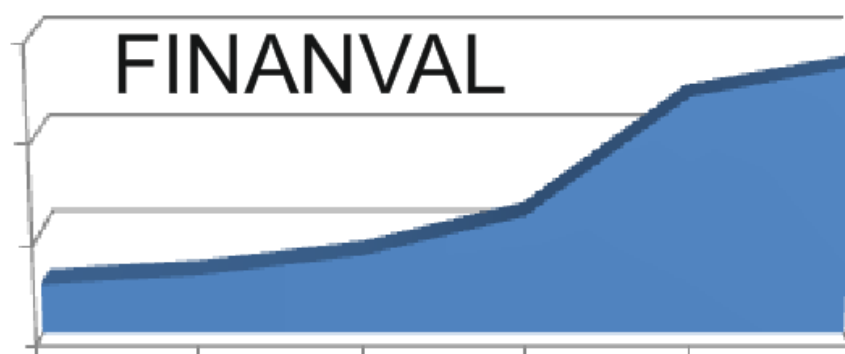




**Universitat Oberta  
de Catalunya**

## **TFC-Plataforma GNU/Linux 2011-2012 Q2**

**FINANVAL – APLICACIÓN DE CÁLCULO FINANCIERO**



06/04/2012

UOC

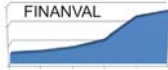
Manuel Martínez Liébana

Profesor responsable de la asignatura: Helena Rifà Pous

Consultores: Joaquín López Sánchez-Montañés y Miquel Àngel Senar Rosell

## Contenido

1	INTRODUCCIÓN .....	4
1.1	Descripción.....	4
1.2	Alcance.....	4
1.2.1	Datos de Mercado: .....	5
1.2.2	Datos estáticos: .....	5
1.2.3	Datos contractuales.....	5
1.2.4	Net Present Value.....	5
1.2.5	Cálculo de sensibilidades.....	5
1.3	Enfoque.....	6
1.4	Planificación .....	7
1.4.1	Desglose.....	7
1.4.2	Cronograma de tareas.....	8
1.4.3	Diagrama de Gantt .....	8
1.4.4	Diagrama visual y de convergencia con PECs.....	9
2	ANÁLISIS .....	10
2.1	Requisitos funcionales .....	10
2.1.1	Estructuración de la información y modularización del aplicativo. ....	10
2.1.2	Motor de valoración.....	10
2.1.3	Captura dinámica y automática de datos de mercado. ....	15
2.2	Requisitos técnicos .....	16
2.2.1	Lenguaje de programación .....	16
2.2.2	Entorno de desarrollo.....	16
2.2.3	Interfaz gráfica.....	16
3	DISEÑO .....	17
3.1	Estructuración gráfica del aplicativo.....	17
3.2	Almacenamiento de DB. ....	24
3.2.1	Tablas.....	25
3.2.2	Esquema entidad-relación.....	36
3.3	Motor de valoración. ....	38
4	DESARROLLO .....	38
4.1	Ficheros del aplicativo. ....	38
4.1.1	Módulos de proyecto .....	38



4.1.2	Módulos c++, cabeceras y ficheros de definición de pantalla ui.....	38
5	PRUEBAS.....	41
6	MANUAL DE INSTALACIÓN Y USO .....	42
6.1	Instalación.....	42
6.1.1	Instalación automática .....	42
6.1.2	Instalación Manual .....	43
6.2	Manual de Usuario.....	44
6.2.1	Configuración.....	44
6.2.2	Set de productos líquidos .....	50
6.2.3	Uso.....	55
7	Licencia.....	61

# 1 INTRODUCCIÓN

## 1.1 Descripción

Después de un análisis del campo financiero en la plataforma Linux y más en concreto sobre aplicaciones dedicadas a la valoración de estos productos, se observa que no hay una gran especialización en esta materia. Podemos encontrar diferentes productos pero en su mayoría dedicados a gestión contable y administración empresarial. Alguna calculadora que ofrece de manera muy superficial y lateral el aspecto financiero y poco más.

Así pues, el objetivo de este TFC se centrará en la construcción de una calculadora especializada en la valoración de productos financieros, que hasta la fecha, han estado restringidos a las tesorerías de banca o de algunas sociedades de inversión.

El usuario potencial será el ciudadano de a pie, que mediante acceso a estos productos a través de paquetes comercializados mediante campañas por banca dirigidos a al cliente medio, que básicamente suelen dividir grandes contratos de estos tipos de productos en participaciones que se comercializan habitualmente en oficinas y encapsulados por el concepto de fondo de inversión.

De esta manera el usuario potencial podrá pre calcular el riesgo de este fondo midiendo de forma cuantitativa las expectativas y riesgos de estos productos observando cómo se moverán durante la vida de su productos los subyacentes a los que este es sensible, tales como el valor esperado de tipos de interés, spots de divisas, etc, etc. Como output, el usuario obtendrá cuánto vale el producto hoy (NPV, Net Present Value) y qué exposición cuantificada en valor monetario está asumiendo.

## 1.2 Alcance

Se desarrollará una aplicación en entorno gráfico para gestionar todos los datos necesarios para realizar valoraciones de productos los productos financieros descritos a continuación. Además y aunque la principal finalidad del producto será la valoración se proveerá funcionalidad para poder realizar el booking de contratos (también llamados boletas o deals), para poder analizar en distintos días el mismo contrato sin tener que insertar todos los datos cada vez que se desee consultar su evolución.

De este modo las funcionalidades sobre la que la aplicación será enfocada son:

### 1.2.1 Datos de Mercado:

Estos son los inputs necesarios en la valoración cuantitativa de productos financieros, por ejemplo algunos de ellos son curvas de tipos, curvas de divisas, spots, volatilidades en formato matricial, etc. Se estudiará y en su caso se implementará la alimentación automática de estos datos de mercados. Básicamente se estudiará y es posible alimentar aunque con delay de forma gratuita a la aplicación.

### 1.2.2 Datos estáticos:

Definen las características contractuales del producto, podemos incluir en este grupo datos como la base de cálculo elegida, divisas disponibles, convenciones, etc.

### 1.2.3 Datos contractuales

Definen el settlement del producto, hablamos por ejemplo de maturity, start date, strikes para el caso de opciones, etc.

### 1.2.4 Net Present Value

Proporcionar NPV de los siguientes productos: Interest Rate Swaps y Opciones Europeas. Si la planificación lo permitiese se podrían incluir productos adicionales. Se introducirá exotividad en algunos de estos productos, es decir, valoración de opciones sobre estos productos con modelo de valoración estocástico, habitualmente con metodología Montecarlo, estos serían productos con volatilidad, un ejemplo podrían ser Swaptions (Opciones sobre IRS). En el caso que nos aplica esta exotividad la podremos encontrar en la Opciones Europeas incluidas en el producto.

### 1.2.5 Cálculo de sensibilidades

El Cálculo de sensibilidades será realizado mediante generación de escenarios de mercado. Básicamente se trata de alterar los datos de mercado de los que depende el producto incrementando habitualmente un punto básico, se realiza la valoración del producto antes y después de este incremento obteniendo así la cantidad de dependencia por punto básico del producto. Visto desde un punto matemático, lo que hacemos es calcular la derivada de la curva de valoración. Las sensibilidades que se calcularán serán: Delta (sensibilidad a subyacente), Gamma (segunda derivada de Delta), Rho (sensibilidad a divisa), Vega (sensibilidad a volatilidad), Volga (segunda derivada de Vega), Theta (sensibilidad al tiempo).

## 1.3 Enfoque

Se trata de construir una aplicación gráfica que permita al usuario la gestión de una cartera de productos financieros. Entendiendo como la gestión el cálculo de sensibilidades que permitirá al usuario conocer qué fechas son cruciales para su cartera, obteniendo información de la cantidad en divisas que su cartera se podría ver afectada en estas fechas.

Además de esto, el usuario debería poder acceder a la posición real de su cartera, es decir, si decidiese vender hoy sus productos, cual es el precio exacto de mercado. Es necesario notar que rara vez ese precio de mercado coincidirá con el precio al que el usuario adquirió su producto. En función del estado actual de mercado y la previsión de futuro de todos los actores que confluyen en el mercado el producto habrá sido revalorizado o depreciado. Un ejemplo básico puede ser la valoración de un IRS (Interest Rate Swap), un IRS es un producto en el que el usuario y una contraparte comparten un contrato por la que el usuario recibe un tipo fijo o variable (indexado) y la contraparte un tipo fijo o variable. Se suele decir que es un producto con dos patas, una fija y otra flotante. El usuario estará posicionado en una y la contraparte en otra. Si ocurre por ejemplo que el tipo variable está indexado sobre un índice, supongamos EURIBOR y el Banco Central sube los tipos de interés, esto redundará sobre Euribor esperando así que quien está posicionado en la pata variable del producto pase a recibir más cantidad de dinero en los próximos flujos (pagos). Así pues si el poseedor de la pata flotante vende el contrato, recibirá una mayor cantidad de dinero que cuando lo compró.

Se analizará e intentará dotar al sistema de una interfaz de recogida automática de datos de mercado, es de esperar si se encuentra algún proveedor dispuesto a compartir sus datos de manera gratuita sea porque estos tienen demora. También suele ser habitual que esta demora esté en 15 minutos y 1 hora. Para el objetivo de nuestro proyecto es más que suficiente, ya que la aplicación va enfocada a un usuario no profesional, es decir, a alguien que no trabaja en la tesorería de un banco y que tiene que tomar decisiones en tiempo real.

También sería necesario poder dotar al usuario de la posibilidad de alterar determinados datos de mercado bajo petición para poder evaluar cómo reaccionaría su producto a estos cambios.

## 1.4 Planificación

### 1.4.1 Desglose

La planificación del proyecto se divide en 4 bloques o fases:

#### 1.4.1.1 Análisis

Las tareas principales son identificadas a analizar divididas en dos grandes grupos, requerimientos funcionales y técnicos.

##### 1.4.1.1.1 Requisitos funcionales

- Estructuración de la información y modularización del aplicativo.
- Motor de valoración.
- Captura dinámica y automática de datos de mercado.

##### 1.4.1.1.2 Requisitos técnicos

- Lenguaje de programación
- Entorno de desarrollo
- Interfaz gráfica

Las tareas de Análisis irán desde el 26 de Marzo hasta el 24 de Abril.

#### 1.4.1.2 Diseño

- Estructuración gráfica del aplicativo.
- Almacenamiento de DB.
- Motor de valoración.

Las tareas de diseño irán desde el 26 de Abril hasta el 9 de Mayo.

#### 1.4.1.3 Desarrollo

- Estructuración gráfica del aplicativo y conexión con la algoritmia.
- Motor de valoración.

Las tareas de desarrollos se extenderán desde el 10 de Mayo hasta el 4 de Junio

#### 1.4.1.4 Pruebas

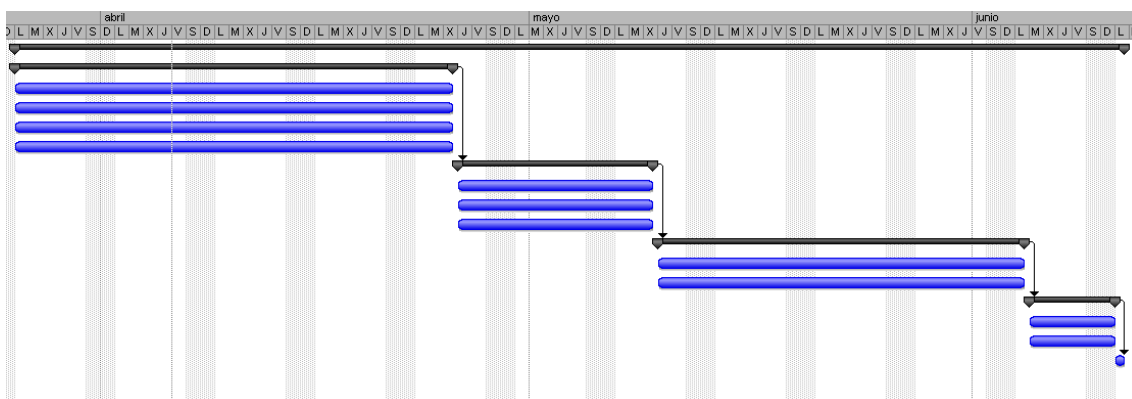
- Unitarias.
- Integradas.

Las pruebas serán realizadas en el periodo comprendido entre el 5 de Junio y el 10 de Junio.

### 1.4.2 Cronograma de tareas

		Nombre de tarea	Duración	Comienzo	Fin	Predecesoras
1		<input type="checkbox"/> <b>TFC</b>	<b>56 días</b>	<b>lun 26/03/12</b>	<b>lun 11/06/12</b>	
2		<input type="checkbox"/> <b>Análisis</b>	<b>23 días</b>	<b>lun 26/03/12</b>	<b>mié 25/04/12</b>	
3		Entorno, Interfaz y	23 días	lun 26/03/12	mié 25/04/12	
4		Estructuración Infr	23 días	lun 26/03/12	mié 25/04/12	
5		Motor de valoració	23 días	lun 26/03/12	mié 25/04/12	
6		Captura dinámica i	23 días	lun 26/03/12	mié 25/04/12	
7		<input type="checkbox"/> <b>Diseño</b>	<b>10 días</b>	<b>jue 26/04/12</b>	<b>mié 09/05/12</b>	<b>2</b>
8		Estructuración gráf	10 días	jue 26/04/12	mié 09/05/12	
9		DB	10 días	jue 26/04/12	mié 09/05/12	
10		Motor de valoració	10 días	jue 26/04/12	mié 09/05/12	
11		<input type="checkbox"/> <b>Desarrollo</b>	<b>18 días</b>	<b>jue 10/05/12</b>	<b>lun 04/06/12</b>	<b>7</b>
12		Entorno gráfico	18 días	jue 10/05/12	lun 04/06/12	
13		Motor de valoració	18 días	jue 10/05/12	lun 04/06/12	
14		<input type="checkbox"/> <b>Pruebas</b>	<b>4 días</b>	<b>mar 05/06/12</b>	<b>dom 10/06/12</b>	<b>11</b>
15		Unitarias	4 días	mar 05/06/12	dom 10/06/12	
16		Integración	4 días	mar 05/06/12	dom 10/06/12	
17		Go Live	1 día	lun 11/06/12	lun 11/06/12	14

### 1.4.3 Diagrama de Gantt

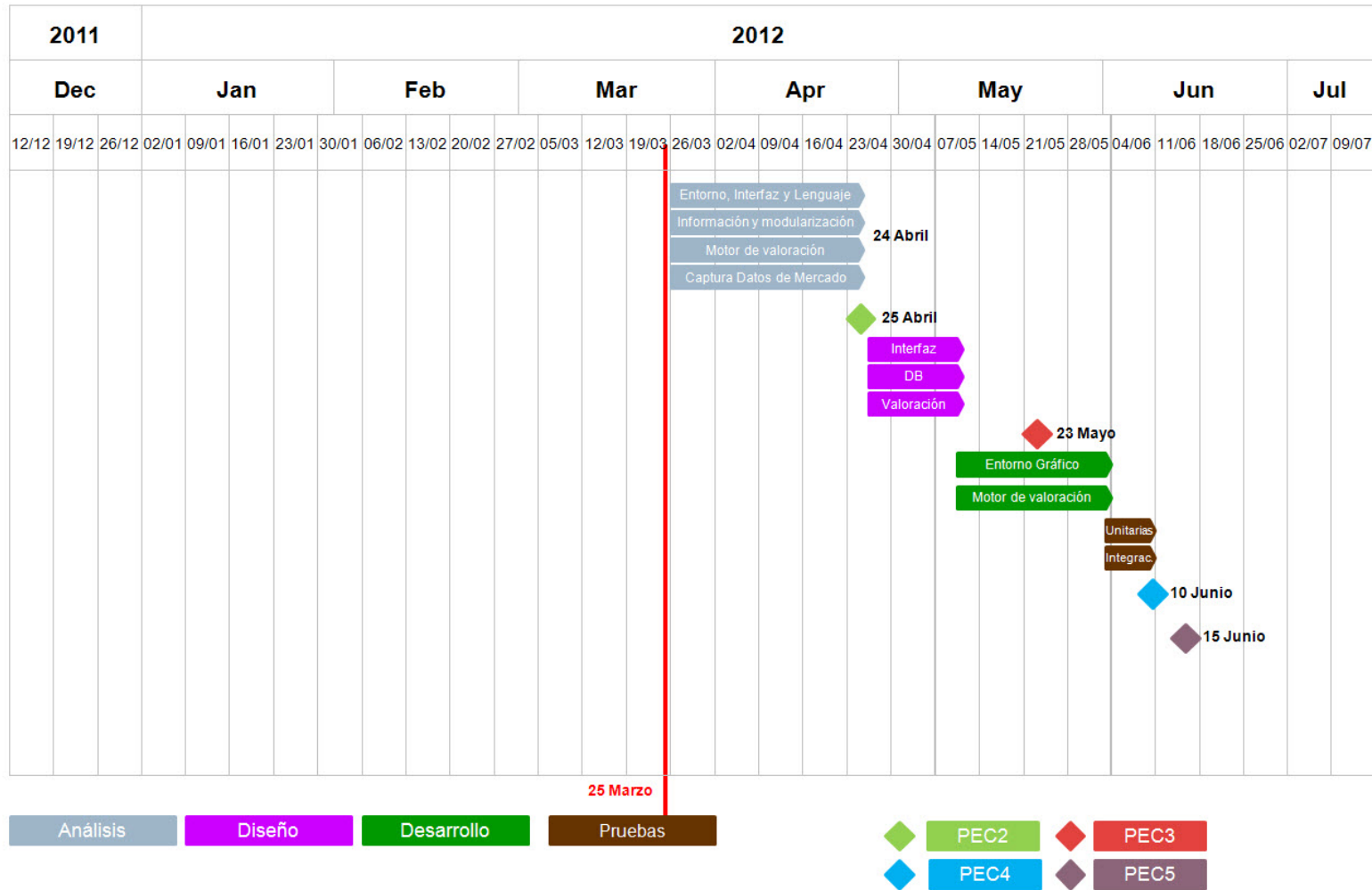




1.4.4 Diagrama visual y de convergencia con PECs

# TFC-Plataforma GNU/Linux

## Calculadora Financiera. Manuel Martínez Liébana



## 2 ANÁLISIS

### 2.1 Requisitos funcionales

Analizaremos en este apartado cuáles son los requerimientos funcionales deseados para nuestro aplicativo.

