



FLAME

Flexible Large-scale Agent Modelling Environment

Introducción a FLAME

Ander Ibáñez de Gauna Navazo

¿QUE ES FLAME?

FLAME (Flexible Large-Scale Agent Modelling Environment) es un sistema de modelado basado en agentes genéricos, con los cuales se pueden implementar aplicaciones para diferentes áreas.



The
University
Of
Sheffield.



Science & Technology
Facilities Council

EPSRC

Engineering and Physical Sciences
Research Council

¿CÓMO SURGE FLAME?

- En Junio de 2011, el consejo de investigación de ingeniería y ciencias físicas del Reino Unido financia el proyecto FLAME.
- El primer prototipo de FLAME lo desarrolla la Universidad de Sheffield en colaboración con el laboratorio STFC Rutherford Appleton



The
University
Of
Sheffield.

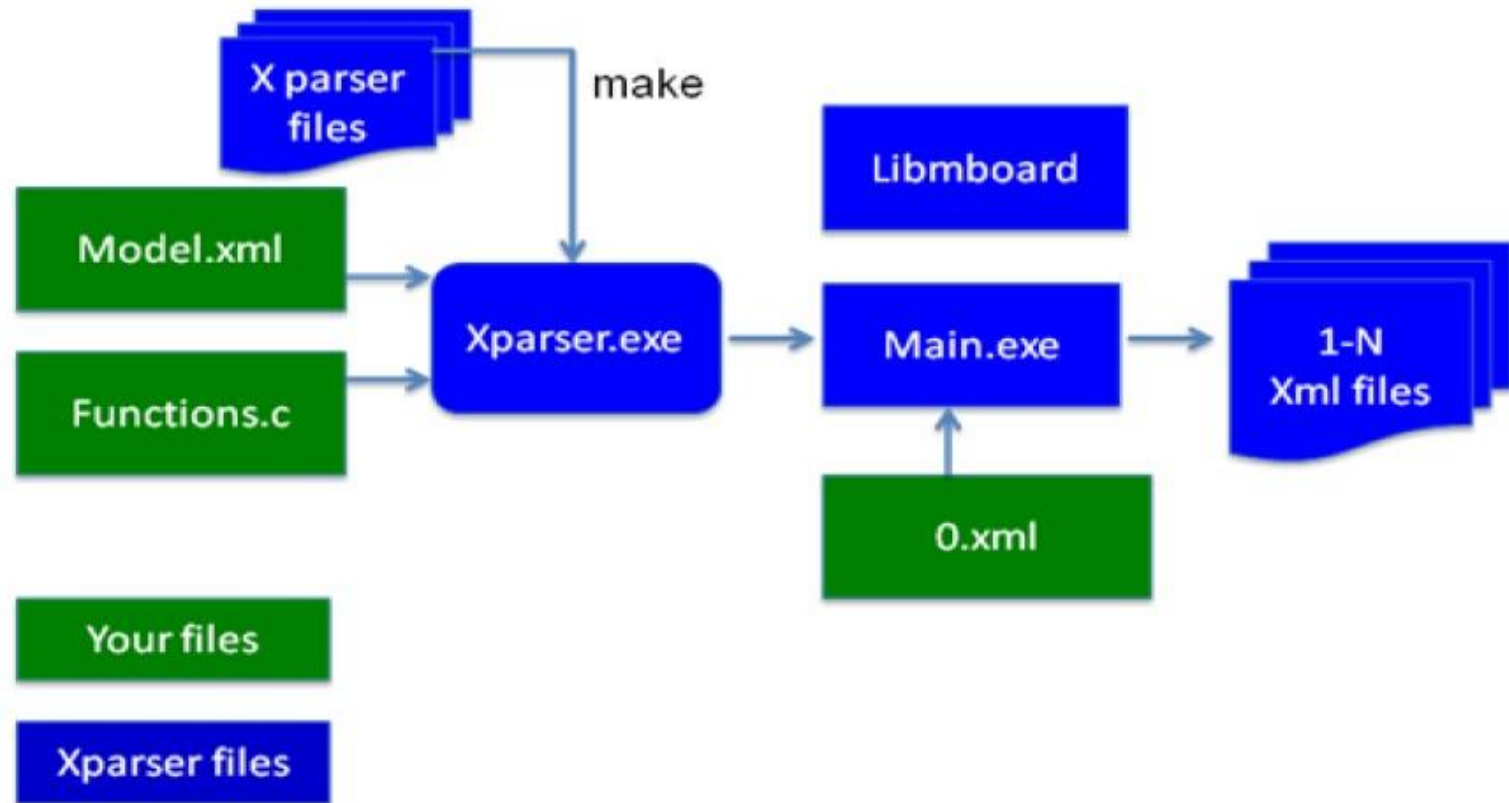


Science & Technology
Facilities Council

EPSRC

Engineering and Physical Sciences
Research Council

ARQUITECTURA DE FLAME



ARQUITECTURA DE FLAME

- **Model.xml** : Contiene toda la estructura del modelo (descripción de agentes, variables de memoria,...).
