



Control Basado en Misiones

DPI2011-28507-C02-01/02



Informe interno

T0305:

Desarrollo de herramientas de diseño

Autor:	J.L. Beltrán, A. Catalá, P. Balbastre, E. Munera,
Revisor:	José Simó
Fecha:	
Resumen:	

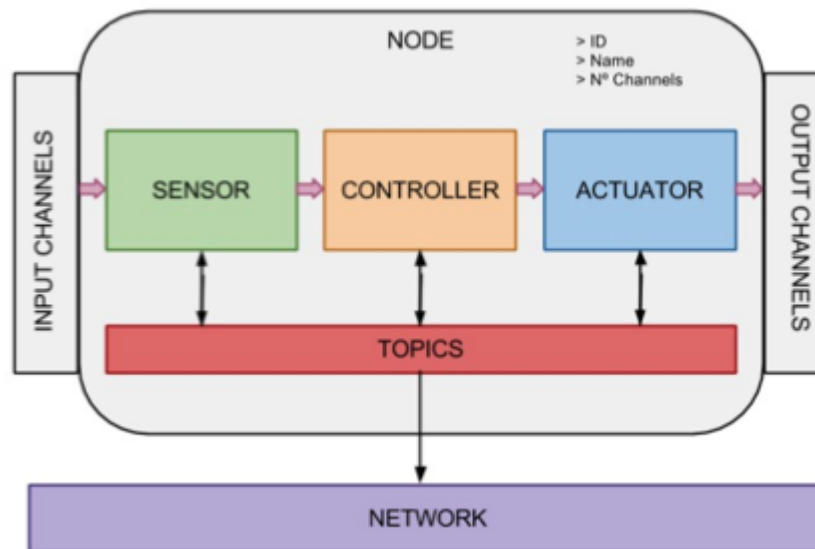


CONTENIDO

1. CKCEditor	3
2. Monitorización.....	6
3. Configuración de sistemas particionados	7

1. CKCEditor

El editor de configuración "Control Kernel Configuration Editor (CKCEditor)" es una herramienta gráfica destinada a facilitar al ingeniero el desarrollo de aplicaciones sobre el middleware de kernel de control. Este middleware dispone de un conjunto de servicios responsables de administrar los componentes básicos de un sistema de control: sensores, actuadores y controladores. Las capacidades de configuración de cada uno de los servicios son muy variadas, es por ello, se ha definido un modelo de configuración en el que se contemplan todas ellas. Este modelo ha sido incluido en CKCEditor permitiendo al usuario diseñar sus aplicaciones sin la necesidad de tener un profundo conocimiento acerca de la estructura interna del sistema. En la siguiente figura se muestra una representación gráfica del mismo.



Un sistema definido por el usuario puede estar compuesto por uno o varios nodos. El nodo es tratado como un componente lógico que no tiene una funcionalidad específica asociada sino que representa un elemento físico que contiene a otros componentes como son canales, sensores, actuadores y controladores. Los canales son considerados como una representación lógica de un elemento físico de entrada/salida mientras que sensores y actuadores conectan a ellos para la adquisición o envío de la información correspondiente. Tanto sensores como actuadores incluyen un conjunto de funciones predefinidas que permiten realizar diferentes tipos de adecuación sobre el valor recibido. Por último se encuentra el controlador que se le considera como un componente puramente lógico caracterizado por no estar asociado a ningún canal hardware. Las comunicaciones entre componentes del mismo nodo así como componentes distribuidos se realiza mediante topics, que no es más que un sistema basado en el modelo publicación-suscripción. El middleware hace uso de una implementación propia permitiendo ofrecer una determinada calidad de servicio. La calidad se encarga de determinar el nivel de satisfacción de un servicio garantizando un cierto grado de rendimiento en su ejecución.

CKCEditor cumple varios propósitos donde destaca el diseño de aplicaciones a través de una interfaz gráfica (ver figura 2), la generación automática de código (ver figura 3) y la monitorización del estado del sistema (ver figura 4). En el panel de edición se han incluido un conjunto de diseños a modo de biblioteca de dispositivos y periféricos que el usuario puede importar a su workspace con el fin de implementar soluciones basadas en estos componentes. La funcionalidad de generación automática de código construye ficheros en formato xml creados a partir del diseño especificado por el ingeniero. Estos ficheros son enviados al middleware para la configuración de sus servicios. Los ficheros de configuración pueden ser editados manualmente utilizando un simple editor de texto, si bien este modo de edición es algo tedioso que puede conducir a errores.