#### 2.1.1 Estructuración de la información y modularización del aplicativo.

Dada la naturaleza del aplicativo, el modo más racional de modularizar la información será en torno a los siguientes ejes:

- **Valoración:**  
Agrupará toda la funcionalidad específica de valoración cuantitativa de los modelos. Además proveerá la funcionalidad adicional necesaria por esta, tal como la construcción de curvas, procesos de bootstrapping, etc.
- **Interfaz:**  
Agrupará toda la funcionalidad asociada a la inserción, modificación o eliminación de información proporcionada por el usuario, así como todo lo relacionado con la gestión de outputs del aplicativo tales como informes, presentación de valoraciones, etc.
- **DB:**  
Agrupará toda la funcionalidad necesaria para la persistencia de la información en DB. Así como su recuperación tanto para procesos relacionados con la interfaz como aquellos relacionados con valoración.

#### 2.1.2 Motor de valoración.

Resultado del análisis quedan clarificados los datos necesarios para valorar este tipo de productos. Profundizamos a continuación en todo lo necesario para estructurar la valoración:

##### 2.1.2.1 Curvas (por divisas)

- **Curva de descuento:** Se utilizará para obtener el valor presente los distintos flujos que se percibirán en el futuro por los pagadores (fijo o variable).
- **Curva de estimación de cupones:** En algunos productos una de las partes, generalmente es pagador variable, en este caso debemos estimar los flujos que pagará el receptor fijo con los tipos forward. Los tipos forward estimados se obtienen de la Curva de Estimación que no necesariamente debe coincidir con la Curva de Descuento.

Para construir la Curva de Estimación y la Curva de Descuento necesitamos definir los componentes en cada uno de los plazos en los tramos cortos (hasta 1 año), lo habitual es utilizar Euribor, Tipos FRA, Call Money Swap y Futuros, mientras que a partir de 2

años hasta 30 o 60 años, o el plazo al que construimos la curva debemos utilizar los tipos EuroSwap.

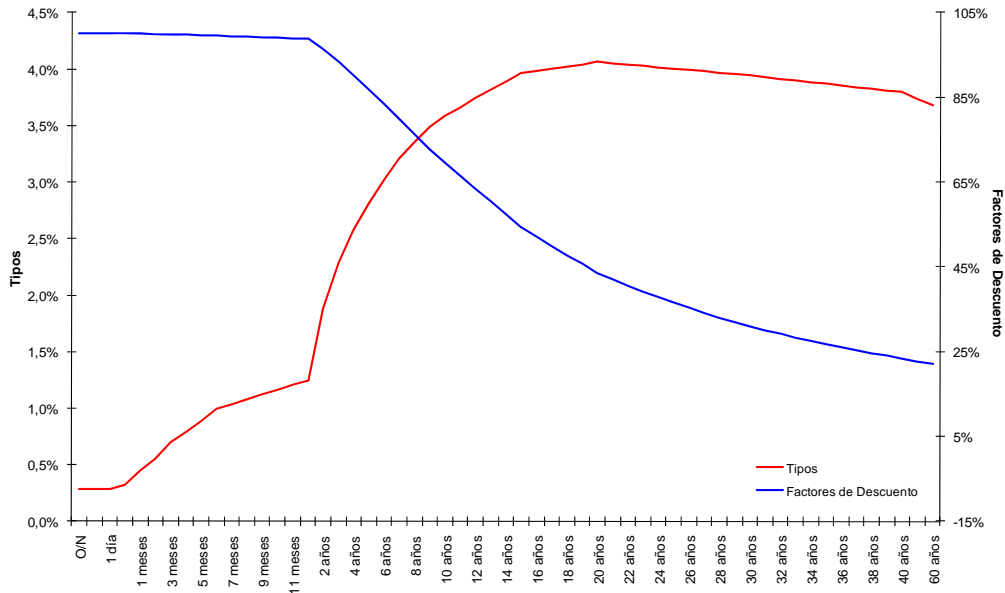
Una vez definida la curva encontramos los factores de descuento para cada plazo utilizando Bootstrapping. Veamos como ejemplo una posible forma como podría definirse la curva:

Fuente	PLAZOS	TIPOS	Fuente	PLAZOS	TIPOS
Depositos Interbancarios	O/N	0,290%	EuroSwap	14 años	3,889%
Depositos Interbancarios	T/N	0,290%	EuroSwap	15 años	3,963%
Depositos Interbancarios	1 día	0,290%	EuroSwap	16 años	3,983%
Call Money Swap	7 días	0,325%	EuroSwap	17 años	4,003%
Call Money Swap	1 meses	0,453%	EuroSwap	18 años	4,022%
Tipos FRA	2 meses	0,556%	EuroSwap	19 años	4,042%
Tipos FRA	3 meses	0,700%	EuroSwap	20 años	4,062%
Tipos FRA	4 meses	0,794%	EuroSwap	21 años	4,050%
Tipos FRA	5 meses	0,887%	EuroSwap	22 años	4,038%
Tipos FRA	6 meses	0,994%	EuroSwap	23 años	4,027%
Tipos FRA	7 meses	1,035%	EuroSwap	24 años	4,015%
Tipos FRA	8 meses	1,080%	EuroSwap	25 años	4,003%
Tipos FRA	9 meses	1,127%	EuroSwap	26 años	3,991%
Tipos FRA	10 meses	1,166%	EuroSwap	27 años	3,979%
Tipos FRA	11 meses	1,207%	EuroSwap	28 años	3,968%
Tipos FRA	12 meses	1,248%	EuroSwap	29 años	3,956%
EuroSwap	2 años	1,883%	EuroSwap	30 años	3,944%
EuroSwap	3 años	2,285%	EuroSwap	31 años	3,929%
EuroSwap	4 años	2,572%	EuroSwap	32 años	3,914%
EuroSwap	5 años	2,809%	EuroSwap	33 años	3,900%
EuroSwap	6 años	3,024%	EuroSwap	34 años	3,885%
EuroSwap	7 años	3,208%	EuroSwap	35 años	3,870%
EuroSwap	8 años	3,358%	EuroSwap	36 años	3,855%
EuroSwap	9 años	3,482%	EuroSwap	37 años	3,840%
EuroSwap	10 años	3,591%	EuroSwap	38 años	3,826%
EuroSwap	11 años	3,665%	EuroSwap	39 años	3,811%
EuroSwap	12 años	3,740%	EuroSwap	40 años	3,796%
EuroSwap	13 años	3,814%	EuroSwap	50 años	3,738%
			EuroSwap	60 años	3,680%

Después de definir la curva obtenemos los Factores de Descuento por Bootstrapping (se requiere una función/librería que haga el Bootstrapping):

PLAZOS	FD	PLAZOS	FD
O/N	99,999%	14 años	57,402%
T/N	99,998%	15 años	54,430%
1 día	99,996%	16 años	52,127%
7 días	99,992%	17 años	49,881%
1 meses	99,959%	18 años	47,719%
2 meses	99,904%	19 años	45,623%
3 meses	99,816%	20 años	43,586%
4 meses	99,728%	21 años	42,047%
5 meses	99,625%	22 años	40,577%
6 meses	99,490%	23 años	39,165%
7 meses	99,390%	24 años	37,828%
8 meses	99,272%	25 años	36,552%
9 meses	99,142%	26 años	35,329%
10 meses	99,021%	27 años	34,161%
11 meses	98,885%	28 años	33,045%
12 meses	98,742%	29 años	31,976%
2 años	96,325%	30 años	30,961%
3 años	93,407%	31 años	30,040%
4 años	90,257%	32 años	29,162%
5 años	86,918%	33 años	28,325%
6 años	83,381%	34 años	27,522%
7 años	79,817%	35 años	26,765%
8 años	76,318%	36 años	26,044%
9 años	72,905%	37 años	25,354%
10 años	69,559%	38 años	24,697%
11 años	66,491%	39 años	24,071%
12 años	63,426%	40 años	23,470%
13 años	60,407%	50 años	22,730%
		60 años	22,015%

Gráficamente los tipos (eje izquierdo) y los factores de descuento (eje derecho) tienen el siguiente comportamiento:



Un punto importante a señalar en el gráfico anterior es que los factores de descuento y los tipos tienen convexidad.

Es requerida una fuente de Datos para las curvas, a analizar en 2.1.3.

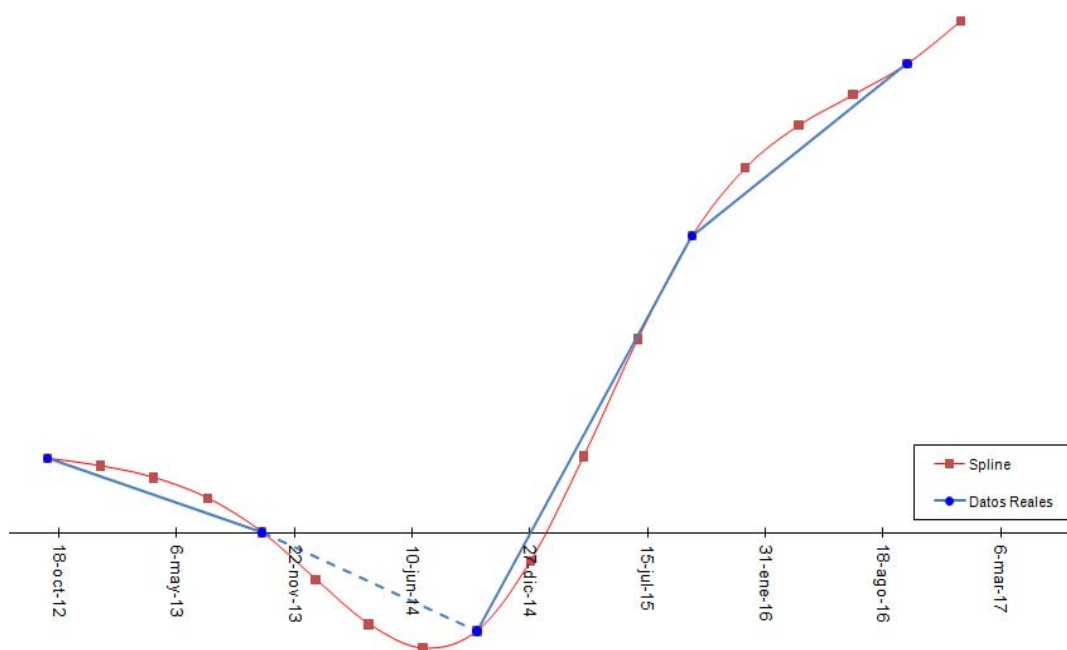
### 2.1.2.2 Función de interpolación

Una vez que tengamos definidos los calendarios de fijación y de pago, estos probablemente no coincidan con los plazos para los que se han definido las curvas de estimación y descuento.

Por ejemplo, si tenemos el tipo a 6 meses y a 7 meses con sus respectivos Factores de Descuento, pero uno de los pagos se hace a los 6,8 meses, para el cual no hay ni tipo ni factor de descuento definidos.

Debemos encontrar el tipo (y Factor de descuento) a 6,8 meses. Necesitamos una función de interpolación, que dado un tipo  $r_1$  y un tipo  $r_2$  en los plazos  $t_1$  y  $t_2$ , pueda encontrar un tipo  $r_i$  que este incluido en cualquier punto del intervalo  $t_1, t_2$ .

Como conocemos que hay convexidad en los tipos, lo más adecuado sería utilizar una función de interpolación que capte este efecto. Un método que cumple con este requerimiento es **Splines Cúbicos**<sup>1</sup>.



Una función de interpolación lineal, interpolaría los datos entre 22-nov-2013 y 27-dic-2014 en la línea azul puntuada, mientras que con Splines Cúbicos se haría en la línea roja.

<sup>1</sup> [http://es.wikipedia.org/wiki/Spline#Interpolaci.C3.B3n\\_Segmentaria\\_C.C3.BAlica](http://es.wikipedia.org/wiki/Spline#Interpolaci.C3.B3n_Segmentaria_C.C3.BAlica)

### 2.1.2.3 Información General del Producto:

En este punto se incluye información del producto como Vencimiento, Divisa, Nominal, Etc., datos generales necesarios para la valoración de los flujos del pagador fijo y o del pagador variable.

### 2.1.2.4 Calendarios:

Se requiere definir las siguientes fechas en cada periodo para el pagador fijo y receptor fijo:

Start Dates, End Dates, Fixing Dates y Payment Dates. Mostramos un ejemplo de calendarios para un Swap pagador fijo contra Euribor 6M a 3 años.

PAGADOR FIJO					
Start Dates	End Dates	Tipo Fijo	Nominal	Payment Dates	Pagos Fijos
30/09/2011	28/09/2012	2,500%	100.000.000,00	28/09/2012	PAGO FIJO 1
28/09/2012	30/09/2013	2,500%	100.000.000,00	30/09/2013	PAGO FIJO 2
30/09/2013	30/09/2014	2,500%	100.000.000,00	30/09/2014	PAGO FIJO 3
30/09/2014	30/09/2015	2,500%	100.000.000,00	30/09/2015	PAGO FIJO 4
30/09/2015	30/09/2016	2,500%	100.000.000,00	30/09/2016	PAGO FIJO 5

PAGADOR VARIABLE						
Start Dates	End Dates	Fixing Dates	Margin	Nominal	Payment Dates	Pagos Flotantes
30/09/2011	30/03/2012	28/09/2011	0,000%	100.000.000,00	30/03/2012	PAGO FLOTANTE 1
30/03/2012	28/09/2012	28/03/2012	0,000%	100.000.000,00	28/09/2012	PAGO FLOTANTE 2
28/09/2012	28/03/2013	26/09/2012	0,000%	100.000.000,00	28/03/2013	PAGO FLOTANTE 3
28/03/2013	30/09/2013	26/03/2013	0,000%	100.000.000,00	30/09/2013	PAGO FLOTANTE 4
30/09/2013	31/03/2014	26/09/2013	0,000%	100.000.000,00	31/03/2014	PAGO FLOTANTE 5
31/03/2014	30/09/2014	27/03/2014	0,000%	100.000.000,00	30/09/2014	PAGO FLOTANTE 6
30/09/2014	30/03/2015	26/09/2014	0,000%	100.000.000,00	30/03/2015	PAGO FLOTANTE 7
30/03/2015	30/09/2015	26/03/2015	0,000%	100.000.000,00	30/09/2015	PAGO FLOTANTE 8
30/09/2015	30/03/2016	28/09/2015	0,000%	100.000.000,00	30/03/2016	PAGO FLOTANTE 9
30/03/2016	30/09/2016	24/03/2016	0,000%	100.000.000,00	30/09/2016	PAGO FLOTANTE 10

EL Calendario considerara los días no hábiles, festivos y reglas para finales de mes.