- **Functions.c** : Contiene las implementaciones de las funciones especificadas en model.xml.
- **0.xml** : Contiene los estados iniciales de las variables de memoria de los agentes.

PROYECTOS DE FLAME

- EURACE
- SUMO2
- FLAMEGPU
- EURACE@UNIBI
- NF- κ B
- EPIDERMIS
- BLOOD-BRAIN BARRIER
- NATURAL KILLER/ TUMOUR CELLS
- EPITHELIOM
- NEOANGIOGENESIS IN CANCER



PROYECTO EURACE

- Estudio y desarrollo de modelos multiagente a nivel económico global.
- plataforma de software para realizar un potente entorno para las simulaciones basadas en agentes económicos de gran escala.



PROYECTO SUMO2

Investigación de la bacteria *Escherichia coli*, considerando como componentes individuales que componen un sistema biológico que trabajan juntos para producir patrones de comportamiento coherentes.



PROYECTO FLAMEGPU

- Es una unidad de procesamiento de gráficos de alto rendimiento (GPU), extensión del framework FLAME.
- Video: : <http://www.youtube.com/watch?v=LSoDfG-p6lk>

FLAME GPU

PRUEBAS CON FLAME

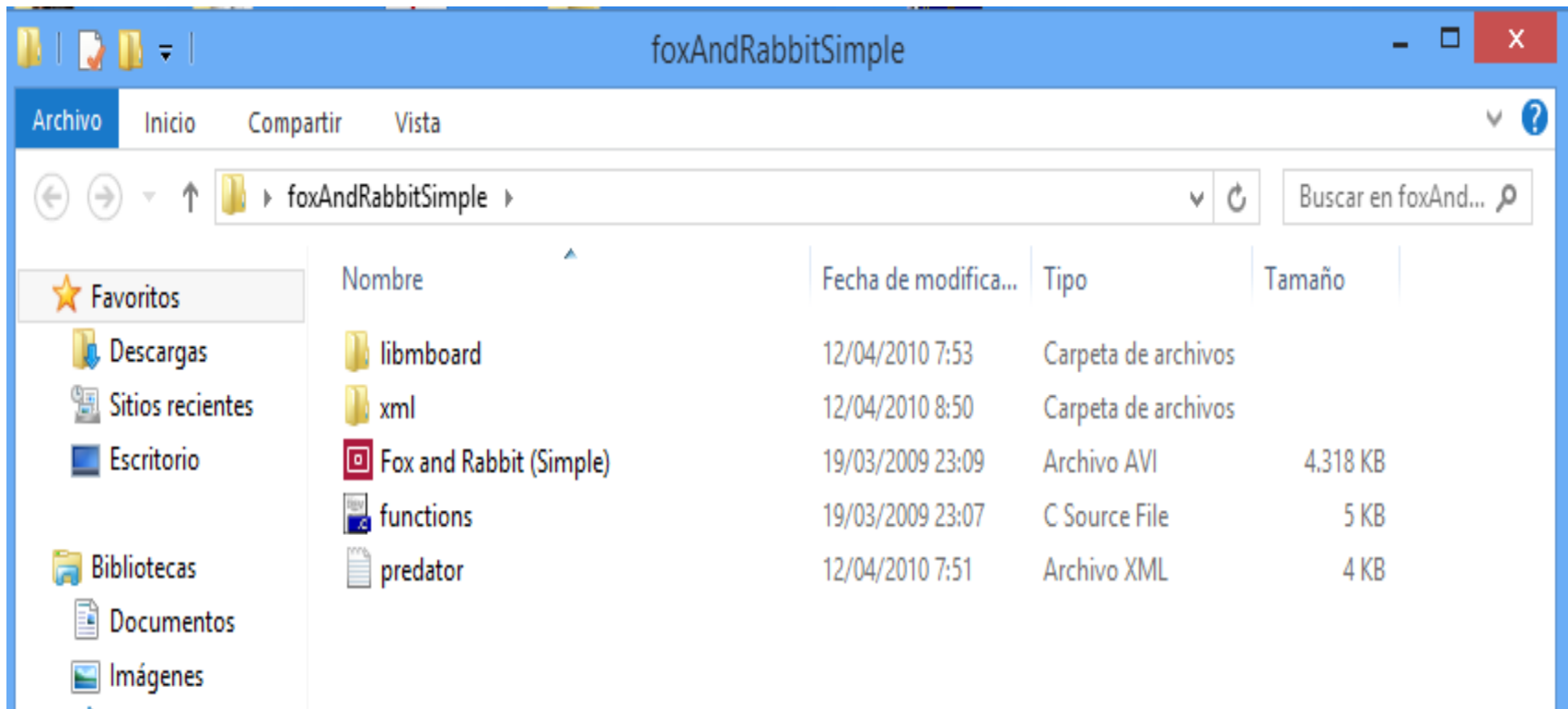
Material necesario:

