box ()	público: void	
initComponents ()	privado: void	

### *1.1.13 ANALIZADOR::SELECCIONARSCRIPTSGBD*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:12. Modificado el 16/07/2009 11:32:12.

Autor: Jose Antonio Jamilena Daza

Selecciona un sistema gestor de bases de datos, previamente seleccionando un script.

#### **1.1.13.1 Atributos de analizador::SeleccionarScriptSGBD**

Atributos	Tipo	Notas
lista	privado : List<String>	
sscript	privado :	

	String	
cargarGrafica	privado : javax.swing.JBut ton	
closeButton	privado : javax.swing.JBut ton	
icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	

#### 1.1.13.2 Métodos de analizador::SeleccionarScriptSGBD

Método	Tipo	Notas
SeleccionarScriptSGBD (java.awt.Frame, List<String>, String)	público:	
closeScriptBox ()	público: void	
initComponents ()	privado: void	
cargarGrafica ()	público: void	

#### *1.1.14 ANALIZADOR::SELECCIONARSGBD*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:12. Modificado el 16/07/2009 11:32:12.

Autor: Jose Antonio Jamilena Daza

Selecciona un sistema gestor de bases de datos.

##### **1.1.14.1 Atributos de analizador::SeleccionarSGBD**

Atributos	Tipo	Notas
lista	privado : List<String>	
icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	
siguienteButton	privado : javax.swing.JBut	

	ton	
--	-----	--

#### **1.1.14.2 Métodos de analizador::SeleccionarSGBD**

Método	Tipo	Notas
SeleccionarSGBD (java.awt.Frame, List<String>)	público:	
siguienteSeleccionarScript Box ()	público: void	
initComponents ()	privado: void	

#### *1.1.15 ANALIZADOR::SELECCIONARSGBDSCRIPT*

Tipo: Clase Pública

Implementa: javax.swing.JDialog.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:13. Modificado el 16/07/2009 11:32:13.

Autor: Jose Antonio Jamilena Daza

Selecciona un script, previamente habiendo seleccionado un sistema gestor de bases de datos.

#### 1.1.15.1 Atributos de analizador::SeleccionarSGBDScript

Atributos	Tipo	Notas
lista	privado : List<String>	
hostSGBD	privado : String	
cargarGrafica	privado : javax.swing.JBut ton	
closeButton	privado : javax.swing.JBut ton	
icono	privado : javax.swing.JLa bel	
listaScript	privado : javax.swing.JCo mboBox	

#### 1.1.15.2 Métodos de analizador::SeleccionarSGBDScript

Método	Tipo	Notas
--------	------	-------

SeleccionarSGBDScript (java.awt.Frame, List<String>, String)	público:	
closeSeleccionarScriptBox ()	público: void	
initComponents ()	privado: void	
cargarGrafica ()	público: void	

#### *1.1.16 ANALIZADOR::VIEW*

Tipo: Clase Pública

Implementa: FrameView.

Estado: Versión 1.0. Fase 1.0.

Paquete: analizador

Detalles: Creado el 16/07/2009 11:32:13. Modificado el 16/07/2009 11:32:13.

Autor: Jose Antonio Jamilena Daza

Pantalla principal de la aplicación.

##### **1.1.16.1 Atributos de analizador::View**

Atributos	Tipo	Notas
exportarCSV	privado : javax.swing.JMenu Item	
generarInforme	privado : javax.swing.JButton ton	
graficasMenu	privado : javax.swing.JMenu nu	
graficoPorCliente	privado : javax.swing.JMenu Item	
graficoPorSGBD	privado : javax.swing.JMenu Item	
graficoPorScript	privado : javax.swing.JMenu Item	
mainPanel	privado : javax.swing.JPanel nel	
menuBar	privado : javax.swing.JMe	

	nuBar	
openMenuItem	privado : javax.swing.JMenuItem	
progressBar	privado : javax.swing.JProgressBar	
receiveMenuItem	privado : javax.swing.JMenuItem	
EstadoAnimationLabel	privado : javax.swing.JLabel	
EstadoMessageLabel	privado : javax.swing.JLabel	
EstadoPanel	privado : javax.swing.JPanel	
messageTimer	privado no modificable : Timer	
busyIconTimer	privado no modificable :	