Hay dos formas estándar de definir cuando se fijan Fixing Dates:

- In Advance: Los pagos se fijan en el inicio del periodo de cálculo de intereses y se pagan al final del periodo de cálculo de intereses. A inicio solo se conoce el primer "Fixing".
- In Arrears: Los pagos se fijan al final del periodo de cálculo de intereses (dos días hábiles antes) y se pagan al final del periodo de cálculo de intereses.

Es necesario también conocer la Frecuencia de pago y Basis (convención para contar días/años), para la construcción de calendarios.

Es necesario saber la frecuencia de pago del pagador fijo y o del receptor fijo. Generalmente la frecuencia del pagador fijo es anual y la del receptor fijo viene

vinculada a la frecuencia del índice de referencia de los pagos variables. Por ejemplo si el receptor fijo paga Euribor 3M, la frecuencia de pago es trimestral.

El Basis habitual para el pagador fijo es 30/360 (todos los días del mes se consideran de 30 días y todos los años de 360 días, los meses siempre son número enteros) mientras que para el receptor fijo es ACT/360 (los días se cuentan como número total de días en cada período de cálculo de intereses, los años se consideran de 360 días)

#### **2.1.2.5 Código/Librería de Valoración:**

Finalmente necesitamos una Función de Valoración que utilice todos los inputs necesarios para la valoración que hemos mencionado y determine cuánto es el valor actual de los flujos del pagador fijo, cuánto es el valor actual de los flujos del receptor fijo, la valoración del producto y sensibilidad a los plazos del mismo.

#### **2.1.3 Captura dinámica y automática de datos de mercado.**

Después de una amplia prospección en busca de un proveedor que facilite un pool de datos de mercado completo en diferido de forma gratuita, se comprueba que no existe ninguno. Habitualmente se localizan proveedores que proporcionan un set limitado de datos de mercado de pruebas con licencia trial, pero insuficiente a todas luces para nuestro propósito. Por tanto, se analiza la mejor forma de dejar abierta la entrada de datos de mercado al aplicativo por parte de proveedores. Se ha valorado la construcción de un API que el usuario final (por otro lado muy especializado en este tipo de aplicación) usaría para conectar su proveedor de pago. No obstante, se descarta esta posibilidad ya que ataríamos al usuario a un lenguaje y plataforma determinado.

La solución pasa por diseñar la base de datos de tal manera que el aplicativo recoja estos datos desde ella, de este modo, el usuario podrá utilizar un lenguaje estándar ya que casi todos ellos proveen interfaces de conexión a la base de datos.

## 2.2 Requisitos técnicos

### 2.2.1 Lenguaje de programación

La decisión sobre el lenguaje de programación afecta en gran medida al análisis y toma de decisión sobre el Entorno de desarrollo e interfaz gráfica.

Así pues se han analizado 2 tipologías de lenguajes, compilados e interpretados. Dentro del grupo de los lenguajes interpretados, se ha analizado Python y Java. El segundo de ellos fue descartado ya que el lenguaje de programación debe proveernos alta velocidad. Python en este ámbito es bastante más rápido que Java después de alguna prueba de performance realizada adhoc para este fin. Python ofrece una amplia gama de módulos para programación gráfica. No obstante finalmente se descarta ya que a pesar de ser muy rápido para ser un lenguaje interpretado, sigue siendo excesivamente lento para realizar las tareas de valoración que nos aplican.

Así pues, la primera decisión es clara, el lenguaje apropiado es compilado. Entre ellos se seleccionó C y C++. El primero es más rápido que el segundo sin embargo el segundo es lo suficientemente rápido para la aplicación que queremos construir y por tanto se selecciona por proveer de toda la potencia de la programación orientada a objetos.

### 2.2.2 Entorno de desarrollo

Ya centrados en la interfaz gráfica a usar, se analizó GTK y QT. A pesar de tener la primera más bagaje, se observó que ha quedado más obsoleta, además QT está mucho más documentada que GTK y se observa fácilmente que está mejor soportada y que la tendencia general es usar QT.

Finalmente, después de tener seleccionado C++ y QT el entorno de desarrollo es obvio. QT Creator, ya que provee de una gran funcionalidad a la hora de crear pantallas. Se basa en la generación gráfica de Forms y la implementación de eventos y funciones asociados a los controles incluidos en estos. Debido al escaso tiempo para el desarrollo del aplicativo, esta elección resulta crucial para el éxito del mismo.

### 2.2.3 Interfaz gráfica

Consecuencia inmediata de 2.2.2, la interfaz gráfica será generada con QT Designer. El aplicativo será construido en formato clásico. Una ventana principal con la funcionalidad básica y un menú principal que nos provea de toda la funcionalidad, tanto de configuración como inserción y edición de datos de mercado, así como la gestión de los productos objetivo de este aplicativo.



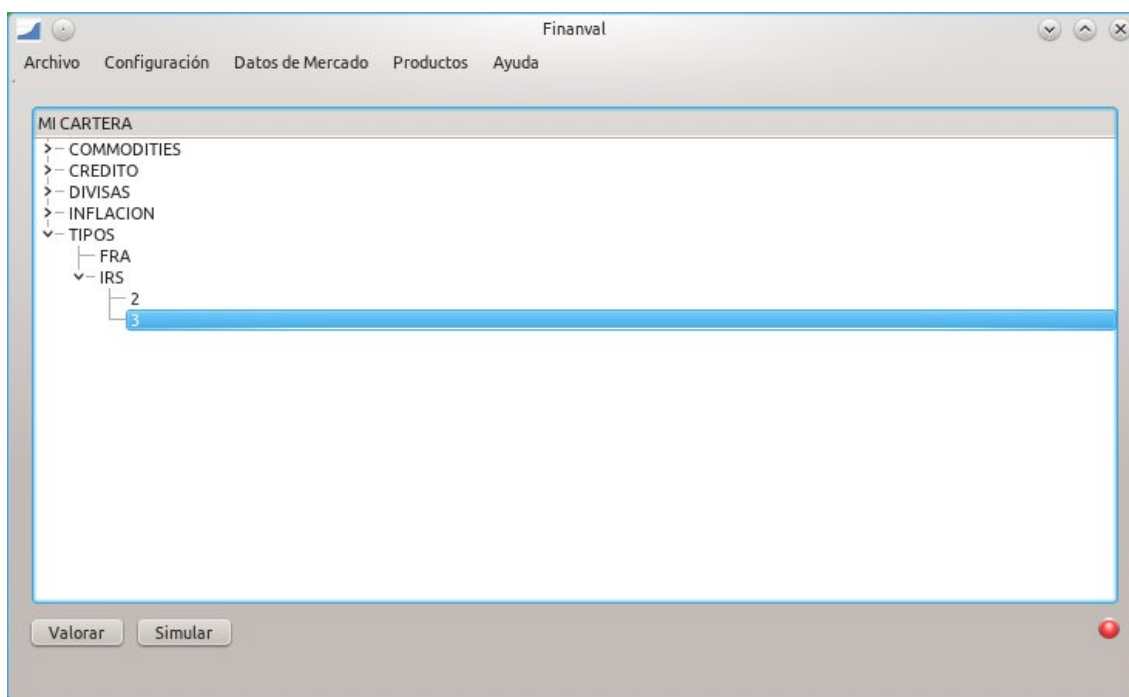
## 3 DISEÑO

### 3.1 Estructuración gráfica del aplicativo.

Se decide implementar una pantalla principal en la que el usuario podrá consultar un árbol en el que se mostrarán las familias de productos y en cada una de ella los tipos de productos disponibles. Dentro de cada uno de los tipos de productos disponibles se mostrarán los contratos que el usuario posee.

Además se habilitará desde esta pantalla controles para poder valorar un producto, es decir, mostrar la pantalla del producto desde la que se podrá obtener su NPV (Net Present Value). Además se disponibilizará la posibilidad de realizar una simulación del producto para obtener la sensibilidad del mismo, o en otras palabras el riesgo que el producto posee.

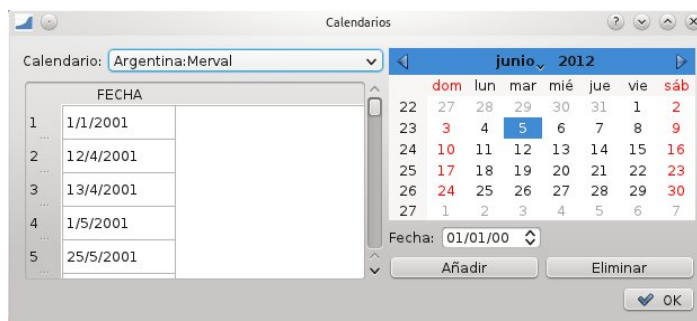
Presenta este aspecto:



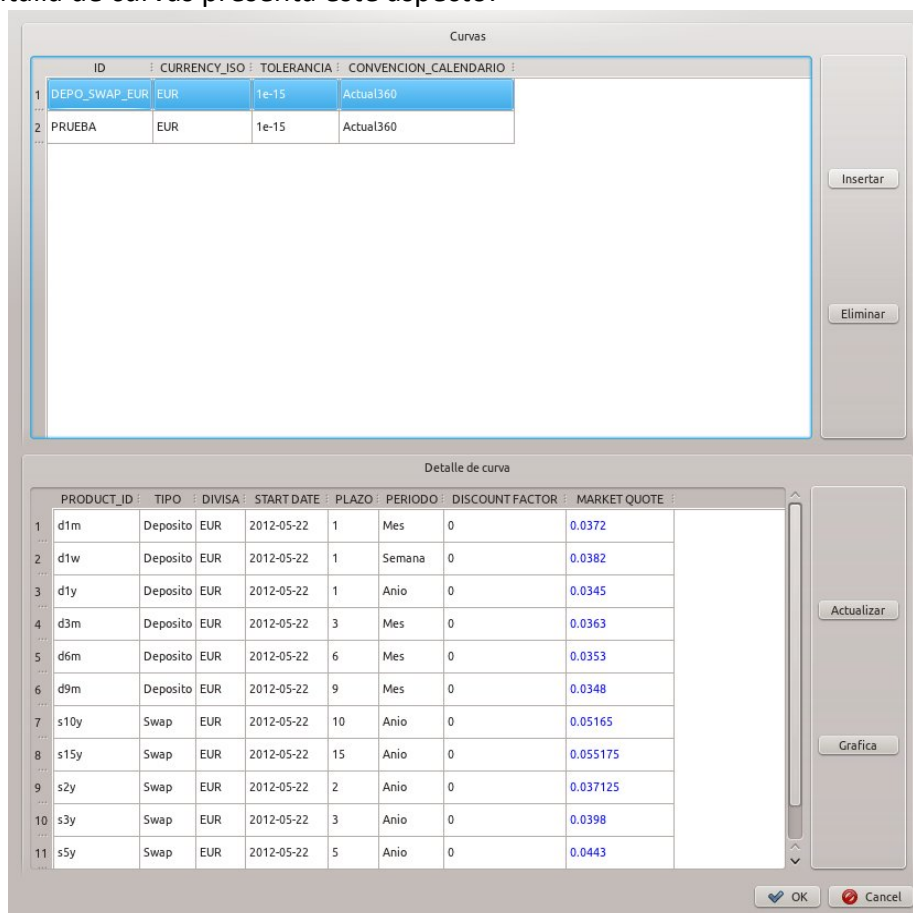
En esta pantalla principal, está incluido un menú superior que provee toda la funcionalidad del aplicativo, asociando una pantalla de input a cada una de las opciones de este menú. Estas opciones son:

- Archivo (exportar, salir):  
Desde este submenú se podrán exportar los deals a formato texto importable desde Excel o su equivalente gnu. Se exporta un fichero con los campos separados por “;” por bloques. Cada bloque contendrá las cabeceras y datos de cada uno de los productos disponibles en Finanval. También se podrá salir del aplicativo.
- Configuración(Calendarios, Curvas, Divisas, Tipos de Cambio, Índices):  
Aquí se podrán imputar todas las configuraciones necesarias para el aplicativo. Estas son la configuración de calendarios disponibles, es decir, qué calendario se asociará a cada divisa con sus festivos prefijados. Será posible insertar y eliminar

festivos (algunas veces ocurre, por ejemplo la boda del príncipe Carlos se marcó como festivo en los mercados ingleses a última hora).  
La pantalla de calendarios presenta este aspecto:



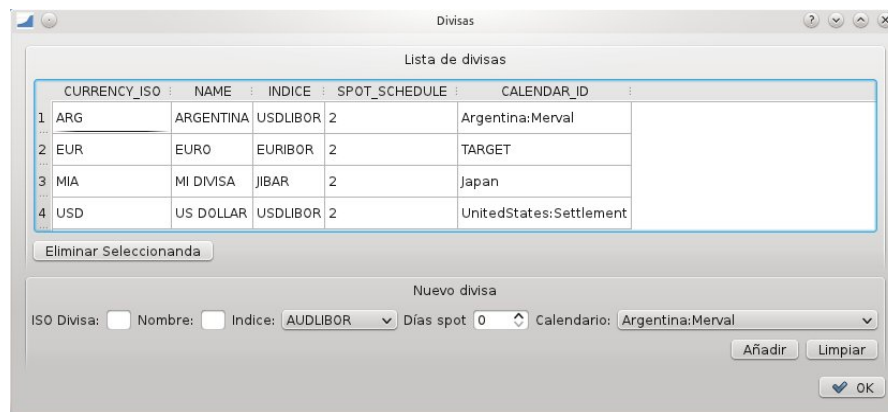
También hablamos de configuración de curvas que se usarán para valorar los productos, estas curvas podrán ser usadas tanto para descontar precios (traer a precio presente valoraciones en el futuro) o como curvas forward, es decir, curvas que se podrán usar para estimar flujos y precios de índices en el futuro. Estas curvas están compuestas de pilares (instrumentos) de mercado líquidos cuyo precio sirve para estimar el rate asociado de la curva en un plazo determinado. En palabras sencillas, una curva da la estimación del mercado de cuanto tipo de interés se puede obtener de una cantidad para un plazo determinado. Para la generación de esta curva se usarán Depósitos, FRAs, Swaps y Futuros, todos ellos estándar y con alta liquidez en el mercado, se definen en el punto siguiente. La pantalla de curvas presenta este aspecto:



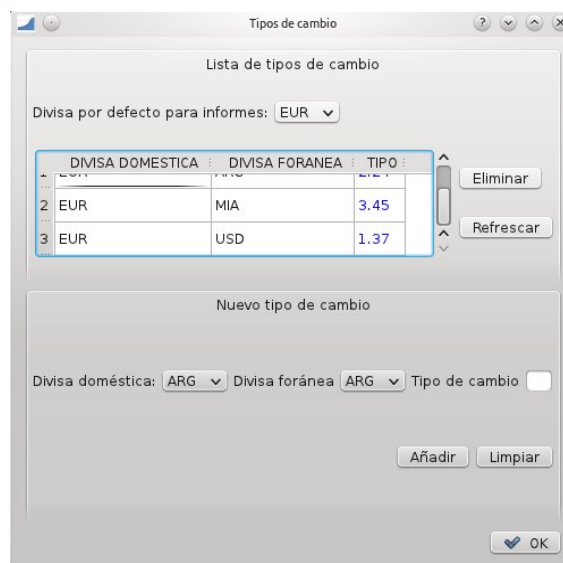
Y la inclusión de una nueva curva (insertar) se gestiona desde la siguiente pantalla:



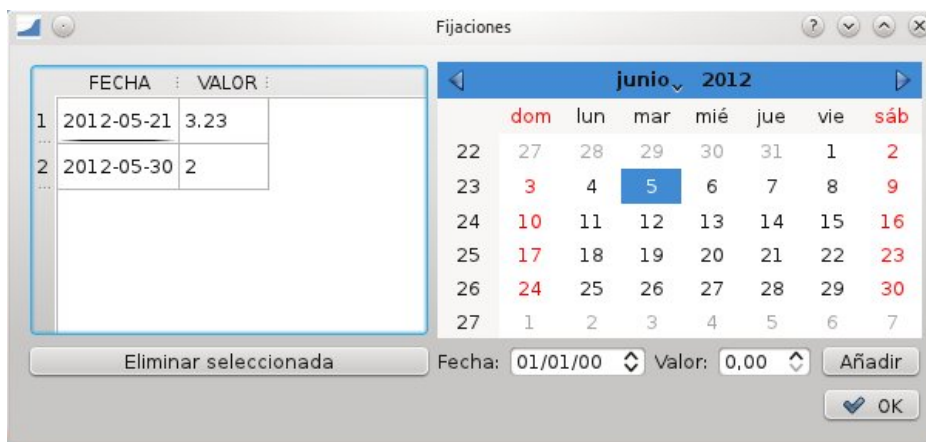
Siguiendo con la configuración, nos encontramos con la opción de configurar divisas. Son aquellas en las que valoraremos posteriormente nuestros productos:



Otra configuración necesaria será la asociación de divisas en pares para establecer los tipos de cambio. En esta pantalla además se definirá la divisa estándar en la que recibiremos los informes:



Dentro de esta sección de configuración también podremos introducir aquellos índices que serán usados en el aplicativo. Estos índices serán configurables con un periodo de fijación definida por el usuario y referenciarán a un índice estándar ya definido en el aplicativo. Además será posible establecer las fijaciones necesarias asociadas al índice para poder realizar valoraciones pasadas. Las pantallas de configuración de índices y fijaciones para el índice son:



- **Datos de mercado:** En esta pantalla se podrán añadir los datos de mercado líquidos junto con su valoración de mercado que serán usados para generar las curvas definidas en el punto anterior.