- xparser
 - libmboard
 - winGW
 - Eclipse
 - CZero
 - jdom
 - Modelo Fox & Rabbit
-

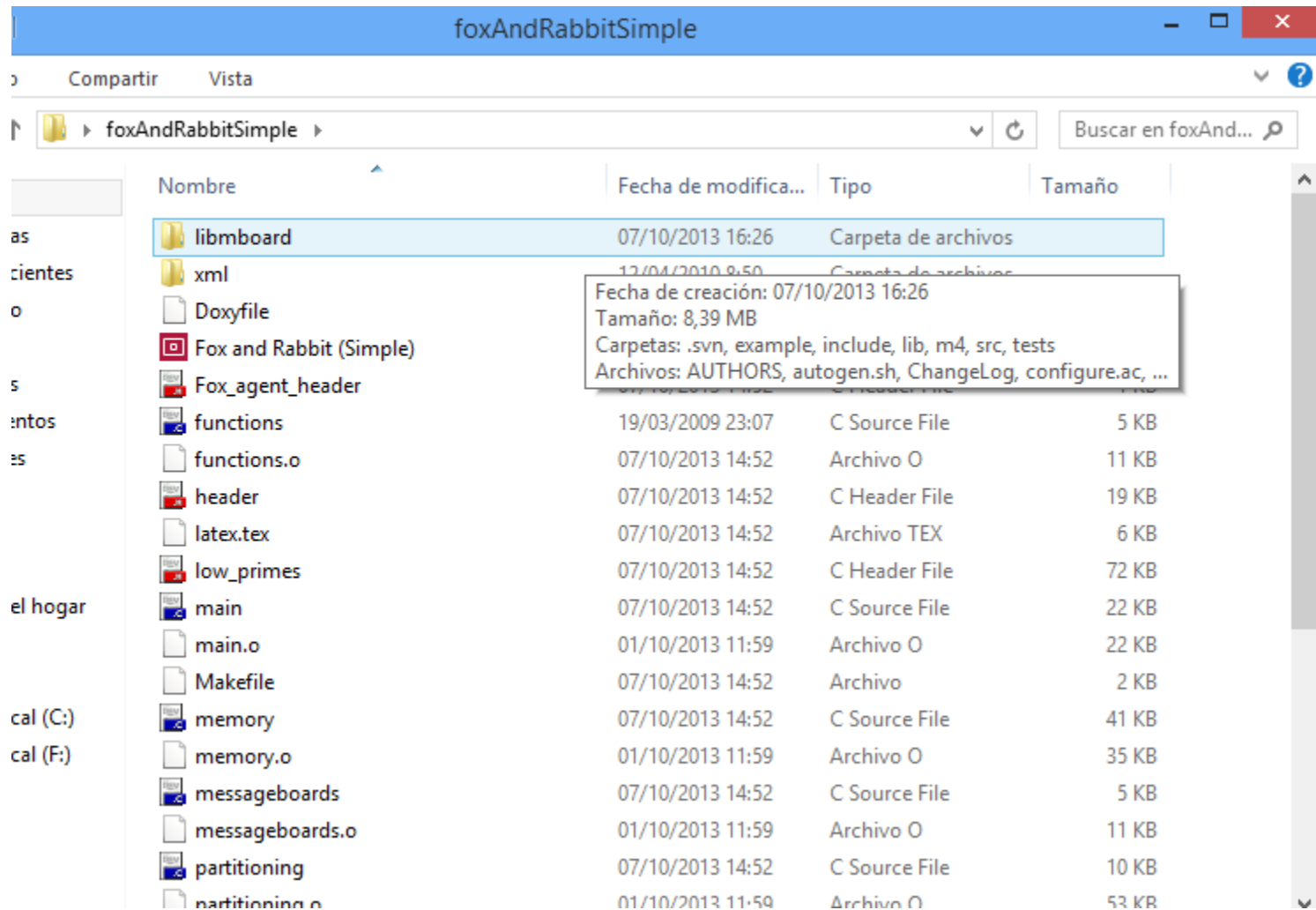
PRUEBAS FLAME

- Descargarse y descomprimir los ficheros de xparser y libmboard.
 - Instalación de Mingw.
 - Compilar el archivo xparser.c con la instrucción make de mingw desde la consola de comandos.
 - Ejecutar xparser.exe para el modelo elegido para la prueba (fox & Rabbit)
-

ANTES DE EJECUTAR XPARSER



DESPUES DE EJECUTAR XPARSER



Windows Explorer window titled "foxAndRabbitSimple" showing the contents of a directory. The address bar shows the path "foxAndRabbitSimple". The main pane displays a list of files and folders. A tooltip is visible over the "libmboard" folder, showing its creation date and size.

Nombre	Fecha de modifica...	Tipo	Tamaño
libmboard	07/10/2013 16:26	Carpeta de archivos	
xml	12/04/2010 9:50	Carpeta de archivos	
Doxyfile			
Fox and Rabbit (Simple)			
Fox_agent_header			
functions	19/03/2009 23:07	C Source File	5 KB
functions.o	07/10/2013 14:52	Archivo O	11 KB
header	07/10/2013 14:52	C Header File	19 KB
latex.tex	07/10/2013 14:52	Archivo TEX	6 KB
low_primes	07/10/2013 14:52	C Header File	72 KB
main	07/10/2013 14:52	C Source File	22 KB
main.o	01/10/2013 11:59	Archivo O	22 KB
Makefile	07/10/2013 14:52	Archivo	2 KB
memory	07/10/2013 14:52	C Source File	41 KB
memory.o	01/10/2013 11:59	Archivo O	35 KB
messageboards	07/10/2013 14:52	C Source File	5 KB
messageboards.o	01/10/2013 11:59	Archivo O	11 KB
partitioning	07/10/2013 14:52	C Source File	10 KB
partitioning.o	01/10/2013 11:59	Archivo O	53 KB

Tooltip for libmboard:
Fecha de creación: 07/10/2013 16:26
Tamaño: 8,39 MB
Carpetas: .svn, example, include, lib, m4, src, tests
Archivos: AUTHORS, autogen.sh, ChangeLog, configure.ac, ...