	Timer	
idleIcon	privado no modificable : Icon	
busyIcons	privado no modificable : Icon	
busyIconIndex	privado : int	
aboutBox	privado : JDialog	
openBox	privado : JDialog	
selectBox	privado : JDialog	
sgbdBox	privado : JDialog	
clienteBox	privado : JDialog	
netBox	privado : JDialog	

csvBox	privado : JDialog	
--------	----------------------	--

#### 1.1.16.2 Métodos de analizador::View

Método	Tipo	Notas
View (SingleFrameApplication )	público:	
openFile ()	público: void	
showAboutBox ()	público: void	
initComponents ()	privado: void	
graficosPorScript ()	público: void	
graficosPorSGBD ()	público: void	
graficosPorCliente ()	público: void	
generarInforme ()	público: void	
recibirPorRed ()	público: void	
exportarCSV ()	público: void	

## 1.2 JOSEJAMILENA::PFC::ANALIZADOR::SQL

### 1.2.1 *SQL::SQLITE2CSV*

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: sql

Detalles: Creado el 16/07/2009 11:32:09. Modificado el 16/07/2009 11:32:09.

Autor: Jose Antonio Jamilena Daza

Convierte datos SQLite a fichero CSV.

#### 1.2.1.1 Métodos de sql::SQLite2CSV

Método	Tipo	Notas
SQLite2CSV ()	público:	
listaRegistros (Connection, String)	privado: List < List < String > >	
csv (String, Connection, String)	público no modificable: void	

### 1.2.2 *SQL::SQLUTILS*

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: sql

Detalles: Creado el 16/07/2009 11:32:09. Modificado el 16/07/2009 11:32:09.

Autor: Jose Antonio Jamilena Daza

Utilidades SQL. No se permiten instancias.

#### 1.2.2.1 Métodos de *sql::SQLUtils*

Método	Tipo	Notas
SQLUtils ()	privado:	
listaScript (Connection)	público y de clase: List<String>	
listaHostSgbd (Connection)	público y de clase: List<String>	
listaHostCliente (Connection)	público y de clase: List<String>	

## 1.3 JOSEJAMILENA::PFC::SERVIDOR::CHARTSERVER

### 1.3.1 CHARTSERVER::CLIENTHANDLER

Tipo: Clase de paquete

Implementa: Thread.

Estado: Versión 1.0. Fase 1.0.

Paquete: chartserver

Detalles: Creado el 16/07/2009 11:33:53. Modificado el 16/07/2009 11:33:53.

Autor: Jose Antonio Jamilena Daza

Hilo de ejecución del servidor.

#### 1.3.1.1 Atributos de chartserver::ClientHandler

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(ClientHandler.class) ;
miSocketServidor	privado : Socket	Socket.
nombreFichero	privado : String	Nombre de fichero index para este hilo.

bufferSize	privado no modificable: int	Tamaño del buffer de lectura.  Valor inicial: 4096;
numGrupos	privado no modificable : int	Número de grupos de estadísticas.  Valor inicial: 3;
longitudNombre	privado no modificable : int	Longitud del nombre de fichero.  Valor inicial: 8;

#### 1.3.1.2 Métodos de chartserver::ClientHandler

Método	Tipo	Notas
ClientHandler (Socket)	público:	Parámetro: s [ Socket - in ] socket  Hilo de ejecución del navegado cliente. Genera una página web para el explorador cliente según los datos disponibles.
run ()	público: void	Método que ejecuta el Thread.  annotations = '@Override'

### 1.3.2 CHARTSERVER::GRAFICO

Tipo: Clase de paquete

Estado: Versión 1.0. Fase 1.0.

Paquete: chartserver

Detalles: Creado el 16/07/2009 11:33:53. Modificado el 16/07/2009 11:33:53.

Autor: Jose Antonio Jamilena Daza

Sus instancias son objetos que se emplean para definir un Grafico.

#### 1.3.2.1 Atributos de chartserver::Grafico

Atributo	Tipo	Notas
tituloEjeX	privado String	: Titulo para el eje X.
tituloEjeY	privado String	: Titulo para el eje Y.
lista	privado LinkedList String <	: lista de consultas.

#### 1.3.2.2 Métodos de chartserver::Grafico

Método	Tipo	Notas
Grafico ()	público:	Constructor.