Dentro de datos de mercado se definen los cuatro grupos diferentes de productos que mejor representan la situación del mercado para los distintos plazos, estos son por orden de periodo de menor a mayor: Depósitos, Fras, Futuros y Swaps. Las pantallas de imputación son las siguientes por el mismo orden:

Depósitos

Lista de depósitos

	IDENTIFICADOR	TIPO	DIVISA	PLAZO	PERIODO	MARKET QUOTE
1	d1m	Deposito	EUR	1	Mes	0.0372
2	d1w	Deposito	EUR	1	Semana	0.0382
3	d1y	Deposito	EUR	1	Anio	0.0345

Nuevo depósito

ID Depósito  Divisa  Plazo  Periodo  Market Quote

Añadir Limpiar

OK Cancel

Forward Rate Agreements

Lista de fras

	IDENTIFICADOR	TIPO	DIVISA	MESES INICIO	MESES FIN	MARKET QUOTE
1	fra3x6	Fra	EUR	3	6	0.037123
2	fra6x12	Fra	EUR	6	12	0.37125
3	fra6x9	Fra	EUR	6	9	0.37125

Nuevo fra

ID Fra  Divisa  Meses a inicio  Meses a fin  Market Quote

Añadir Limpiar

OK Cancel

Futuros

Lista de futuros

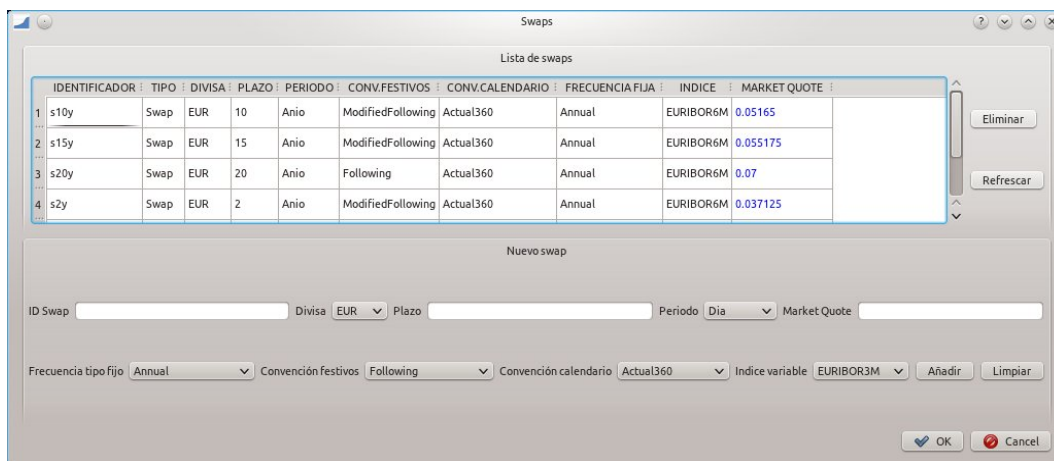
	IDENTIFICADOR	TIPO	DIVISA	MESES	FECHA IMM	MARKET QUOTE
1	fut1	Futuro	EUR	3	2012-06-20	96.2875
2	fut2	Futuro	EUR	3	2012-09-19	96.7875
3	fut3	Futuro	EUR	3	2012-12-19	96.9875
4	fut4	Futuro	EUR	3	2013-03-20	96.6875

Nuevo futuro

ID Futuro  Divisa  Meses  Market Quote  IMM Date:

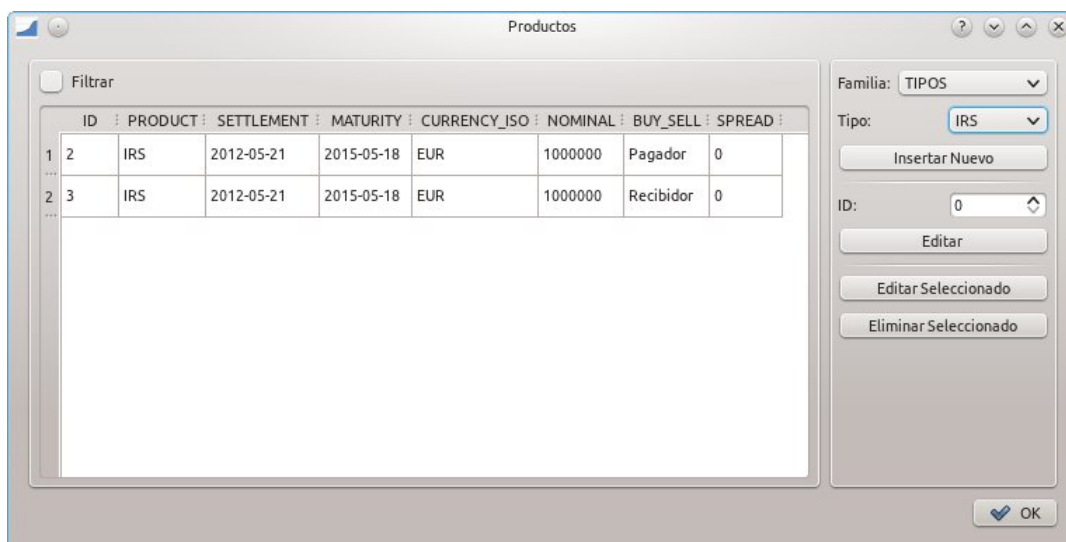
Añadir Limpiar

OK Cancel



- Productos: Aquí encontraremos 2 opciones, la primera “editar” se usará para introducir, valorar, simular, modificar y eliminar contratos de usuario.

El aspecto es el siguiente:



La edición de cada producto será customizada a las necesidades del mismo. En la versión final de este aplicativo:

Interest Rate Swaps en sus dos versiones, estándar y forward start y Opción Europea.

No obstante será diseñado con el resto de familias (Crédito, Equities, etc) para que una vez puesto a disposición de la comunidad (probablemente en sourceforge) sea fácil de escalar a más productos.

Un ejemplo de Interest Rate Swap:

The screenshot shows the 'IRS' configuration window. It is divided into several sections:

- Información de contrato:**
  - Familia: TIPOS
  - Tipo de producto: IRS
  - Contrato: Nuevo
  - Fecha de contratación: 25/05/12 (Desde calendario)
  - Fecha de expiración: 19/05/15 (Desde calendario)
  - Nominal: 100000,00 Divisa: EUR
  - Rol: Pagador
  - Margen(%): 0,00
  - Forward Start
- Información de patas:**
  - Pata Fija:**
    - Frecuencia: Annual
    - Convención de festivos: Following
    - Convención de calendario: Actual360
    - Tipo de interés(%): 0,00
    - Curva de Descuento: DEPO\_SWAP\_EUR
    - Flujos
  - Pata Flotante:**
    - Frecuencia: Semiannual
    - Convención de festivos: ModifiedFollowing
    - Convención de calendario: Actual365Fixed
    - Índice: EURIBOR6M
    - Curva Forward: PRUEBA
    - Flujos
- Summary:**
  - Valorar Simular NPV: 27065.9 Fair Rate: 0.0968202 Fair Spred: -0.0972128
  - Save Cancel

A calendar for May 2015 is visible on the right, with the date 19 highlighted.

Y un ejemplo de European Option:

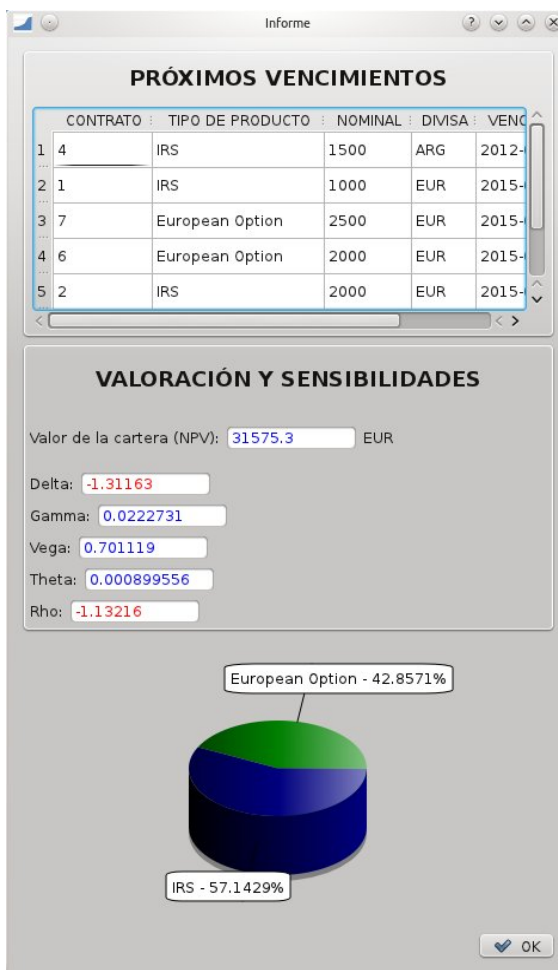
The screenshot shows the 'EuropeanOption' configuration window. It is divided into several sections:

- Información de contrato:**
  - Familia:
  - Tipo de producto: EuropeanOption
  - Contrato: 5
  - Fecha de contratación: 31/05/12 (Desde calendario)
  - Fecha de expiración: 17/05/16 (Desde calendario)
  - Nominal: 10002,00000000 Divisa: EUR
- Información de opción:**
  - Tipo de opción: Put
  - Tipo de subyacente: Deposito Subyacente: d1m Precio spot: 0,03720000
  - Strike: 1,26750000 Tipo libre de riesgo: 0,05000000 Volatilidad (%): 15,00000000
  - Convención de calendario: Actual360
- Summary:**
  - Valorar NPV: Elasticidad: Save Cancel

A calendar for June 2012 is visible on the right, with the date 5 highlighted.

- La segunda opción “informe” se usará para generar un informe de la situación actual de la cartera del usuario. El aspecto es el siguiente:





### 3.2 Almacenamiento de DB.

Se elige Mysql por la excelente difusión que posee en la comunidad, la estandarización de la misma y la multitud de APIs disponibles para la interconexión con ella. Otra de las características determinantes es la excelente interconexión con QT a través de libqt4-sql-mysql.

Después de un exhaustivo análisis de las entidades y relaciones entre ellas existentes en nuestro modelo de datos se diseña la base de datos detallada en los siguientes puntos 3.2.1 y 3.2.2.



### 3.2.1 Tablas

El modelo de datos ha sido normalizado y es descrito a continuación:

#### 3.2.1.1 Estructura de tabla para la tabla calendarios

La tabla de calendarios representa el calendario de una divisa. Se podrán añadir o eliminar festivos.

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>CALENDAR_ID</u>	varchar(50)	No		Identificador de calendario
ADDED_HOLIDAYS	mediumtext	No		Campo serializado de festivos añadidos
REMOVED_HOLIDAYS	mediumtext	No		Campo serializado de festivos eliminados

#### 3.2.1.2 Estructura de tabla para la tabla convenciones\_calendario

Representa las distintas convenciones que hay en el mercado para contar días que contiene un año. En zonas como por ejemplo en Brasil, un año tiene 252 días financieros y son estos los que se tienen en cuenta a la hora de realizar los cálculos pertinentes.

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>ID</u>	varchar(50)	No		Identificador del contador estándar

#### 3.2.1.3 Estructura de tabla para la tabla convenciones\_festivos

Contiene las diferentes convenciones de mercado para reaccionar ante un festivo. Si una fecha de flujo cae por ejemplo en domingo, esta convención especifica si el flujo se pasa al lunes o al viernes por ejemplo

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>ID</u>	varchar(50)	No		Identificador
DESCRIPCION	varchar(200)	No		Comentario sobre el modo de operar del elemento

### 3.2.1.4 Estructura de tabla para la tabla curvas

Representa la información de una curva que será usada en la valoración para descontar precios futuros a valor presente o bien para estimar precios en el futuro.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No		Identificador
CURRENCY_ISO	varchar(3)	No	divisas -> CURRENCY_ISO	Divisa de la curva
TOLERANCIA	double	No		Margen de error en la calibración
CONVENCION_CALENDARIO	varchar(50)	No	convenciones_calendario -> ID	Contador de días asociado

### 3.2.1.5 Estructura de tabla para la tabla curva\_producto

Usada para establecer la relación de los productos líquidos de mercado que componen una curva.

Columna	Tipo	Nulo	Predeterminado	Enlaces a
ID_CURVA	varchar(50)	No		curvas -> ID
ID_PRODUCTO	varchar(50)	No		productos -> ID

### 3.2.1.6 Estructura de tabla para la tabla deals

Contiene la información común a todos los contratos de productos.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		Identificador de contrato
PRODUCT	varchar(50)	No	tipo_producto -> ID	Tipo de producto
SETTLEMENT	date	No		Fecha de inicio de contrato
MATURITY	date	No		Fecha de expiración de contrato
CURRENCY_ISO	varchar(3)	No	divisas -> CURRENCY_ISO	Divisa del nocional
NOMINAL	int(15)	No		Montante del contrato
BUY_SELL	enum('Pagador', 'Recibidor')	No		Si se es vendedor o comprador
SPREAD	double	No		Margen adicional definido por la parte

### 3.2.1.7 Estructura de tabla para la tabla deals\_leg

Tabla que relaciona la pertenencia de una pata a un contrato. Por ejemplo un Swap de tipos tiene dos patas (2 partes) una pagadora fija y otra pagadora variable. Así pues este Swap tendrá 2 entradas, una para cada pata.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID_DEAL</u>	int(10)	No		deals -> ID	Identificador de contrato
<u>ID_LEG</u>	int(10)	No		leg -> ID	Identificador de pata

### 3.2.1.8 Estructura de tabla para la tabla depósitos

Representa los datos específicos de un producto de tipo depósito líquido de mercado. Usado para construir curvas.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No	productos -> ID	Identificador del producto
PLAZO	int(2)	No		Número de periodos
PERIODO	enum('Dia', 'Semana', 'Mes', 'Anio')	No		Unidad de tiempo que se multiplicará por plazo.

### 3.2.1.9 Estructura de tabla para la tabla divisa\_defecto

Representará la divisa elegida para la presentación de cantidades en informes.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>CURRENCY_ISO</u>	varchar(3)	No		divisas -> CURRENCY_ISO	Divisa por defecto

### 3.2.1.10 Estructura de tabla para la tabla divisas

Representará los datos relevantes de una divisa.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>CURRENCY_ISO</u>	varchar(3)	No		Identificador internacional de divisa “EUR”
NAME	varchar(50)	No		Nombre largo de divisa
F_REFERENCE	varchar(50)	No		Indice de referencia
SHORT_CONVENTION	varchar(50)	No	convenciones_calendario -> ID	Convención de calendario a corto plazo
LONG_CONVENTION	varchar(50)	No	convenciones_calendario -> ID	Convención de calendario a largo plazo
LONG_SCHEDULE	varchar(50)	No	convenciones_festivos -> ID	Convención de festivos
SPOT_SCHEDULE	int(2)	No		Fecha valor de contrato (EUR=2, CAD=1,USD 2, ...)
CALENDAR_ID	varchar(50)	No	calendarios -> CALENDAR_ID	Identificador del calendario de festivos

### 3.2.1.11 Estructura de tabla para la tabla european\_option

Representa los datos específicos de contrato de una opción europea.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		Identificador de boleta
TIPO_SUBYACENTE	varchar(50)	No	tipo_producto_mercado -> ID	Tipo de producto subyacente

SUBYACENTE	varchar(50)	No	productos -> ID	Identificador del producto subyacente
STRIKE	double	No		Precio de ejecución de la opción
TIPO_LIBRE	double	No		Tipo libre de riesgo
VOLATILIDAD	double	No		Volatilidad asociada al subyacente
CONVENCION_CALENDARIO	varchar(50)	No	convenciones_calendario -> ID	Base de calendario

### 3.2.1.12 Estructura de tabla para la tabla familias

Representa las familias de los productos que serán contratados por el usuario

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>ID</u>	varchar(50)	No		Nombre de la familia correspondiente

### 3.2.1.13 Estructura de tabla para la tabla familia\_tipo\_producto

Relación entre cada tipo de producto y su familia asociada.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID_FAMILIA</u>	varchar(50)	No		familias -> ID	Identificador de familia
<u>ID_PRODUCTO</u>	varchar(50)	No		tipo_producto -> ID	Identificador de producto

### 3.2.1.14 Estructura de tabla para la tabla fixings

Tabla de fijaciones. Usada para anotar valores de referencia de un índice en fecha pasada. Necesario para la valoración continuada en el tiempo de un producto.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>INDICE_ID</u>	varchar(50)	No		indices -> ID	Identificador de Índice al que pertenece el fix.
<u>DATE</u>	date	No			Fecha para la fijación
FIXING	double	No			Valor de la fijación

### 3.2.1.15 Estructura de tabla para la tabla fras

Representa los datos específicos de un producto de tipo fra (forward rate agreement) líquido de mercado. Usado para construir curvas.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No		productos -> ID	Identificador del Fra
MONTHSTOSTART	int(10)	No			Meses para el inicio del contrato
MONTHSTOEND	int(10)	No			Meses para el final del contrato

### 3.2.1.16 Estructura de tabla para la tabla frecuencias

Posibles frecuencias de pago.