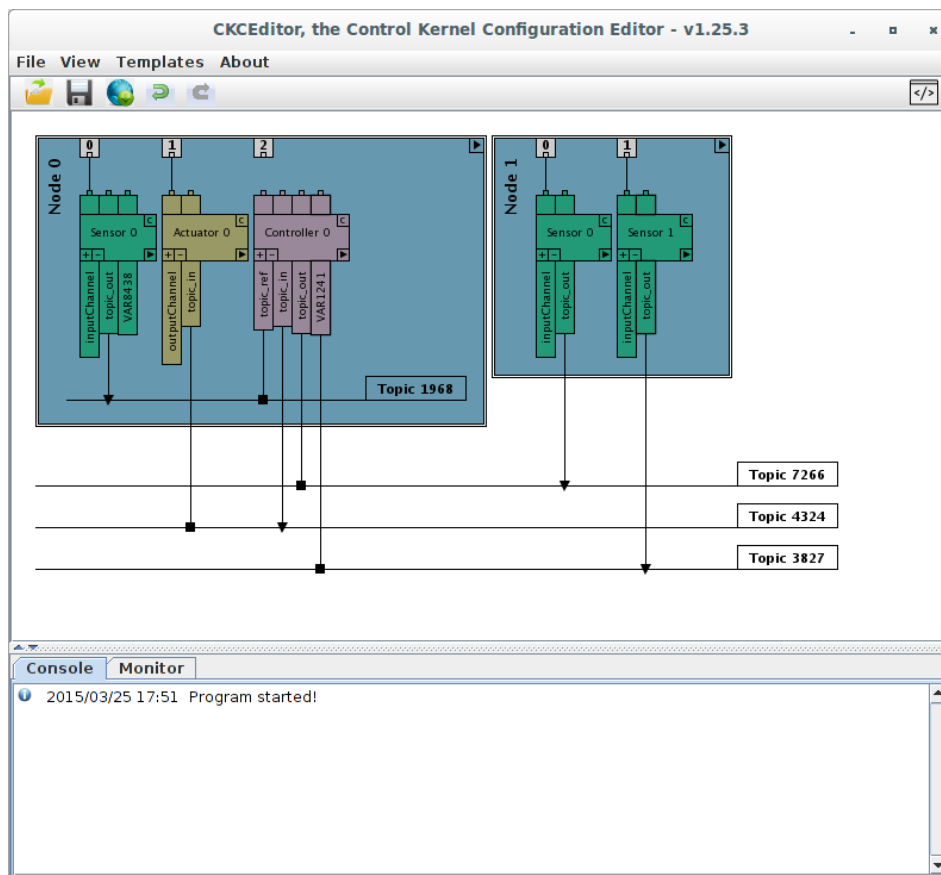
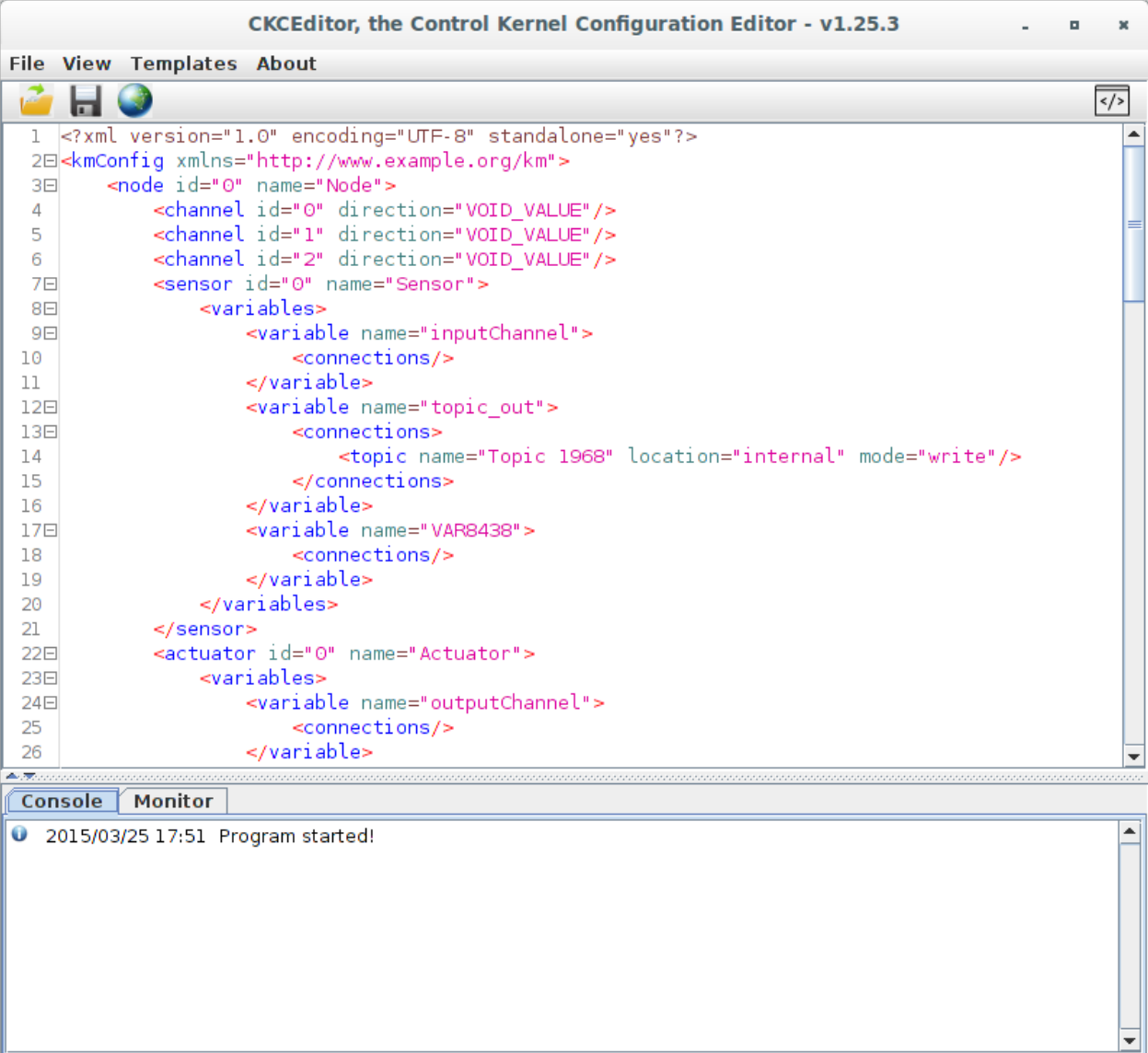