Grafico (String, String, List<String>)	público:	<p>Parámetro: tituloX [ String - in ] nombre eje X</p> <p>Parámetro: tituloY [ String - in ] nombre eje Y</p> <p>Parámetro: list [ List&lt;String&gt; - in ] lista</p> <p>Constructor.</p>
getLista ()	público: List<String>	<p>Obtiene la lista.</p> <p>Devuelve: lista</p>
setLista (List < String >)	público: void	<p>Parámetro: list [ List &lt; String &gt; - in ] lista</p> <p>Establece la lista.</p>
getTituloEjeX ()	público: String	<p>Obtiene nombre del eje X.</p> <p>Devuelve: nombre</p>
setTituloEjeX (String)	público: void	<p>Parámetro: tituloX [ String - in ] nombre</p> <p>Define el titulo del eje Y.</p>
getTituloEjeY ()	público: String	<p>Obtiene nombre del eje Y.</p> <p>Devuelve: nombre</p>
setTituloEjeY (String)	público: void	<p>Parámetro: tituloY [ String - in ] nombre</p>



		Define el titulo del eje Y.
--	--	-----------------------------

### 1.3.3 CHARTSERVER::GRUPOCONSULTA

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: chartserver

Detalles: Creado el 16/07/2009 11:33:53. Modificado el 16/07/2009 11:33:53.

Autor: Jose Antonio Jamilena Daza

Las instancias de esta clase son objetos formados por una lista de consultas, y un nombre para dicho grupo.

#### 1.3.3.1 Atributos de chartserver::GrupoConsulta

Atributo	Tipo	Notas
titulo	privado String	: Nombre del grupo.
lista	privado List < String >	: Lista de consultas.

#### 1.3.3.2 Métodos de chartserver::GrupoConsulta

Método	Tipo	Notas
GrupoConsulta (String, List < String >)	público:	<p>Parámetro: titu [ String - in ] nombre del grupo.</p> <p>Parámetro: list [ List &lt; String &gt; - in ] lista de consultas.</p> <p>Constructor.</p>
getTitulo ()	público no modificable: String	Devuelve: el titulo
setTitulo (String)	público no modificable: void	Parámetro: titu [ String - in ] el titulo to set
getLista ()	público no modificable: List < String >	Devuelve: the lista
setLista (List < String >)	público no modificable: void	Parámetro: list [ List < String > - in ] the lista to set

### 1.3.4 CHARTSERVER::SQLUTILS

Tipo: Clase de paquete

Estado: Versión 1.0. Fase 1.0.

Paquete: chartserver

Detalles: Creado el 16/07/2009 11:33:53. Modificado el 16/07/2009 11:33:53.

Autor: Jose Antonio Jamilena Daza

Utilidades SQL.

#### 1.3.4.1 Métodos de chartserver::SQLUtils

Método	Tipo	Notas
SQLUtils ()	privado:	No se permiten instancias.
consultaSQL2Grafico (Connection, String)	público static: Grafico	Parámetro: conn [ Connection - in ] conexion JDBC  Parámetro: s [ String - in ] sentencia SQL  Obtiene datos desde SQL y los convierte en Grafico. Devuelve: Grafico
listaGruposConsultas (Connection)	público static: List<GrupoCons ulta>	Parámetro: conn [ Connection - in ] conexion JDBC

		Obtiene una lista de grupos de consulta. Devuelve: lista de grupos de consulta
--	--	---

### 1.3.5 CHARTSERVER::WEBSERVER

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: chartserver

Detalles: Creado el 16/07/2009 11:33:54. Modificado el 16/07/2009 11:33:54.

Autor: Jose Antonio Jamilena Daza

Servidor web para mostrar gráficas con Google Charts.

#### 1.3.5.1 Atributos de chartserver::Webserver

Atributo	Tipo	Notas
configProperties	privado y de clase: Properties	propiedades de configuración.
logger	privado y de clase: Logger	Logger. Valor inicial: Logger.getLogger(Webserver.class);

serverSocket	privado y de clase: ServerSocket	Socket servidor.
httpPort	privado y de clase: int	Puerto HTTP empleado.

#### 1.3.5.2 Métodos de chartserver::Webserver

Método	Tipo	Notas
Webserver ()	privado:	No se permiten instancias de la clase.
main (String[])	público static: void	<p>Parámetro: args [ String[] - in ] puerto HTTP que abrirá, si no se indica, usa el 80.</p> <p>Método principal. throws = 'IOException,ClassNotFoundException' - @exception ClassNotFoundException no encuentra el driver JDBC @exception java.io.IOException Error.</p>
getConfigProperties ()	público static: Properties	Devuelve: the configProperties
setConfigProperties (Properties)	público static: void	Parámetro: aConfigProperties [ Properties - in ] the configProperties to set

## 1.4 JOSEJAMILENA::PFC::SERVIDOR

### 1.4.1 *SERVIDOR::MAIN*

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: servidor

Detalles: Creado el 16/07/2009 11:35:20. Modificado el 16/07/2009 11:35:20.

Autor: Jose Antonio Jamilena Daza.

Cargador inicial.