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>ID</u>	varchar(50)	No		Identificador de frecuencia
DESCRIPCION	varchar(200)	No		Descripción del plazo indicado en la frecuencia

### 3.2.1.17 Estructura de tabla para la tabla futuros

Representa los datos específicos de un producto de tipo futuro líquido de mercado. Usado para construir curvas.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No		productos -> ID	Identificador del futuro
MONTHS	int(10)	No			Meses de duración del futuro
IMM_DATE	date	No			Fecha oficial de contratación del futuro

### 3.2.1.18 Estructura de tabla para la tabla índices

Representa los índices definidos por el usuario.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No		Identificador del índice
INDICE_ESTANDAR	varchar(50)	No	indices_estandar -> ID	Índice estándar al que hace referencia
PLAZO	int(10)	No		Numero de periodos
PERIODO	enum('Dia', 'Semana', 'Mes', 'Anio')	No		Cantidad de tiempo para multiplicar por plazo

### 3.2.1.19 Estructura de tabla para la tabla indices\_estandar

Índices convencionales de mercado.

Columna	Tipo	Nulo	Predeterminado	Comentarios
---------	------	------	----------------	-------------



<u>ID</u>	varchar(50)	No	Identificador de Índice
DESCRIPCION	varchar(200)	No	Especificación del índice de mercado al que hace referencia

### 3.2.1.20 Estructura de tabla para la tabla irs

Datos específicos de un tipo de producto IRS.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		deals -> ID	Identificador de DEAL-IRS
DEAL_ID	int(10)	No		deals -> ID	Identificador de DEAL
IS_FWDSTART	tinyint(1)	No			Bool indicando si es contrato aplazado futuro.
FWDSTART	date	No			Fecha futura de inicio.

### 3.2.1.21 Estructura de tabla para la tabla leg

Tabla que contiene los datos genéricos de una pata de un producto.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		Identificador de la pata
FRECUENCIA	varchar(50)	No	frecuencias -> ID	Frecuencia de pago de la pata
TIPO	enum('Fija', 'Flotante')	No		Tipo de pago (interés fijo, interés variable)
CONVENCION	varchar(50)	No	convenciones_festivos -> ID	Qué hacer si un flujo cae en festivo
CONTADOR	varchar(50)	No	convenciones_calendario -> ID	Cómo deben contarse los días de un año
CURVA_ID	varchar(50)	No	curvas -> ID	Identificador de curva (descuento-forward)

### 3.2.1.22 Estructura de tabla para la tabla *leg\_fixed*

Datos específicos para patas de tipo fijo (paga un 3% por ejemplo)

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		leg -> ID	Identificador de pata
TIPO_FIJO	double	No			Tipo de interés que paga la pata fija

### 3.2.1.23 Estructura de tabla para la tabla *leg\_floating*

Datos específicos para patas de tipo flotante o variable (paga Euribor)

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	int(10)	No		leg -> ID	Identificador para la pata variable
ID_FLOATING_INDEX	varchar(50)	No		indices -> ID	Indice asociado a la pata variable

### 3.2.1.24 Estructura de tabla para la tabla *productos*

Tipos de productos líquidos de mercado usados para construir y calibrar curvas.

Columna	Tipo	Nulo	Predeterminado	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	No			Identificador de producto.
TIPO	varchar(50)	No			Tipo de producto.
CURRENCY_ISO	varchar(3)	No		divisas -> CURRENCY_ISO	Divisa del producto
QUOTE	double	No			Precio de mercado del prod.

### 3.2.1.25 Estructura de tabla para la tabla swaps

Representa los datos específicos de un producto de tipo swap líquido de mercado. Usado para construir curvas.

Columna	Tipo	Enlaces a	Comentarios
<u>ID</u>	varchar(50)	productos -> ID	Identificador del Swap
PLAZO	int(3)		Numero de periodos
PERIODO	enum('Dia', 'Semana', 'Mes', 'Anio')		Slash de tiempo
FIXED_BUS_CONVENTION	varchar(50)	convenciones_festivos -> ID	Cómo actuar ante un festivo
FIXED_CAL_CONVENTION	varchar(50)	convenciones_calendario -> ID	Cómo contar los días de un año
FIXEDLEG_FRECUENCY	varchar(50)	frecuencias -> ID	Frecuencia de la pata fija
FLOATING_INDEX	varchar(50)	indices -> ID	Indice asociado.

### 3.2.1.26 Estructura de tabla para la tabla tipos\_cambio

Representa las asociaciones de divisas para establecer el tipo de cambio entre ellas.

Columna	Tipo	Nulo	Enlaces a	Comentarios
<u>DOMESTIC_ID</u>	varchar(50)	No	divisas -> CURRENCY_ISO	Identificador de divisa local
<u>FOREIGN_ID</u>	varchar(50)	No	divisas -> CURRENCY_ISO	Identificador de divisa foránea
QUOTE	double	No		Tipo de cambio.

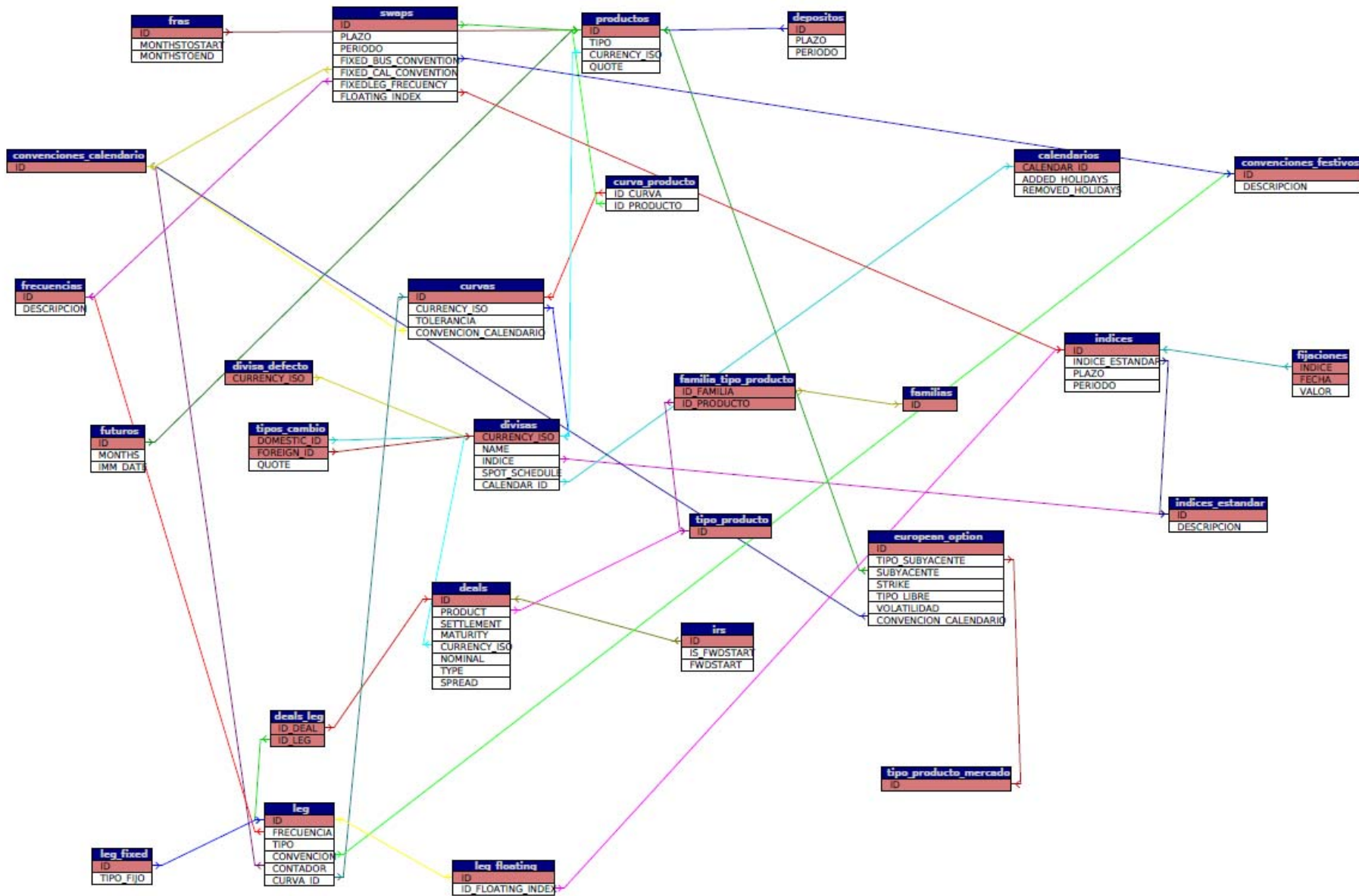
### 3.2.1.27 Estructura de tabla para la tabla tipo\_producto

Tabla para almacenar los tipos de producto que el usuario podrá valorar y simular

Columna	Tipo	Nulo	Predeterminado	Comentarios
<u>ID</u>	varchar(50)	No		Identificador del tipo de producto.

### 3.2.2 Esquema entidad-relación

El gráfico entidad relación es el siguiente:



### 3.3 Motor de valoración.

Se localiza un proyecto open source llamado quantlib, lo podemos encontrar en <http://quantlib.org>

Después de un exhaustivo análisis cumple exactamente con el análisis realizado sobre productos y datos de mercado necesarios, al fin y al cabo la metodología para realizar este tipo de valoraciones es bastante similar y estándar. Sólo cambian ciertos matices entre implementaciones diferentes y son referentes al suavizado de curvas y gestión periférica a la valoración. Por tanto se decide usar esta librería de valoración y no rehacer un código que ya existe.

La forma de trabajar se basa en el wrapping lógico de funciones y constantes de la librería para hacer una correspondencia directa a los datos que nos aplican dentro de Finanval.

## 4 DESARROLLO

### 4.1 Ficheros del aplicativo.

El aplicativo como bien se ha comentado en puntos anteriores, ha sido desarrollado íntegramente en QT Creator. Así pues, el método de compilación usado es qmake.

A continuación se exponen los módulos desarrollados con una breve descripción de su cometido.

#### 4.1.1 Módulos de proyecto

- Finanval.pro y Finanval.pro.user:  
Ficheros que contienen la configuración del proyecto. Desde los directorios donde buscar ficheros de cabeceras pasando por librerías dinámicas necesarias hasta flags de configuración.

#### 4.1.2 Módulos c++, cabeceras y ficheros de definición de pantalla ui

Los siguientes módulos incluyen un fichero .h de cabecera y otro .cpp con el código asociado, así mismo la mayoría tienen asociado un fichero con extensión ui, este tipo de fichero no es más que un xml con la descripción de componentes qt que se incluirán en la pantalla. A continuación se describe la funcionalidad de cada uno:

- aboutwindow  
Fichero de cabecera y código para gestionar la opción del menú ayuda acerca de.
- calendariowindow  
Gestión de calendarios. Muestra los calendarios existentes y permite agregar o eliminar festivos.

- **curvaswindow**  
Gestión de curvas, permite visualizar las curvas existentes, agregar nuevas y eliminar aquellas que ya no se van a usar.
- **DBTools**  
Este módulo carece de fichero asociado de interfaz gráfica. Es un módulo de apoyo y hace de wrapper de acceso a la base de datos. Contiene todas las funciones necesarias por el resto de módulos del aplicativo para consulta y almacenamiento en DB.
- **dealswindow**  
Habilita la funcionalidad para visualizar los contratos del usuario. Permite insertar, borrar y modificar aquellos seleccionados por el usuario.
- **depositoswindow**  
Gestión de productos líquidos de mercado de tipo depósito. Muestra los existentes y permite agregar, eliminar y modificar el market quote asociado.
- **divisaswindow**  
Permite la inserción de divisas que podrán ser usadas en el aplicativo. Así mismo permite eliminar aquellas que el usuario estime que ya no son necesarias.
- **Europeanoptionwindow**  
Pantalla de pricing de una Opción Europea. Muestra todos los datos que componen este producto y permite su valoración. También es posible modificar cualquier componente del producto y salvarlo.
- **fijacioneswindow**  
Cuando se valora un producto que tiene asociado un índice, si la fecha de valoración es posterior a la fecha de contratación, es muy posible que se necesiten fixings. Un fixing es una anotación de lo que valía el índice un día concreto. Así pues, para realizar las valoraciones se estimarán los precios futuros del índice con una curva forward y se recogerán las correspondientes fijaciones pasadas.
- **flujoswindow**  
En productos que periodifican un pago periódico, las fechas de los pagos es calculada atendiendo a las convenciones de calendario y de festivos indicadas. El resultado del cálculo de fechas de pago se puede consultar en la pantalla asociada a este módulo.
- **fraswindow**  
Gestión de productos líquidos de mercado de tipo forward rate agreement. Muestra los existentes y permite agregar, eliminar y modificar el market quote asociado.
- **futuroswindow**  
Gestión de productos líquidos de mercado de tipo futuro. Muestra los existentes y permite agregar, eliminar y modificar el market quote asociado.
- **indiceswindow**  
Habilita la inserción y borrado de índices que serán usados a la postre en las valoraciones de los contratos en Finanval. Estos índices siempre contendrán un índice subyacente ya existente en el sistema y sobre el cual, se indicará el plazo y periodo de fijación del índice.

- **informewindow**  
Es el módulo que hace un estudio completo de la cartera. Muestra los próximos vencimientos, el cálculo del valor presente de nuestra cartera así como de las sensibilidades que tenemos asociadas. Además muestra un gráfico del porcentaje en la composición por productos de nuestra cartera.
- **lrswindow**  
Pantalla de pricing de un Interest Rate Swap. Muestra todos los datos que componen este producto y permite su valoración. También es posible modificar cualquier componente del producto y salvarlo.
- **mainwindow**  
Gestión de pantalla principal. Incluye todo el control del menú principal así como del visionado de nuestros contratos ordenados por tipologías. Además habilita la posibilidad de realizar valoraciones de familias, subfamilias o directamente productos finales.
- **myqframe**  
Fichero de cabecera necesario para la redefinición de la clase QFrame que es usada en informewindow para mostrar el gráfico.
- **nightcharts**  
Módulo fuente externo (h y cpp) que provee el código para la construcción del gráfico de tarta.
- **nuevacurvawindow**  
Pantalla para la gestión de creación de una nueva curva. Habilitará la inserción en ella de todos los productos líquidos de mercado disponibles en el sistema.
- **swapswindow**  
Gestión de productos líquidos de mercado de tipo depósito. Muestra los existentes y permite agregar, eliminar y modificar el market quote asociado.
- **tiposcambiowindow**  
Control de Inserción y borrado de tipos de cambio entre divisas así como la elección de la divisa principal que será usada para mostrar todos los informes y valoraciones del sistema.
- **utils**  
Módulo sin interfaz gráfica que habilita funciones de apoyo y redefiniciones de clase para control de input de datos, etc.
- **valoración**  
Módulo sin interfaz gráfica encargado de proveer las funciones necesarias para llevar a cabo el cálculo de los contratos que se encuentran en el sistema.