Figura 1 - Vista de diseño de aplicación

El middleware de kernel de control está definido como un conjunto de servicios e interfaces que se utilizan para la gestión de los componentes básicos de un sistema de control. Cada uno de estos servicios cuenta con unas capacidades de configuración a modificar por parte del usuario en función de la tarea a realizar. Se ha definido un modelo de configuración basado en ficheros XSD que contiene información precisa sobre la definición de los parámetros requeridos para realizar una petición de servicio y carga de la aplicación deseada. Con el fin de ofrecerle una asistencia en el diseño de aplicaciones, se ha implementado una herramienta gráfica (CKCEditor) que guía al usuario a través del modelo de programación a seguir. Dicho modelo, va incluido en la herramienta.

La configuración de los servicios del middleware es obtenida a través del uso de ficheros xml que son generados a partir del modelo. Cada xml contiene configuraciones de servicio que pueden ser validadas contra dicho modelo a través de los mecanismos de validación del estándar xml/xsd. Mediante la implementación de CKCEditor se logra delegar esta tarea a la herramienta. De esta forma, un usuario puede construir su aplicación de forma gráfica (ver figura 1) y ajeno por completo a como después ese diseño se traducirá a su equivalente en xml (ver figura 2). Para lograrlo se ha hecho uso de JAXB que proporciona la capacidad de serializar las referencias de objetos Java a XML y a la inversa, es decir, deserializar XML en objetos Java.



```
CKCEditor, the Control Kernel Configuration Editor - v1.25.3
File View Templates About
1 <?xml version="1.0" encoding="UTF-8" standalone="yes"?>
2 <kmConfig xmlns="http://www.example.org/km">
3   <node id="0" name="Node">
4     <channel id="0" direction="VOID_VALUE"/>
5     <channel id="1" direction="VOID_VALUE"/>
6     <channel id="2" direction="VOID_VALUE"/>
7     <sensor id="0" name="Sensor">
8       <variables>
9         <variable name="inputChannel">
10           <connections/>
11         </variable>
12         <variable name="topic_out">
13           <connections>
14             <topic name="Topic 1968" location="internal" mode="write"/>
15           </connections>
16         </variable>
17         <variable name="VAR8438">
18           <connections/>
19         </variable>
20       </variables>
21     </sensor>
22     <actuator id="0" name="Actuator">
23       <variables>
24         <variable name="outputChannel">
25           <connections/>
26         </variable>
27       </variables>
28     </actuator>
29   </node>
30 </kmConfig>
31 </pre>

Console Monitor



2015/03/25 17:51 Program started!