#### 1.4.1.1 **servidor::Main Attributes**

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(Main.class);

#### 1.4.1.2 **Métodos de servidor::Main**

Método	Tipo	Notas
Main ()	privado:	

main (String[])	público y de clase: void	Parámetro: args [ String[] - in ] argumentos. main.
-----------------	-----------------------------	--

## 1.5 JOSEJAMILENA::PFC::SERVIDOR::CONEXION

### 1.5.1 CONEXION::COMUN

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: conexion

Detalles: Creado el 16/07/2009 11:35:17. Modificado el 16/07/2009 11:35:17.

Autor: Jose Antonio Jamilena Daza.

Métodos compartidos y conexiones. Diseñada con un patrón singleton.

#### 1.5.1.1 Atributos de conexion::Comun

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(Comun.class);

comun	privado y de clase: Comun	Instancia Singleton.  Valor inicial: null;
estadisticas	privado y de clase: Estadisticas	Instancia de Estadisticas.  Valor inicial: null;
hostname	privado y de clase: String	Hostname.  Valor inicial: "";
configProperties	privado : Properties	properties de configuración.

#### 1.5.1.2 Métodos de conexion::Comun

Método	Tipo	Notas
Comun ()	privado:	Constructor.
getComun ()	público y de clase: Comun	Obtiene instancia de Comun. Devuelve: instancia de Comun.
iniciarPropiedades ()	privado: void	Iniciar properties.
getConfigProperties ()	público: Properties	Devuelve: the ConfigProperties
getEstadisticas ()	público y de clase:	Devuelve: the estadisticas



	Estadísticas	
getHostname ()	público: String	Devuelve: the hostname

### 1.5.2 CONEXION::

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: conexion

Detalles: Creado el 16/07/2009 11:35:18. Modificado el 16/07/2009 11:35:18.

Autor: Jose Antonio Jamilena Daza.

Crono se encarga de medir el tiempo.

#### 1.5.2.1 Atributos de conexion::Crono

Atributo	Tipo	Notas
t0	privado : java.util.Date	Tiempo inicial.
t1	privado : java.util.Date	Tiempo final.

#### 1.5.2.2 Métodos de conexion::Crono

Método	Tipo	Notas
Crono ()	público:	Constructor.
inicializa ()	público      no modificable: void	Inicia la cuenta.
tiempo ()	público      no modificable: long	Devuelve el tiempo desde el inicio de cuenta            en            segundos. Devuelve: tiempo desde el inicio.

### 1.5.3 CONEXION::ESTADISTICAS

Tipo:            Clase Pública

Estado:        Versión 1.0. Fase 1.0.

Paquete:      conexion

Detalles:      Creado el 16/07/2009 11:35:18. Modificado el 16/07/2009 11:35:18.

Autor:         Jose Antonio Jamilena Daza.

Estadísticas.

#### 1.5.3.1 Atributos de conexion::Estadísticas

Atributo	Tipo	Notas

logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(Estadisticas.class);
uno	privado no modificable : int	1.  Valor inicial: 1;
dos	privado no modificable : int	2.  Valor inicial: 2;
tres	privado no modificable : int	3.  Valor inicial: 3;
cuatro	privado no modificable : int	4.  Valor inicial: 4;
cinco	privado no modificable : int	5.  Valor inicial: 5;
conexion	privado : Connection	conexion.  Valor inicial: null;
configProperties	privado : Properties	properties.

### 1.5.3.2 Métodos de conexión::Estadísticas

Método	Tipo	Notas
Estadísticas (String, String, String)	público:	<p>Parámetro: driver [ String - in ] driver JDBC.</p> <p>Parámetro: url [ String - in ] enlace.</p> <p>Parámetro: properties [ String - in ] fichero de properties para Estadísticas.</p> <p>Constructor.</p>
iniciarPropiedades (String)	privado: void	<p>Parámetro: prop [ String - in ] nombre del fichero.</p> <p>Iniciar propiedades.</p>
insertarEstadística (long, String, String, String)	público no modificable: void	<p>Parámetro: tiempo [ long - in ] tiempo.</p> <p>Parámetro: tipoConsultaStr [ String - in ] tipo de consulta como cadena de texto.</p> <p>Parámetro: sgbd [ String - in ]</p> <p>Parámetro: host [ String - in ]</p> <p>Insertar estadísticas en la base de datos.</p>