## 5 PRUEBAS

Las pruebas que se han realizado en el aplicativo han pasado por ejecutar cada una de las acciones que son posibles en el aplicativo incluyendo valores límite en los campos y siguiendo todas las posibilidades de workflow posibles. Estas pruebas se realizan corriendo la aplicación con valgrind (<http://valgrind.org>). Valgrind es una muy potente herramienta para detectar leaks de memoria, accesos indebidos a memoria, accesos a variables no inicializadas, etc. Este aplicativo no instrumentaliza la aplicación a chequear por lo que no se aliena el comportamiento del aplicativo. El resultado una vez finalizado todo el proceso de depuración y una vez subsanados los problemas encontrados en código se obtiene el siguiente resultado:

```

==25092== LEAK SUMMARY:
==25092==   definitely lost: 2,620 bytes in 17 blocks
==25092==   indirectly lost: 432,912 bytes in 1,980 blocks
==25092==   possibly lost: 9,822 bytes in 124 blocks
==25092==   still reachable: 199,576 bytes in 3,044 blocks
==25092==   suppressed: 0 bytes in 0 blocks
==25092== Reachable blocks (those to which a pointer was found) are not shown.
==25092== To see them, rerun with: --leak-check=full --show-reachable=yes
==25092==
==25092== For counts of detected and suppressed errors, rerun with: -v
==25092== Use --track-origins=yes to see where uninitialised values come from
==25092== ERROR SUMMARY: 51 errors from 33 contexts (suppressed: 223 from 16)
    
```

El listado de los fuentes y funciones donde se encuentran los leaks es bastante extenso (decenas de páginas), no es el objetivo incluir en esta memoria todo el detalle pero los leaks y problemas encontrados residen en librerías de terceros como quantlib y qt, debido a la gran extensión de uso en la comunidad, especialmente de esta última librería, entendemos que podemos catalogar estos problemas como menores y no críticos, además el objetivo de subsanar esos errores queda totalmente fuera de scope.

## 6 MANUAL DE INSTALACIÓN Y USO

### 6.1 Instalación

Finanval se distribuye en un paquete comprimido con nombre Finanval.zip.

Se facilita un método de instalación automático para plataformas Ubuntu-i386. Para aquellas otras plataformas será necesaria una compilación manual del aplicativo. Es necesario notar, que para esta versión, Finanval se provee compilada (para el proceso automático de instalación) con símbolos de debug para facilitar posibles tareas de depuración del tutor. Además se provee con un set de datos de mercado, curvas y contratos de ejemplo que el usuario (en este caso el tutor) puede usar para comprobar el funcionamiento del aplicativo. Para una release pública evidentemente se proveerá con un set de datos vacío.

Pasamos a detallar cada uno de los dos procesos.

#### 6.1.1 Instalación automática

Una vez descomprimido el fichero Finanval.zip, encontramos una carpeta con nombre Finanval.install. Es necesario entrar en esa carpeta donde se localizan los siguientes ficheros:

```
drwxrwxr-x 2 root root 4,0K 2012-06-07 20:14 ayuda_archivos
-rwxr-xr-x 1 root root 803 2012-06-07 20:15 ayuda.htm
drwxrwxr-x 2 root root 4,0K 2012-06-07 20:14 Finanval_src
-rwxr-xr-x 1 root root 803 2012-06-07 20:15 install.sh
-rw-r--r-- 1 root root 80 2012-06-07 20:16 crea_user.sql
-rw-r--r-- 1 root root 25 2012-06-07 20:17 crea_database.sql
-rw-r--r-- 1 root root 31K 2012-06-07 20:23 crea_finanval.sql
-rwxr-xr-x 1 root root 22M 2012-06-07 20:26 Finanval
```

Para realizar la instalación automática es necesario contemplar 2 casos:

- Ya tenemos instalado mysql en el sistema y tenemos asociada clave al usuario root, en este caso será necesario editar el fichero install.sh y añadir nuestra clave al final de la línea habilitada para ello (export PASS=), por ejemplo:  
export PASS=MI\_CLAVE
- Si no tenemos instalado mysql no es necesario realizar ninguna modificación.

Ahora ya estamos preparados para realizar la instalación. Ejecutamos desde una consola root:

```
./install.sh
```

Se instalarán automáticamente las dependencias del sistema y se copiará el ejecutable Finanval en la carpeta `/usr/local/bin`. La salida que observaremos es:

*Instalando Quantlib...*

*Instalando QT...*

*Comprobando mysql...*

*Instalando DB*

*Instalacion finalizada.*

Y ya tenemos Finanval listo para valorar.

### 6.1.2 Instalación Manual

Es necesario instalar en el sistema las dependencias de Finanval, por ejemplo con `apt-get`. Estas son:

- `libquantlib-1.1`
- `libquantlib0-dev`
- `libqt4-sql`
- `libqtgui4`
- `libqtcore4`
- `mysql-server`

Una vez instaladas, es necesario instalar QT-Creator si no lo tenemos ya instalado en el sistema.

A continuación abrimos el proyecto que encontraremos en `Finanval_src/Finanval.pro` de nuestra carpeta descomprimida `Finanval.install`. Ejecutamos una compilación completa y conseguimos nuestro binario.

Es necesario crear la base de datos, para ello ejecutaremos los siguientes comandos sustituyendo la palabra `PASS` por la clave del usuario `root` (`mysql`) si es que existe alguna, en caso contrario eliminamos `-pPASS` de los comandos:

```
mysql -uroot -pPASS mysql < ./crea_user.sql
```

```
mysql -uroot -pPASS < ./crea_database.sql
```

```
mysql -uroot -pPASS finanval < ./crea_finanval.sql
```

Finalmente copiamos los archivos que hay en la carpeta de instalación `Finanval.install` del siguiente modo (desde dentro de la carpeta):

```
mkdir /usr/share/doc/Finanval
```

```
cp -R ./Finanval.install/ayuda* /usr/share/doc/Finanval
```

Y copiamos nuestro recién compilado binario “Finanval” en `/usr/local/bin`

Listo, ya tenemos Finanval preparada para valorar.

## 6.2 Manual de Usuario

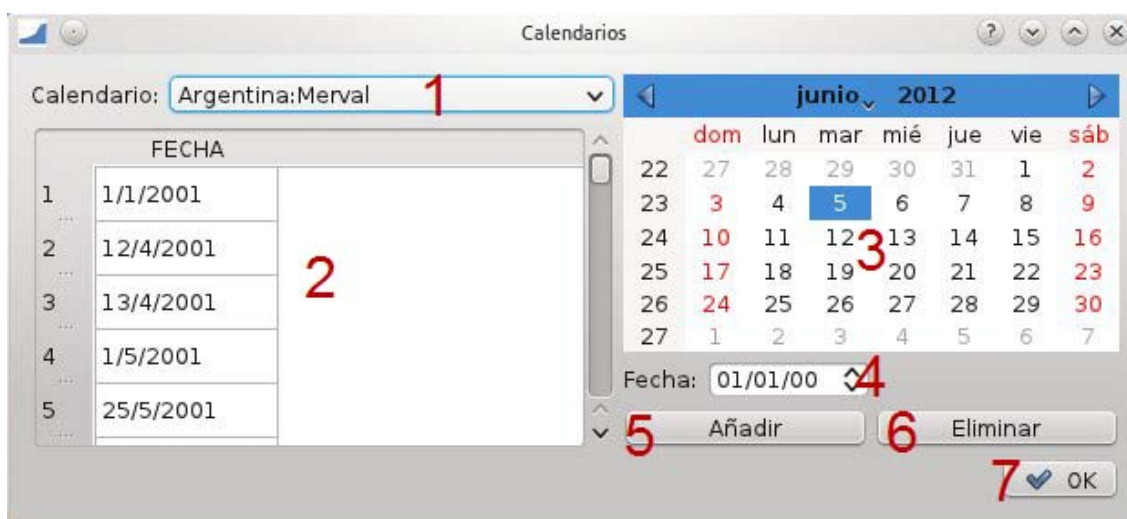
### 6.2.1 Configuración

Para valorar un producto financiero, es necesario disponer de unos datos de base sobre los que realizar los cálculos, estos son por ejemplo divisas, tipos de cambio entre ellas, índices que serán usados para los cálculos relacionados con pagos indexados a un tipo de interés variable, calendarios con sus festivos correspondientes, bases de cálculo disponibles, convenciones para definir cómo actuar en caso de pagos en festivos, etc.

Muchos de estos datos ya están previstos en el sistema. Estos son los que tienen un carácter más estático, por ejemplo, bases de cálculo, calendarios mundiales e índices mundiales más usados. Con este pool de datos disponible es fácil poder valorar casi todos los productos típicos de un mercado financiero. Sin embargo, para comenzar a usar Finanval es necesario configurar el resto de datos de referencia y que están más ligados a las preferencias del usuario. En las 5 secciones que encontramos a continuación se muestra cómo configurar este tipo de datos.

#### 6.2.1.1 Calendarios

Los calendarios definen los días laborables de la zona en la que cotiza el subyacente que se pretende valorar. La pantalla de configuración de los mismos es la siguiente:

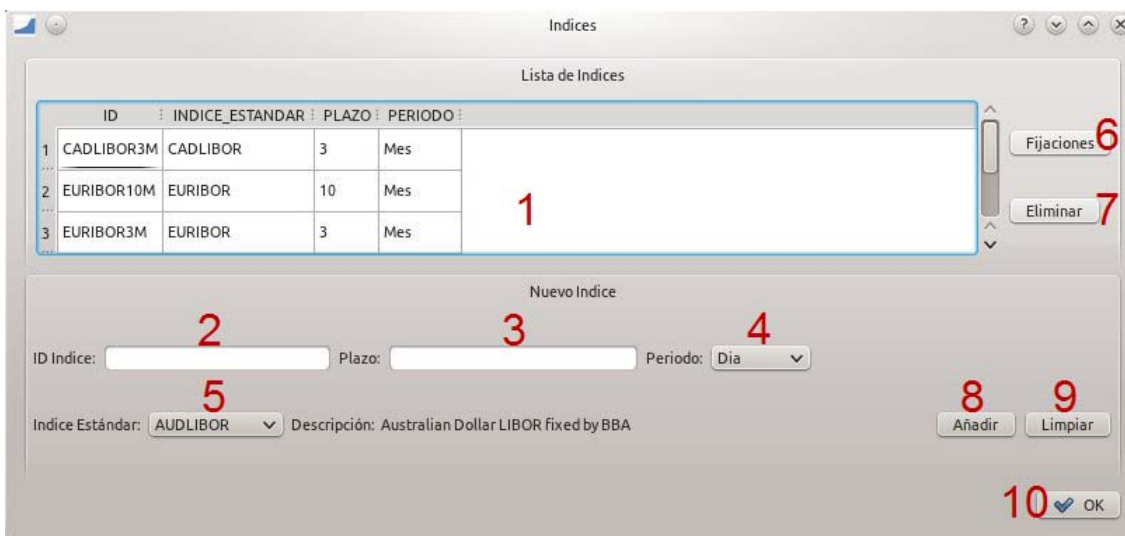


1. Seleccionamos el calendario que queremos configurar. El sistema tiene previstos los siguientes calendarios: Argentina::Merval, Australia, Brazil::Exchange, Brazil::Settlement, Canada::Settlement, Canada::TSX, China, CzechRepublic::PSE, Denmark, EUR, Finland, GBP, Germany::Eurex, Germany::FrankfurtStockExchange, Germany::Settlement, Germany::Xetra, HongKong::HKEx, Hungary, Iceland::ICEX, India::NSE, Indonesia::BEJ, Indonesia::JSX, Italy::Exchange, Italy::Settlement, Japan, LONDON,

- Mexico::BMV, NewZealand, Norway, Poland, SaudiArabia::Tadawul, Singapore::SGX, Slovakia::BSSE, SouthAfrica, SouthKorea::KRX, Sweden, Switzerland, TARGET, Taiwan::TSEC, Turkey, Ukraine::USE, UnitedKingdom::Exchange, UnitedKingdom::Metals, UnitedKingdom::Settlement, UnitedStates::GovernmentBond, UnitedStates::NERC, UnitedStates::NYSE, UnitedStates::Settlement
- En esta tabla aparecerán los festivos previstos para el calendario que esté seleccionado en (1). El sistema posee los festivos predefinidos para casi la completitud de este siglo. Sin embargo, en ocasiones ocurren hechos en los países que hace que se añadan o eliminen calendarios. El resto de los controles de la pantalla están enfocados a contemplar estas situaciones.
  - Este control de calendario nos facilitará la elección de una fecha, si pulsamos sobre una fecha o la seleccionamos manualmente (modificando el mes y el año de la cabecera) esa fecha será automáticamente completada en (4).
  - Elección de la fecha que será añadida o eliminada como festivo del calendario.
  - Se inserta la fecha de (4) como un nuevo festivo en el calendario. Esta configuración será permanente aunque cerremos nuestra aplicación.
  - Elimina la fecha de (4) de la lista de los festivos del calendario. Al igual que 5, esta eliminación se produce de forma permanente.
  - Damos por finalizada la configuración, cerramos la ventana y volvemos al menú principal.

### 6.2.1.2 Índices

Es esta sección es posible configurar los índices que se usarán para valoraciones indexadas. Describimos como hacerlo:



The screenshot shows a window titled 'Indices' with two main sections: 'Lista de Indices' and 'Nuevo Indice'.

**Lista de Indices:** A table with columns: ID, INDICE\_ESTANDAR, PLAZO, PERIODO, and an empty column for actions. The table contains three rows:

ID	INDICE_ESTANDAR	PLAZO	PERIODO		
1	CADLIBOR3M	CADLIBOR	3	Mes	
2	EURIBOR10M	EURIBOR	10	Mes	1
3	EURIBOR3M	EURIBOR	3	Mes	

Buttons 'Fijaciones' (6) and 'Eliminar' (7) are on the right.

**Nuevo Indice:** A form with fields: 'ID Indice:' (2), 'Plazo:' (3), 'Periodo:' (4) with a dropdown menu, 'Indice Estándar:' (5) with a dropdown menu, and 'Descripción:' (8) with the text 'Australian Dollar LIBOR Fixed by BBA'. Buttons 'Añadir' (9) and 'Limpiar' are on the right. An 'OK' button (10) is at the bottom right.

- En esta tabla se mostrarán los índices que ya tenemos configurados en el sistema.

2. Escribiremos el identificador del índice que vamos a configurar. A pesar de ser un campo dependiente del usuario, usualmente se usa el nombre del índice estándar sobre el que hace referencia añadiéndole el plazo de fijación.
3. Establecemos el número de “periodos” (4) que el índice fijará.
4. Indica la unidad de tiempo a la que referencia plazo.
5. Seleccionamos el índice estándar al que referencia. Finanval provee los siguientes: AUDLIBOR, CADLIBOR, CDOR, CHFLIBOR, DKKLIBOR, EURIBOR, EURIBOR365, EURLIBOR, GBPLIBOR, JIBAR, JPYLIBOR, NZDLIBOR, SEKLIBOR, TIBOR, TRLIBOR, USDLIBOR, ZIBOR. A la derecha de este control encontraremos la descripción del índice estándar activo en ese momento.
6. Con este botón abriremos la pantalla de fijaciones. Esta pantalla será comentada a continuación.
7. Eliminaremos el índice marcado en la tabla (1).
8. Añadiremos el índice que esté configurado en los controles 2,3,4 y 5
9. Limpia 2,3,4 y 5 para realizar una nueva inserción.
10. Cierra la pantalla y vuelve al menú principal.

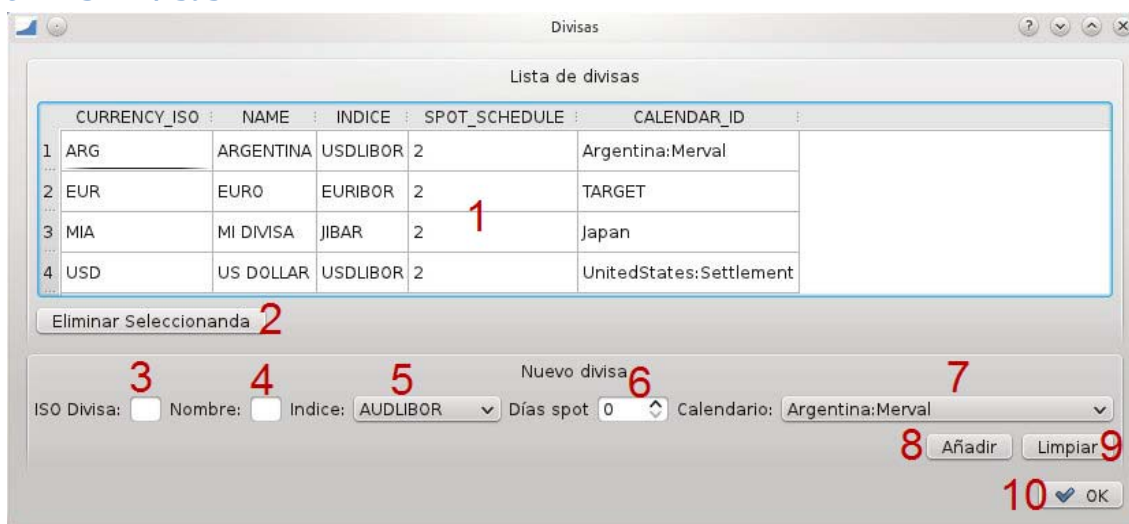
A continuación detallaremos la pantalla de fijaciones a la que hace referencia el control 6, en esta pantalla podremos añadir o eliminar fijaciones.