PRUEBAS FLAME

- Una vez obtenidos los archivos del modelo se compila el modelo con la instrucción `make.exe`.
 - Para ejecutar el modelo se ejecuta `main.exe` detallando el número de iteraciones y el estado inicial del modelo “0.xml”
-

CREAR ARCHIVO 0.XML

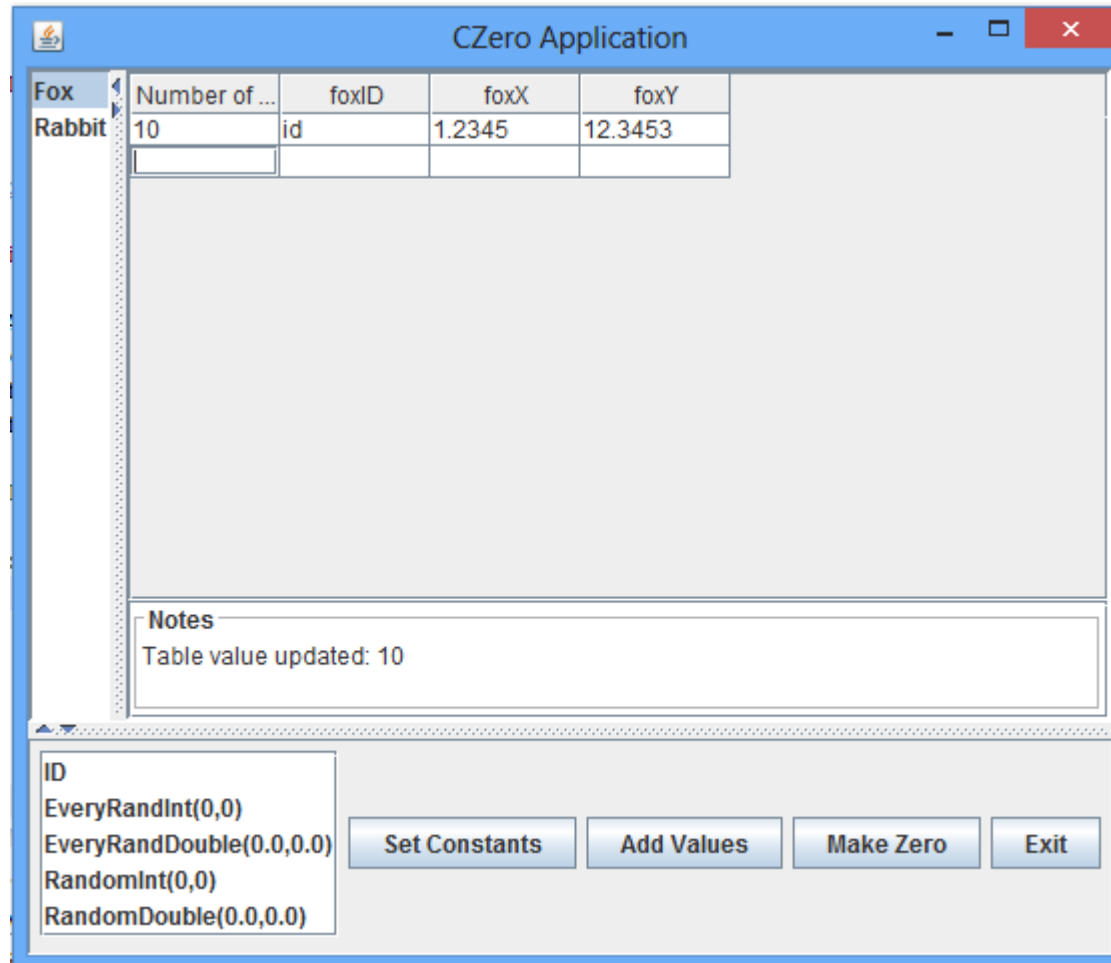
2 posibles caminos:

- Creando manualmente el archivo 0.xml
- Utilizando CZero en Eclipse.

Con CZero, se crean los agentes con la interfaz gráfica de la siguiente imagen.

Una vez esten todos los agentes añadidos se crea el documento pulsando el boton “make file”.

CZERO EN ECLIPSE



FUENTES

- Web oficial de FLAME : <http://www.flame.ac.uk/>
 - Manual de usuario Universidad de Sheffield:
http://www.dcs.shef.ac.uk/~wmlh/FLAME_manual.pdf
 - [http://gow.epsrc.ac.uk/NGBOViewGrant.aspx?
GrantRef=EP/I030654/1](http://gow.epsrc.ac.uk/NGBOViewGrant.aspx?GrantRef=EP/I030654/1)
-

Fin.