iniciarEstructuraEnBD (Connection)	privado: void	<p>Parámetro: conn [ Connection - in ] conexión a la base de datos.</p> <p>Inicializa la tabla en la que se guardan las estadísticas.</p>
consultaSQL (String)	privado: String	<p>Parámetro: s [ String - in ] consulta SQL.</p> <p>Lanza la consulta SQL.</p> <p>Devuelve: resultado.</p>
mostrarMedia ()	público no modificable: String	<p>Media de tiempos del fichero.</p> <p>Devuelve: tiempo de la media en segundos.</p> <p>throws = 'SQLException' - @exception java.sql.SQLException excepcion SQL.</p>
mostrarMediaSeleccion ( )	público no modificable: String	<p>Media de tiempos en consultas de selección (SELECT).</p> <p>Devuelve: tiempo de la media en segundos.</p> <p>throws = 'SQLException' - @exception java.sql.SQLException excepcion SQL.</p>
mostrarMediaManipulacionDatos ()	público no modificable:	<p>Media de tiempos en manipulación de datos (INSERT / UPDATE /DELETE).</p> <p>Devuelve: tiempo de la media en</p>

	String	segundos.  throws = 'SQLException' - @exception  java.sql.SQLException      excepcion SQL.
mostrarMediaManipulacionTablas ()	público      no modificable: String	Media de tiempos en manipulación de tablas (DROP / CREATE). Devuelve: tiempo de la media en segundos.  throws = 'SQLException' - @exception  java.sql.SQLException      excepcion SQL.
mostrarMediaOtrasOperaciones ()	público      no modificable: String	Media de tiempos de otras operaciones (?). Devuelve: tiempo de la media en segundos.  throws = 'SQLException' - @exception  java.sql.SQLException      excepcion SQL.
mostrarMediaOtrasOperacionesPrueba ()	público      no modificable: String	Media de tiempos de otras operaciones de la última prueba (?). Devuelve: tiempo de la media en segundos.

		throws = 'SQLException' - @exception java.sql.SQLException      excepcion SQL.
--	--	--

## 1.6 JOSEJAMILENA::PFC::SERVIDOR::TAREAS

### 1.6.1 TAREAS::CARGARTAREAS

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: tareas

Detalles: Creado el 16/07/2009 11:35:19. Modificado el 16/07/2009 11:35:19.

Autor: Jose Antonio Jamilena Daza.

Cargador de tareas.

#### 1.6.1.1 Atributos de tareas::CargarTareas

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial:

		Logger.getLogger(CargarTareas.class);
tablaScript	privado : Map<String, String>	Tareas SQL.  Valor inicial: null;
tablaConexionScript	privado : Map<String, TokenConexion >	Conexión SQL.  Valor inicial: null;
tablaPlsql	privado : Map<String, String>	Tareas PLSQL.  Valor inicial: null;
tablaConexionPlsql	privado : Map<String, TokenConexion >	Conexión PLSQL.  Valor inicial: null;
numeroTokens	privado no modificable : int	Número de tokens que espera de cada línea del fichero de tareas.  Valor inicial: 8;

#### 1.6.1.2 Métodos de tareas::CargarTareas

Método	Tipo	Notas
CargarTareas ()	privado:	Constructor.
CargarTareas (String)	público:	Parámetro: nombreFichero [ String - in



		<p>] fichero.</p> <p>Carga las tareas del fichero. throws = 'IOException' - @exception java.io.IOException error de lectura.</p>
getTablaScript ()	público no modificable: Map<String, String>	Devuelve: the tablaScript
getTablaPlsql ()	público no modificable: Map<String, String>	Devuelve: the tablaPlsql
getTablaConexionScript ()	público: Map<String, TokenConexion >	Devuelve: the tablaConexionScript
getTablaConexionPlsql ()	público: Map<String, TokenConexion >	Devuelve: the tablaConexionPlsql

### 1.6.2 TAREAS::TASKPLSQL

Tipo: Clase Pública

Implementa: Runnable.

Estado: Versión 1.0. Fase 1.0.

Paquete: tareas

Detalles: Creado el 16/07/2009 11:35:20. Modificado el 16/07/2009 11:35:20.

Autor: Jose Antonio Jamilena Daza.

Define una tarea PL/SQL.

#### 1.6.2.1 Atributos tareas::TaskPLSQL

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(TaskPLSQL.class);
nombreFichero	privado : String	Nombre del fichero.  Valor inicial: "";
nombreTarea	privado : String	Nombre de la tarea.  Valor inicial: "";
token	privado : TokenConexion	Autenticación.  Valor inicial: null;
crn	privado : Crono	Crono.

		Valor inicial: new Crono();
t	privado long	: Tiempo.