Cuando realizamos una valoración de un producto indexado, habitualmente si es valorado en su fecha de contratación, todos los flujos que se desprenden de la valoración tienen lugar en el futuro. Sin embargo, una vez contratada la operación, el usuario muy probablemente desee ir valorándola durante la vida de esta. Cuando una valoración requiere de una fecha de flujo pasada en el tiempo no es posible estimarla con una de las curvas, ya que están estimando precisamente datos futuros. Así pues es necesario, establecer cuál era el precio del índice en el pasado. La introducción de fijaciones es una práctica habitual en todas las entidades financieras. Habitualmente al final del día, un sistema automático o manual se encarga de anotar el precio de cierre de todos los índices que la entidad financiera tiene en su cartera.



1. Tabla que muestra las fijaciones existentes.
2. Aunque no es habitual, se habilita un control que borrará una fijación, en concreto aquella seleccionada en la tabla 1.
3. Calendario de ayuda para seleccionar la fecha del control (4). Pulsando sobre una fecha del calendario, esta será trasladada automáticamente a (4).
4. Fecha que será usada para insertar la fijación.
5. Valor de la fijación.
6. Control para añadir la fijación a la base de datos.
7. Cerramos la pantalla de fijaciones y volveremos a la pantalla de índices.

### 6.2.1.3 Divisas



1. Tabla donde se muestran las divisas configuradas en el sistema.
2. Elimina aquella divisa que se encuentre seleccionada en la tabla (1) o la primera si no hay ninguna.
3. Identificador internacional ISO de divisa.
4. Nombre de la divisa.



5. Índice al que referencia. Por ejemplo EUR hace referencia a EURIBOR
6. Días spot de contrato. Cada divisa tiene prefijados por convención un número de días hasta que un contrato sobre esta divisa tiene efecto. Por ejemplo, para EUR son 2 días, para dólar Canadiense es 1, etc...
7. Calendario que será usado para cálculos asociados a esta divisa. Por ejemplo, para EUR, el calendario de referencia es TARGET.
8. Añadimos la divisa configurada en 3,4,5,6 y 7.
9. Limpiamos los campos de inserción para comenzar con una nueva.
10. Cerramos la pantalla y volvemos al menú principal.

#### 6.2.1.4 Curvas

Esta es una de las configuraciones cruciales y que más dependen de la forma de trabajar del usuario. Una curva se define incluyendo puntos básicos en ella que son representados por los productos líquidos que se describen en 6.2.2

Las curvas son la base para poder estimar precios futuros de flujos y poder traer a valor presente esos mismos precios estimados. La pantalla principal de configuración de curvas:



1. Tabla en la que aparecen las curvas definidas en el sistema
2. Tabla en la que aparece el detalle de la curva seleccionada en (1)
3. Insertar una nueva curva al sistema. Se abre la pantalla de inserción para nuevas curvas que estudiamos a continuación.
4. Eliminamos la curva seleccionada en (1), o la primera de (1) en caso de no existir selección.
5. En caso de disponer de conector de datos de mercado que esté refrescando precios en base de datos, con este botón obtendremos la última actualización de los quotes de los mismos.



6. Cierra la pantalla de curvas y vuelve al menú principal.

Vemos el detalle de inserción de nueva curva (3):

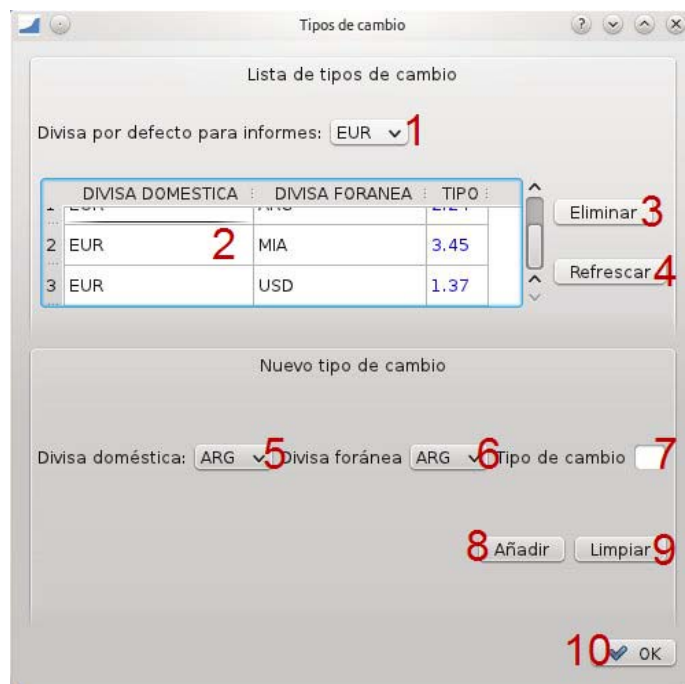


1. Escribimos el nombre que queremos darle a la curva.
2. Seleccionamos la divisa
3. Definimos la tolerancia del proceso iterativo de [Bootstrapping](#). Es decir, el margen de error a la hora de general la curva a partir de los productos.
4. Seleccionamos la base de calendario asociada a la curva. Indica la forma de contar los días laborables en un año.
5. Muestra los productos disponibles para ser añadidos en la curva. Estos productos son los correspondientes a la tipología seleccionada en (7).
6. Muestra los productos que ya han sido insertados (8) para formar parte de la curva.
7. Selecciona la tipología de productos existentes en el aplicativo.
8. Inserta el producto seleccionado en (5) y si no hubiese ninguno seleccionado, el primero.
9. Elimina el producto seleccionado de la tabla (6) o el primero si no hubiese ninguno seleccionado.
10. Cierra la ventana insertando la curva que ha sido definida y volvemos a la pantalla de curvas.
11. Cierra la ventana descartando los cambios producidos y volviendo a la pantalla de curvas.

### 6.2.1.5 Tipos de cambio

En esta pantalla podremos configurar las asociaciones existentes entre las divisas del sistema y la equivalencia entre ellas, esto es, el tipo de cambio entre una divisa y otra.

Además podremos indicar cuál es la divisa por defecto para los informes de Finanval.



1. Divisa por defecto para todos los informes presentados por Finanval. Cuando se realicen valoraciones en otro producto, el resultado se convertirá en esta divisa para ser mostrado.
2. Tabla donde se muestran las asociaciones configuradas en el sistema.
3. Elimina la asociación seleccionada en (2) o la primera de (2) si no hubiese ninguna seleccionada.
4. En caso de tener un conector refrescando los tipos de cambio en DB, pulsando este control obtendremos el último precio.
5. Divisa que jugará el papel de divisa local.
6. Divisa que jugará el papel de divisa foránea.
7. Tipo de cambio de las divisas.
8. Añade el tipo de cambio configurado al sistema.
9. Limpia los campos para preparar una nueva inserción.
10. Cierra la ventana y vuelve al menú principal.

### 6.2.2 Set de productos líquidos

Como hemos descrito en apartados anteriores (análisis y diseño), toda la funcionalidad de Finanval se basa en la estimación de precios de nuestros productos contratados. Para estimar estos precios Finanval se basa en los quotes (o precios) de mercado de productos muy líquidos que es posible contratar en mercados organizados. Estos productos serán a diferentes plazos y observando el tipo de interés (quote) que se ofrece por ellos en los distintos plazos es posible tomar el pulso de lo que vale el dinero en el corto-medio y largo plazo. Este concepto está muy relacionado con el coste de oportunidad, ambos se basan en lo que se ofrece líquidamente por una cantidad de un producto en un plazo predeterminado.

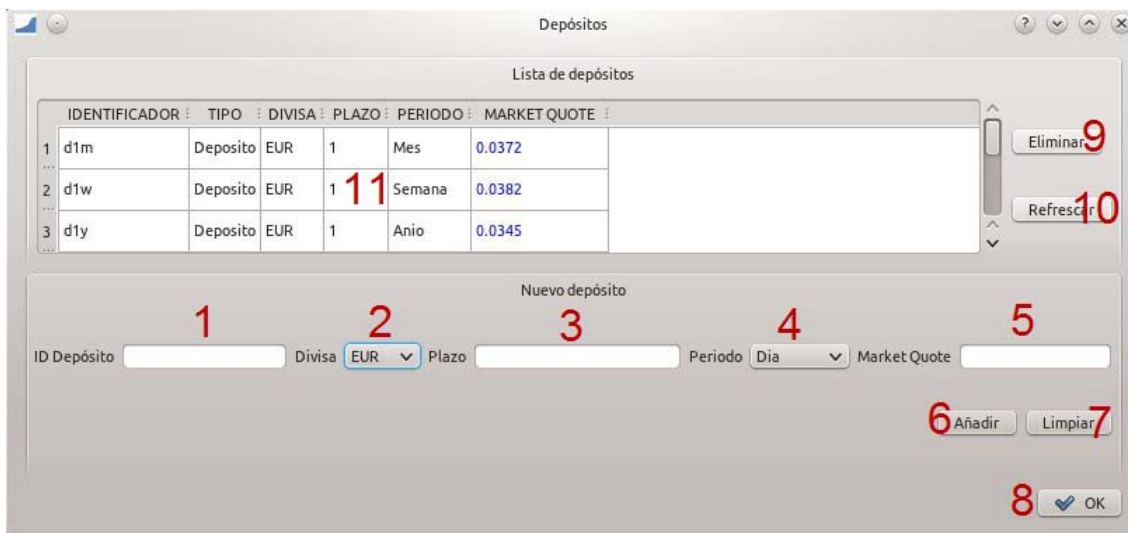
Así pues, lo primero que debemos realizar en Finanval es la inserción de aquellos productos con los que nos sentimos cómodos y que creemos que son muy representativos de lo que el dinero vale en un determinado plazo. Existen 4 tipos estándar de productos que suelen usarse para este fin, depósitos, fras, futuros y swaps. Así pues introduciremos aquellos que nos interesan y que nos van a dar los puntos de una curva (de precio de dinero) hasta el plazo máximo que estimamos vamos a contratar productos que después valoraremos con Finanval. También es cierto que podemos sobrepasar ese plazo y de esta manera obtendremos valoraciones más fidedignas. También es necesario notar, que a más cantidad de productos, más exactas serán las valoraciones que realizará Finanval.

A continuación se expone el modo de inserción de estos productos que podemos encontrar en el menú principal bajo el nombre de “Datos de Mercado”.

### 6.2.2.1 Depósitos

Este tipo de productos suelen ser usados en el corto plazo, habitualmente es complicado ver depósitos a más de 2-3 años. Para ese tipo de periodos se suelen usar los productos que veremos en las siguientes secciones.

Para insertar un nuevo depósito:



The screenshot shows a window titled 'Depósitos' with a table 'Lista de depósitos' and a form 'Nuevo depósito'.

IDENTIFICADOR	TIPO	DIVISA	PLAZO	PERIODO	MARKET QUOTE
1 d1m	Deposito	EUR	1	Mes	0.0372
2 d1w	Deposito	EUR	1	11 Semana	0.0382
3 d1y	Deposito	EUR	1	Anio	0.0345

The 'Nuevo depósito' form has the following fields and controls:

- 1 ID Depósito: text input field
- 2 Divisa: dropdown menu showing 'EUR'
- 3 Plazo: text input field
- 4 Periodo: dropdown menu showing 'Día'
- 5 Market Quote: text input field
- 6 Añadir: button
- 7 Limpiar: button
- 8 OK: button
- 9 Eliminar: button (next to row 1)
- 10 Refrescar: button

1. Insertamos el identificador que queremos asignar. Es un dato privado.
2. Asociamos la divisa que tiene el producto.
3. Incluimos el plazo en el que vence el producto (un número entero positivo)
4. Indicamos el periodo del plazo introducido anteriormente.
5. Insertamos el valor del precio de mercado.
6. Finalmente pulsando el botón añadir veremos que nuestro depósito aparece en la tabla identificada por 11.

7. Si deseamos limpiar los datos de los campos para volver a comenzar este es el botón adecuado.
8. Cierra la pantalla una vez finalizada la inserción y volvemos al menú principal.
9. Eliminaremos el depósito que tenga seleccionada alguna celda en (11), o el primero de la lista (11) en caso de no haber seleccionado ninguno.
10. En caso de disponer de conector de datos de mercado que esté refrescando esos precios en base de datos, con este botón obtendremos la última actualización de los quotes de los mismos.
11. Espacio en la ventana para mostrar los depósitos existentes en el sistema.

### 6.2.2.2 Fras

Al igual que los depósitos, este tipo de producto (Forward Rate Agreement) es también usado en el corto plazo, básicamente es un acuerdo para suscribir un tipo de interés en un futuro inmediato. Aunque es menos común encontrar este tipo de productos siendo usado para “observar” el mercado, Finanval lo incluye porque en algunos círculos (tesorerías de grandes bancos) sí que se suele usar.

Para insertar un nuevo FRA:

1. Insertamos el identificador que queremos asignar. Es un dato privado.
2. Asociamos la divisa que tiene el producto.
3. Incluimos los meses que se estipulan desde la fecha de hoy hasta el inicio del contrato.
4. Incluimos los meses que se estipulan desde la fecha de hoy hasta el final del contrato.
5. Insertamos el valor del precio de mercado.
6. Finalmente pulsando el botón añadir veremos que nuestro FRA aparece en la tabla identificada por 11.

7. Si deseamos limpiar los datos de los campos para volver a comenzar este es el botón adecuado.
8. Cierra la pantalla una vez finalizada la inserción y volvemos al menú principal.
9. Eliminaremos el FRA que tenga seleccionada alguna celda en (11), o el primero de la lista (11) en caso de no haber seleccionado ninguno.
10. En caso de disponer de conector de datos de mercado que esté refrescando los precios en base de datos, con este botón obtendremos la última actualización de los quotes de los mismos.
11. Espacio en la ventana para mostrar los FRAs existentes en el sistema.

### 6.2.2.3 Futuros

Este tipo de productos suele ser usado en el medio plazo. La inserción es bastante similar a la de un FRA:



The screenshot shows a window titled 'Futuros' with a table 'Lista de futuros' and a form 'Nuevo futuro'. Red numbers 1-11 highlight the following elements:

- 1: ID Futuro input field
- 2: Divisa dropdown menu (EUR)
- 3: Meses input field
- 4: Market Quote input field
- 5: IMM Date dropdown menu (2012-6-20)
- 6: Añadir button
- 7: Limpiar button
- 8: OK button
- 9: Eliminar button
- 10: Refresca button
- 11: Row 2 of the table (fut2)

	IDENTIFICADOR	TIPO	DIVISA	MESES	FECHA IMM	MARKET QUOTE
1	fut1	Futuro	EUR	3	2012-06-20	96.2875
2	fut2	Futuro	EUR	3	2012-09-19	96.7875
3	fut3	Futuro	EUR	3	2012-12-10	96.0875

1. Insertamos el identificador que queremos asignar. Es un dato privado.
2. Asociamos la divisa que tiene el producto.
3. Incluimos los meses que dicho futuro estipula como duración del contrato hasta que existe el delivery.
4. Insertamos el valor del precio de mercado.
5. Seleccionamos la fecha de contratación del futuro. Esta fecha es estándar y está definida por [International Monetary Market](#) (IMM). Sólo es posible contratar un futuro en las fechas prefijadas. Estas fechas están calculadas por Finanval en el momento de la inserción del futuro y atendiendo a la fecha actual.
6. Finalmente pulsando el botón añadir veremos que nuestro futuro aparece en la tabla identificada por 11.
7. Si deseamos limpiar los datos de los campos para volver a comenzar este es el botón adecuado.

8. Cierra la pantalla una vez finalizada la inserción y volvemos al menú principal.
9. Eliminaremos el Futuro que tenga seleccionada alguna celda en (11), o el primero de la lista (11) en caso de no haber seleccionado ninguno.
10. En caso de disponer de conector de datos de mercado que esté refrescando los precios en base de datos, con este botón obtendremos la última actualización de los quotes de los mismos.
11. Espacio en la ventana para mostrar los futuros existentes en el sistema.

#### 6.2.2.4 Swaps

Este tipo de productos suelen usarse a largo plazo, pudiendo llegar hasta 20 años. Usualmente, a más plazo no interesa tener definidos puntos en la curva ya que las estimaciones a tan largo plazo tienen una incertidumbre demasiado elevada como para ser tenida en cuenta.