### 1.6.2.2 Métodos de tareas::TaskPLSQL

Método	Tipo	Notas
TaskPLSQL ()	privado:	No se permite el constructor sin parámetros.
TaskPLSQL (String, String, TokenConexion)	público:	<p>Parámetro: nombre [ String - in ] nombre de la tarea.</p> <p>Parámetro: ficheroPLSQL [ String - in ] fichero de la tarea.</p> <p>Parámetro: tc [ TokenConexion - in ] autenticacion</p> <p>Constructor.</p>
run ()	público no modificable: void	Sobrecarga el método.
obtenerTiempo ()	público no modificable: long	<p>Obtiene tiempo.</p> <p>Devuelve: tiempo en segundos.</p>

### 1.6.3 TAREAS::TASKSQL

Tipo: Clase Pública

Implementa: Runnable.

Estado: Versión 1.0. Fase 1.0.

Paquete: tareas

Detalles: Creado el 16/07/2009 11:35:20. Modificado el 16/07/2009 11:35:20.

Autor: Jose Antonio Jamilena Daza.

Define una tarea SQL.

#### 1.6.3.1 Atributos de tareas::TaskSQL

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(TaskSQL.class);
nombreFichero	privado : String	Nombre del fichero.  Valor inicial: "";
nombreTarea	privado : String	Nombre de la tarea.  Valor inicial: "";
token	privado :	Autenticación.

	TokenConexion	Valor inicial: null;
crn	privado : Crono	Crono.  Valor inicial: new Crono();
t	privado : long	Tiempo.

### 1.6.3.2 Métodos de tareas::TaskSQL

Método	Tipo	Notas
TaskSQL ()	privado:	No se permite constructor sin parámetros.
TaskSQL (String, String, TokenConexion)	público:	Parámetro: nombre [ String - in ] nombre de la tarea.  Parámetro: ficheroSQL [ String - in ] fichero scripts.  Parámetro: tc [ TokenConexion - in ] autenticación  Constructor.
run ()	público no modificable: void	Método que sobrescribe.
obtenerTiempo ()	público no modificable:	Obtiene tiempo.

	long	Devuelve: tiempo en segundos.
--	------	-------------------------------

## 1.7 JOSEJAMILENA::PFC::SERVIDOR::AUTENTICACION

### 1.7.1 AUTENTICACION::TOKENCONEXION

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: autenticacion

Detalles: Creado el 16/07/2009 11:35:18. Modificado el 16/07/2009 11:35:18.

Autor: Jose Antonio Jamilena Daza

Token de Autenticación.

#### 1.7.1.1 Atributos de autenticacion::TokenConexion

Atributo	Tipo	Notas
hostname	privado String	: Hostname.  Valor inicial: "";
driver	privado String	: Driver.  Valor inicial: "";

cadenaConexion	privado String	:	Cadena de conexión.  Valor inicial: "";
usuario	privado String	:	Usuario.  Valor inicial: "";
password	privado String	:	Contraseña.  Valor inicial: "";

#### 1.7.1.2 Métodos de autenticación::TokenConexion

Método	Tipo	Notas
TokenConexion (String, String, String, String)	público	Parámetro: sgbd [ String - in ] sgbd  Parámetro: drv [ String - in ] driver.  Parámetro: url [ String - in ] cadena de conexión.  Parámetro: user [ String - in ] usuario.  Parámetro: passwd [ String - in ] contraseña.  Constructor.
getCadenaConexion ()	público no modificable: String	Devuelve: the cadenaConexion

setCadenaConexion (String)	público modificable: void	no	Parámetro: url [ String - in ] the cadenaConexion to set
getUsuario ()	público modificable: String	no	Devuelve: the usuario
setUsuario (String)	público modificable: void	no	Parámetro: user [ String - in ] the usuario to set
getPassword ()	público modificable: String	no	Devuelve: the password
setPassword (String)	público modificable: void	no	Parámetro: passwd [ String - in ] the password to set
getDriver ()	público modificable: String	no	Devuelve: the driver
setDriver (String)	público modificable: void	no	Parámetro: drv [ String - in ] the driver to set
getHostname ()	público modificable:	no	Devuelve: the hostname



	String	
setHostname (String)	público no modificable: void	Parámetro: host [ String - in ] hostname the hostname to set

## 1.8 JOSEJAMILENA::PFC::SERVIDOR::RUNNER

### 1.8.1 RUNNER::PLSQLRUNNER

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: runner

Detalles: Creado el 16/07/2009 11:35:19. Modificado el 16/07/2009 11:35:19.

Autor: Jose Antonio Jamilena Daza.

Lanzador de script con procesos PL/SQL.

#### 1.8.1.1 Atributos de runner::PlsqlRunner

Atributo	Tipo	Notas
logger	privado y de clase:	Logger.  Valor inicial:

	Logger	Logger.getLogger(PlsqlRunner.class);
conexion	privado : Connection	conexion.  Valor inicial: null;
listaDeConsultas	privado : List < String >	lista de consultas.  Valor inicial: null;

#### 1.8.1.2 Métodos de runner::PlsqlRunner

Método	Tipo	Notas
PlsqlRunner (Connection)	público:	Parámetro: connection [ Connection - in ] conexión JDBC.  Lanzador SQL.
PlsqlRunner (String, String, String, String)	público:	Parámetro: driver [ String - in ] driver.  Parámetro: url [ String - in ] cadena de conexión.  Parámetro: username [ String - in ] usuario.  Parámetro: password [ String - in ] contraseña.  Lanzador SQL.  throws = 'IOException,SQLException,ClassNot

		<p>FileNotFoundException' - @exception          java.lang.ClassNotFoundException          driver no encontrado.          @exception java.sql.SQLException          fallo en JDBC.          @exception java.io.IOException fallo          de lectura.</p>
consultaSQL (String)	privado: void	<p>Parámetro: s [ String - in ] consulta          SQL.</p> <p>Lanza la consulta indicada.          throws = 'SQLException' - @exception          java.sql.SQLException fallo.</p>
runScript (Reader)	público no modificable: void	<p>Parámetro: reader [ Reader - in ]          reader.</p> <p>Leer script SQL.</p> <p>throws = 'IOException,SQLException'          - @exception java.sql.SQLException          fallo JDBC.          @exception java.io.IOException error          de lectura.</p>

### 1.8.2 RUNNER::SQLRUNNER

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: runner

Detalles: Creado el 16/07/2009 11:35:19. Modificado el 16/07/2009 11:35:19.

Autor: Jose Antonio Jamilena Daza.

Lanzador de script con procesos SQL.

#### 1.8.2.1 Atributos de runner::SqlRunner

Atributo	Tipo	Notas
logger	privado y de clase: Logger	Logger.  Valor inicial: Logger.getLogger(SqlRunner.class);
conexion	privado : Connection	Conexión.  Valor inicial: null;
listaDeConsultas	privado : List < String >	Lista de consultas.  Valor inicial: null;

#### 1.8.2.2 Métodos de runner::SqlRunner

Método	Tipo	Notas
SqlRunner (Connection)	público:	Parámetro: connection [ Connection - in ] conexión JDBC.  Lanzador SQL.

SqlRunner (String, String, String)	público:	<p>Parámetro: driver [ String - in ] driver.</p> <p>Parámetro: url [ String - in ] cadena de conexión.</p> <p>Parámetro: username [ String - in ] usuario.</p> <p>Parámetro: password [ String - in ] contraseña.</p> <p>Lanzador SQL. throws = 'IOException,SQLException,ClassNotFoundException' - @exception java.lang.ClassNotFoundException driver no encontrado. @exception java.sql.SQLException fallo en JDBC. @exception java.io.IOException fallo de lectura.</p>
consultaSQL (String)	privado: void	<p>Parámetro: s [ String - in ] consulta SQL.</p> <p>Lanza la consulta indicada.</p> <p>throws = 'SQLException' - @exception java.sql.SQLException fallo.</p>
runScript (Reader)	público no modificable:	<p>Parámetro: reader [ Reader - in ]</p>

	void	<p>reader.</p> <p>Leer script SQL.</p> <p>throws = 'SQLException,IOException'</p> <p>- @exception java.sql.SQLException fallo JDBC. @exception java.io.IOException error de lectura.</p>
--	------	--

## 1.9 JOSEJAMILENA::PFC::SERVIDOR::CRYPTO::EASY::CHEC KSUM

### 1.9.1 *CHECKSUM::CHECKSUM*

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: checksum

Detalles: Creado el 16/07/2009 11:34:33. Modificado el 16/07/2009 11:34:33.

Autor: Jose Antonio Jamilena Daza

Comprobador de suma.

#### 1.9.1.1 Métodos de checksum::Checksum

Método	Tipo	Notas
Checksum ()	privado:	No se podrá instanciar.
checksum (String)	público y de clase: long	<p>Parámetro: fichero [ String - in ] fichero</p> <p>Obtiene la suma de comprobación del fichero dado.</p> <p>Devuelve: suma</p> <p>throws = 'IOException' - @exception java.io.IOException error</p>

## 1.10 JOSEJAMILENA::PFC::SERVIDOR::TCP

### 1.10.1 TCP::FILERECEIVER

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: tcp

Detalles: Creado el 16/07/2009 11:34:34. Modificado el 16/07/2009 11:34:34.