A continuación describimos la inserción de un swap:



The screenshot shows the 'Swaps' application window. At the top, there is a table titled 'Lista de swaps' with columns: IDENTIFICADOR, TIPO, DIVISA, PLAZO, PERIODO, CONV.FESTIVOS, CONV.CALENDARIO, FRECUENCIA FIJA, INDICE, and MARKET QUOTE. The table contains three rows of swap data. Below the table is a form titled 'Nuevo swap' with various input fields and dropdown menus. Red numbers 1 through 13 are overlaid on the interface to indicate the steps for adding a new swap:

- 1: ID Swap input field
- 2: Divisa dropdown menu
- 3: Plazo input field
- 4: Periodo dropdown menu
- 5: Market Quote input field
- 6: Frecuencia tipo fijo dropdown menu
- 7: Convención festivos dropdown menu
- 8: Convención calendario dropdown menu
- 9: Índice variable dropdown menu
- 10: Añadir button
- 11: Limpiar button
- 12: OK button
- 13: Eliminar button

1. Insertamos el identificador que queremos asignar. Es un dato privado.
2. Asociamos la divisa que tiene el producto.
3. Insertamos el plazo para definir el vencimiento del swap, junto con periodo definirá la fecha de expiración del producto.
4. Especifica la unidad de tiempo de plazo.
5. Insertamos el precio que tiene este subyacente en mercado.
6. Especificamos cual será el periodo de pago de la entidad que paga un tipo fijo.
7. Indicamos como proceder si uno de los pagos anteriores cae en festivo.
8. Seleccionamos la base para el cálculo de días anuales.
9. Seleccionamos el índice sobre el cual pagará la entidad que paga en tipo variable. Este índice ya especifica cuál es la frecuencia de pago y demás datos adicionales necesario (Ver configuración de índices).
10. Finalmente pulsando el botón añadir veremos que nuestro swap aparece en la tabla identificada por 15.
11. Si deseamos limpiar los datos de los campos para volver a comenzar este es el botón adecuado.

12. Cierra la pantalla una vez finalizada la inserción y volvemos al menú principal.
13. Eliminaremos el Swap que tenga seleccionada alguna celda en (15), o el primero de la lista (15) en caso de no haber seleccionado ninguno.
14. En caso de disponer de conector de datos de mercado que esté refrescando los precios en base de datos, con este botón obtendremos la última actualización de los quotes de los mismos.
15. Espacio en la ventana para mostrar los Swaps existentes en el sistema.

### 6.2.3 Uso

La principal finalidad de finanval es la valoración de productos de mercado y más en concreto de NPV de estos productos, es decir el valor presente neto. Este concepto está asociado a los rendimientos que quedan por ser percibidos del producto que estamos tratando vistos desde un punto de vista del valor acumulado a precio de hoy.

#### 6.2.3.1 Exportación

En el menú principal archivo->exportar, encontramos esta utilidad. Si procedemos a ejecutarla, obtendremos una pantalla para seleccionar el fichero donde se guardarán todos los productos que tenemos en cartera dentro de Finanval. Esta exportación se hará a fichero de texto plano y en bloques por producto, incluyendo cabeceras de los campos exportados. La principal finalidad es poder abrir estos datos desde una hoja de cálculo realizando importación de campos divididos por separador. Un ejemplo puede ser:

DEAL ID;NOMINAL;SETTLEMENT;MATURITY;DIVISA;TIPO DE OPCION;TIPO DE SUBYACENTE;SUBYACENTE;SPOT;STRIKE;TIPO LIBRE;VOLATILIDAD;CONVENCION DE CALENDARIO

5;1000;jue may 31 2012;mar may 17 2016;EUR;Put;Deposito;d1m;0.0372;1.2675;0.05;15;Actual360

6;2000;lun jun 4 2012;mar jun 30 2015;EUR;Put;Futuro;fut1;96.2875;96.2875;0.05;5;Actual360

7;2500;vie jun 29 2012;lun jun 22 2015;EUR;Put;Futuro;fut2;96.7875;100;0.05;15;Actual360

DEAL ID;NOMINAL;SETTLEMENT;MATURITY;DIVISA;ROL;SPREAD;ISFWD;FWDSTART;FRECUENCIA FIJA;CONVENCION FESTIVOS FIJA;CONVENCION CALENDARIO FIJA; TIPO FIJO; CURVA DE DESCUENTO;FRECUENCIA FLOTANTE;CONVENCION FESTIVOS FLOTANTE;CONVENCION CALENDARIO FLOTANTE;INDICE;CURVA FORWARD

4;1500;lun may 28 2012;miÃ© jun 13  
2012;ARG;Pagador;0;0;;Annual;Following;Actual360;3;ARGENTINA;Annual;Following;Actual360;EURIBOR3M;ARGENTINA

1;1000;lun may 28 2012;jue may 14  
2015;EUR;Pagador;1;0;;Annual;ModifiedFollowing;Actual360;3;DEPO\_SWAP\_EUR;Semiannual;ModifiedPreceding;ActualActual;EURIBOR6M;DEPO\_SWAP\_EUR

2;2000;miÃ© may 30 2012;lun sep 28  
2015;EUR;Pagador;2;0;;Annual;Following;Actual360;2;DEPO\_SWAP\_EUR;Annual;Preceding;ActualActual;EURIBOR3M;DEPO\_SWAP\_EUR

3;3000;miÃ© may 30 2012;lun sep 28 2015;EUR;Pagador;2;1;sÃ¡b oct 27  
2012;Annual;ModifiedFollowing;Actual360;2;DEPO\_SWAP\_EUR;Quarterly;ModifiedPreceding;Actua60;EURIBOR3M;DEPO\_SWAP\_EUR

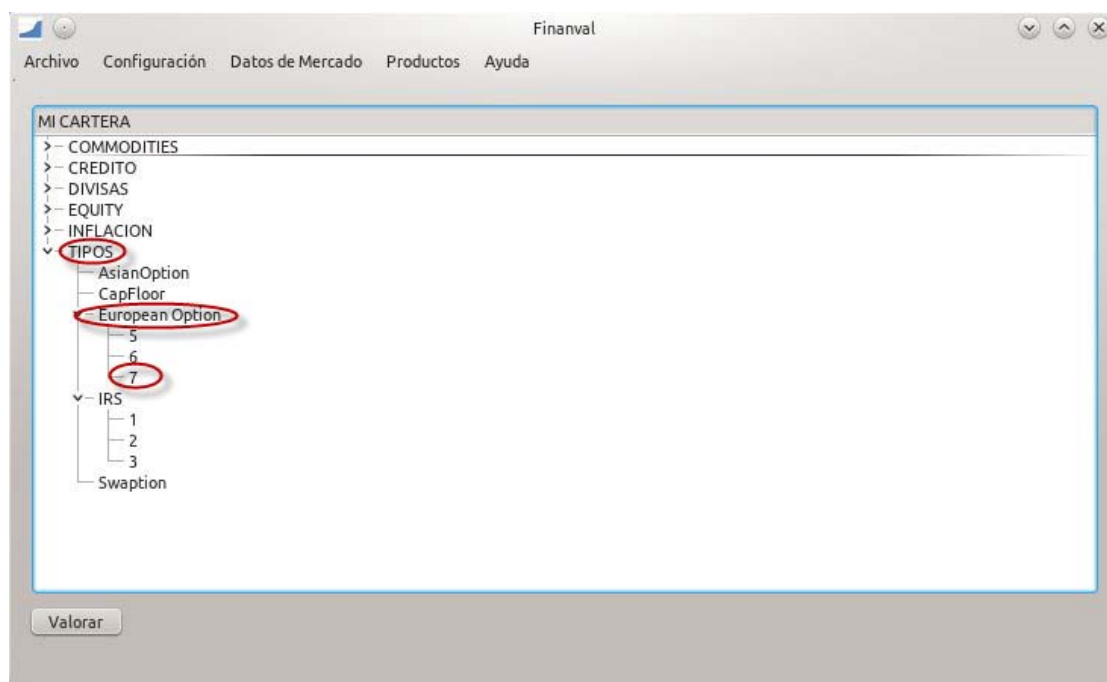


### 6.2.3.2 Valoración

Existen dos vías para poder proceder a la funcionalidad de valoración del aplicativo, la primera es desde el mismo menú principal y la segunda desde el centro de contratos.

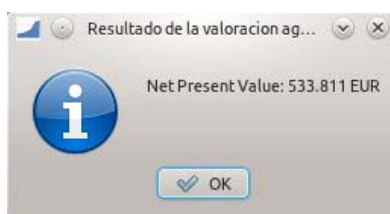
#### 6.2.3.2.1 Valoración a través del menú principal

Es posible valorar productos unitarios o agrupaciones de los mismos, el mecanismo será la valoración de todos los hijos de la rama que esté seleccionada en la pantalla. Lo vemos a continuación:



Si seleccionamos “TIPOS” y pulsamos valorar serán valorados y agregados todos los productos que dependen de la rama, es decir obtendríamos la valoración agregada de (5,6,7,1,2,3).

Sin embargo si seleccionamos European Option la valoración será de (5,6,7) y también sería posible obtener una valoración de deal individual. Por ejemplo seleccionando únicamente el elemento 7. El output que obtendremos será:



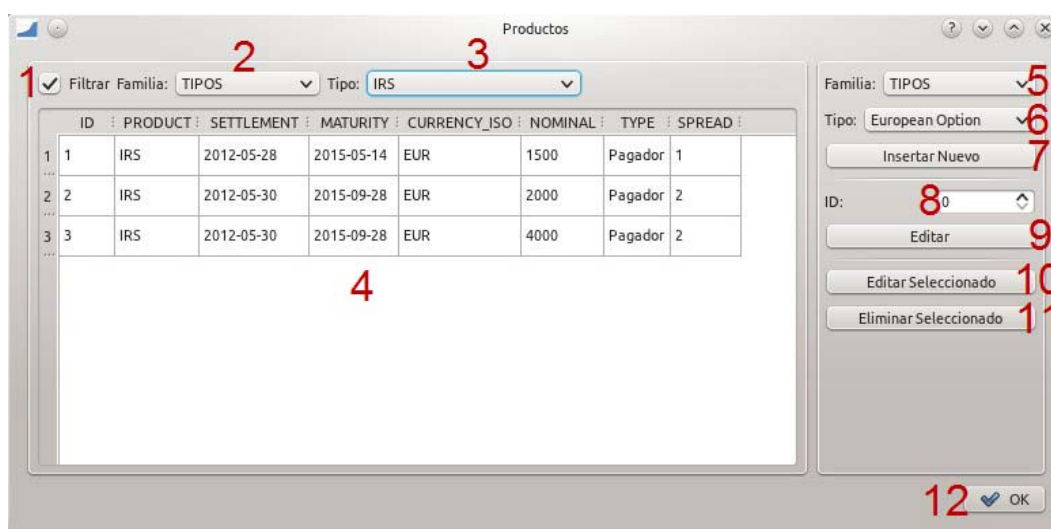
Observamos que se muestra en EUR que es la divisa por defecto seleccionada en la pantalla de configuración de tipos de cambio.



### 6.2.3.2.2 Centro de control de contratos

Desde el menú Productos->editar accedemos al centro de control de contratos. Desde aquí podremos insertar, modificar, eliminar contratos a nuestra cartera. En la versión actual sólo están disponibles dos tipos de contrato: IRS y Opciones Europeas. El motivo es lógico, el desarrollo de un amplio espectro de contratos es trabajo para años y queda fuera del scope de este TFC. No obstante se ha intentado reflejar en los menús y opciones un amplio rango de contratos con la finalidad de que puedan ser implementados por la comunidad ya que el licenciamiento de este proyecto será basado en software libre y por tanto puesto a disposición de la comunidad. Una muy posible opción sería subir el aplicativo a Sourceforge.

De vuelta al centro de contratos:



1. Si seleccionamos este checkbox aparecerán los campos 2 y 3 para poder realizar filtrados de los productos que se muestran en 4. Si no está seleccionado se mostrarán todos los productos.
2. Selecciona la familia de productos a mostrar (4).
3. Selecciona los tipos de productos a mostrar en (4).
4. Tabla que muestra los contratos seleccionados por (2) y (3) o en su defecto todos los contratos del sistema.
5. Selección de la familia para insertar un nuevo producto.
6. Selección del tipo de producto para nueva inserción, por los motivos relatados anteriormente, en esta primera versión sólo están disponibles IRS y Opciones Europeas.
7. Insertar un nuevo producto de familia (5) y tipología (6)
8. Podremos indicar directamente el contrato que queremos editar pulsando el botón (9).
9. Se abre la pantalla del producto seleccionado en (8)

10. Abre la pantalla para el producto seleccionado de la tabla (4) o el primero si no hubiese ninguno seleccionado.
11. Elimina el producto seleccionado de la tabla (4) o el primero si no hubiese ninguno seleccionado.
12. Cierra la pantalla de contratos y vuelve al menú principal.

Como hemos visto es posible trabajar con 2 tipos de contratos, IRS y Opciones Europeas. El usuario conocerá perfectamente cómo manejar los mismos, se detallan las pantallas de edición de cada uno de los contratos:

Para IRS:

The screenshot shows the IRS configuration window with the following details:

- Información de contrato:** Familia: TIPOS; Tipo de producto: IRS; Contrato: 1; Fecha de contratación: 28/05/12; Fecha de expiración: 14/05/15; Nominal: 1500,00; Divisa: EUR; Rol: Pagador; Margen(%): 1,00; Forward Start: .
- Calendar:** junio 2012. The 9th is highlighted in blue.
- Información de patas:**
  - Pata Fija:** Frecuencia: Annual; Convención de festivos: ModifiedFollowing; Convención de calendario: Actual360; Tipo de interés(%): 3,00; Curva de Descuento: DEPO\_SWAP\_EUR.
  - Pata Flotante:** Frecuencia: Semiannual; Convención de festivos: ModifiedPreceding; Convención de calendario: ActualActual; Índice: EURIBOR6M; Curva Forward: DEPO\_SWAP\_EUR.
- Control Bar:** Valorar; NPV: 2425.31; Fair Rate: 0.610128; Fair Spread: -0.572862; Save; Cancel.

1. En esta sección de la pantalla se configuran los datos generales del contrato
2. Con el calendario nos podemos ayudar para seleccionar las fechas de contratación y expiración a través de los controles “desde calendario”. Se copiará en los controles de fechas aquella fecha seleccionada en el calendario.
3. Sección de datos correspondientes a la pata de pago en tipo de interés fijo.
4. Sección de datos correspondientes a la pata de pago variable.
5. Sección de control y output. Podremos valorar el producto en cuyo caso podremos observar los resultados en los campos NPV, Fair Rate y Fair Spread, guardarlo o descartar cambios.

Opción Europea:

1. En esta sección de la pantalla se configuran los datos generales del contrato
2. Con el calendario nos podemos ayudar para seleccionar las fechas de contratación y expiración a través de los controles “desde calendario”. Se copiará en los controles de fechas aquella fecha seleccionada en el calendario.
3. Sección específica de datos para opciones, aquí controlamos los datos específicos de una opción europea que junto con los de la sección (1) conformarán el contrato.
4. Sección de control y output. Podremos valorar el producto en cuyo caso podremos observar los resultados en los campos NPV y Elasticidad, guardarlo o descartar cambios.

### 6.2.3.3 Simulación

Hasta ahora sólo habíamos obtenido valores NPV. En el menú Productos->informe es posible acceder a un informe completo de toda la cartera.

La finalidad es observar que riesgos y sensibilidades existen en nuestra cartera (podemos ver el detalle en 1.2.5).

Si ejecutamos el informe obtendremos la siguiente pantalla:



En la sección **Próximos vencimientos** podemos observar nuestros productos ordenados por orden de vencimiento.

En la sección **Valoración y sensibilidades** observamos el valor presente de nuestra cartera así como el pool de sensibilidades asociadas a ella (detalle en 1.2.5). Esta información es muy útil para la realización de coberturas, etc.

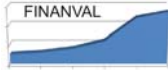
Finalmente observamos un gráfico que desgrena la composición en porcentajes de nuestra cartera.

#### 6.2.3.4 Ayuda

En la sección del menú principal Ayuda podremos acceder a dos apartados:

Contenido, que básicamente muestra información similar al manual de usuario.

Acerca de, muestra los detalles del aplicativo como nombre, versión, fecha y licencia.



## 7 Licencia

El autor Manuel Martínez Liébana, en este trabajo TFC para la UOC que responde al nombre de **Finanval**, licencia este producto bajo **GNU GENERAL PUBLIC LICENSE** en su versión 3 de 29 de Junio de 2007. Para más detalles: [GPL](#)