Autor: Jose Antonio Jamilena Daza

Receptor de ficheros.

#### 1.10.1.1 Métodos de tcp::FileReceiver

Método	Tipo	Notas
receive (String, int)	público y no modificable: void	Parámetro: hostname [ String - in ] host  Parámetro: port [ int - in ] puerto  Recibir.
receive2 (String, int)	público y no modificable: String	Parámetro: hostname [ String - in ] host  Parámetro: port [ int - in ] puerto  Recibir.  Devuelve: nombre fichero  throws = 'TCPException' - @exception TCPException error

#### 1.10.2 TCP::FILESENDER

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.



Paquete: tcp

Detalles: Creado el 16/07/2009 11:34:34. Modificado el 16/07/2009 11:34:34.

Autor: Jose Antonio Jamilena Daza.

Servicio que envía los ficheros de BBDD de estadísticas por TCP.

#### 1.10.2.1 Atributos de tcp::FileSender

Atributo	Tipo	Notas
port	privado y de clase: int	Puerto TCP.
file	privado y de clase: String	Nombre del fichero.  Valor inicial: "";

#### 1.10.2.2 Métodos de tcp::FileSender

Método	Tipo	Notas
getPort ()	público y de clase: int	Devuelve: the port
setPort (int)	público y de clase: void	Parámetro: aPort [ int - in ] the port to set
getFile ()	público y de clase: String	Devuelve: the file

setFile (String)	público y de clase: void	Parámetro: aFile [ String - in ] the file to set
FileSender (String, int)	público:	Parámetro: fileToSend [ String - in ] fichero  Parámetro: tcpPort [ int - in ] puerto  Constructor.
runServer ()	público y no modificable: void	Ejecutor.

### *1.10.3 TCP::FILESENDER::HANDLER*

Tipo: Clase de paquete

Implementa: Thread.

Estado: Versión 1.0. Fase 1.0.

Paquete: tcp

Detalles: Creado el 16/07/2009 11:34:34. Modificado el 16/07/2009 11:34:34.

Autor: Jose Antonio Jamilena Daza

Hilo de ejecución.

#### 1.10.3.1 Atributos de tcp::FileSender::Handler

Atributo	Tipo	Notas
sock	privado Socket	: Socket.
file	privado String	: Fichero.

#### 1.10.3.2 Método de tcp::FileSender::Handler

Método	Tipo	Notas
Handler (Socket, String)	Paquete:	Parámetro: s [ Socket - in ] socket  Parámetro: fileToSend [ String - in ] fichero  Constructor.
run ()	público: void	Método que ejecuta.  annotations = '@Override'
getSock ()	público: Socket	Devuelve: the sock
setSock (Socket)	público: void	Parámetro: aSock [ Socket - in ] the sock to set
getFile ()	público: String	Devuelve: the file

setFile (String)	público: void	Parámetro: aFile [ String - in ] the file to set
------------------	---------------	--

#### *1.10.4 TCP::TCPEXCEPTION*

Tipo: Clase Pública

Implementa: Exception.

Estado: Versión 1.0. Fase 1.0.

Paquete: tcp

Detalles: Creado el 16/07/2009 11:34:34. Modificado el 16/07/2009 11:34:34.

Autor: Jose Antonio Jamilena Daza

Excepción.

##### **1.10.4.1 Métodos de tcp::TCPEXception**

Método	Tipo	Notas
TCPEXception ()	público:	Creates a new instance of <code>TCPEXception</code> without detail message.
TCPEXception (String)	público:	Parámetro: msg [ String - in ] the detail message.  Constructs an instance of

		<code>TCPException</code> with the specified detail message.
--	--	--

## 1.11 JOSEJAMILENA::PFC::SERVIDOR::UTIL

### 1.11.1 UTIL::CHANNELTOOLS

Tipo: Clase Pública

Estado: Versión 1.0. Fase 1.0.

Paquete: util

Detalles: Creado el 16/07/2009 11:34:35. Modificado el 16/07/2009 11:34:35.

Autor: Jose Antonio Jamilena Daza

Utilidades para trabajar con Channels.

#### 1.11.1.1 Métodos de util::ChannelTools

Método	Tipo	Notas
ChannelTools ()	privado:	No instanciable.
fastChannelCopy (ReadableByteChannel, WritableByteChannel)	público y de clase: void	Parámetro: src [ ReadableByteChannel - in ] origen  Parámetro: dest [ WritableByteChannel

		<p>- in ] destino</p> <p>Copia rápida por Channels.</p> <p>throws = 'IOException' - @exception java.io.IOException error</p>
--	--	--