```

Figura 2 - Fichero xml equivalente generado automáticamente

Por tanto, cuando una determinada configuración creada desde la herramienta quiere ser enviada al middleware a través del servicio de comunicaciones se tiene la garantía que ha sido previamente validada contra el modelo definido. En aquellos casos que el usuario

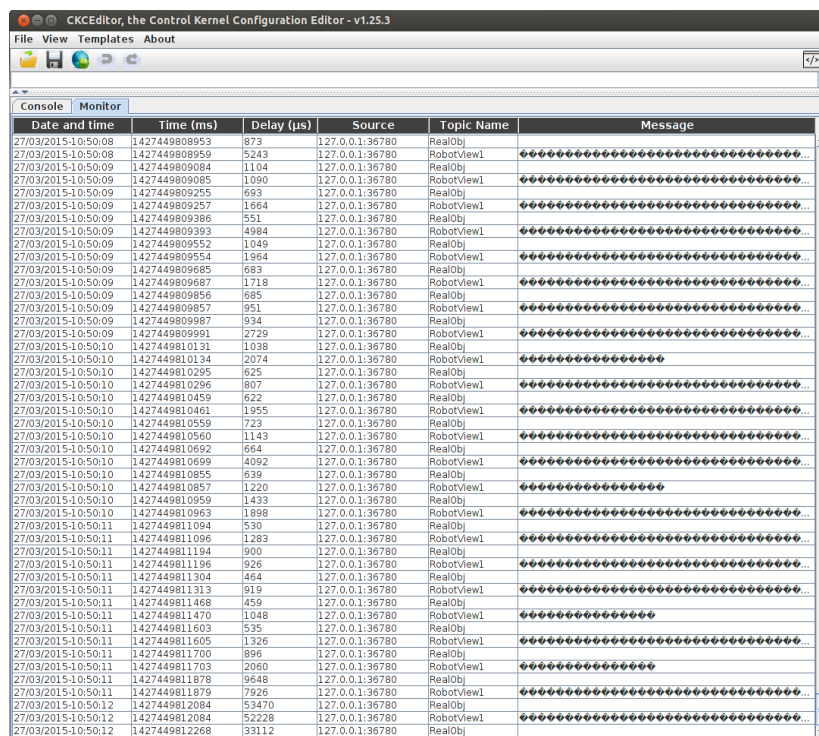
realice cambios de configuración durante la ejecución de la aplicación, estos serán almacenados junto al resto de ficheros de configuración para futuras ejecuciones.

2. Monitorización

La monitorización del sistema tiene por objetivo aumentar la fiabilidad y la aplicación de métodos de tolerancia a fallos. Se han implementado un módulo integrado en la herramienta CKCEditor que permite observar el estado del sistema de comunicaciones y de los procesadores involucrados en la ejecución de la aplicación de control. El módulo de monitorización hace uso del middleware de comunicaciones para suscribirse a los topics existentes en la red con el objetivo de hacer un seguimiento de los mensajes intercambiados atendiendo a:

- **Datetime:** Para conocer los instantes de tiempo en que fueron publicados los datos.
- **Delay:** Establece el tiempo transcurrido desde la publicación de un dato hasta la recepción por parte del suscriptor, en este caso, el visor. Esta medida permite detectar posibles problemas en la red.
- **Source:** Indica quien ha sido el nodo publicador. Para ello se hace uso de la IP que indica el dispositivo y el puerto que identifica al nodo en cuestión.
- **Topic name:** Nombre del topic.
- **Message:** Dato publicado.

Esta información es visualizada en una tabla como la mostrada en la figura 3. Los datos son almacenados en un registro y exportables en formato xls. Un usuario puede hacer uso de ellos como mecanismo de depuración para conocer el flujo y comportamiento de su aplicación dentro del sistema.



Date and time	Time (ms)	Delay (µs)	Source	Topic Name	Message
27/03/2015-10:50:08	1427449808953	873	127.0.0.1:36780	RealObj	
27/03/2015-10:50:08	1427449808959	5243	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809084	1104	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809085	1090	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809255	893	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809257	1664	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809386	551	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809393	4984	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809552	1049	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809554	1964	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809685	683	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809687	1718	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809856	685	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449809857	951	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:09	1427449809987	934	127.0.0.1:36780	RealObj	
27/03/2015-10:50:09	1427449810091	2729	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810131	1038	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810134	2074	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810295	625	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810296	807	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810459	622	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810461	1955	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810559	723	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810560	1143	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810692	864	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810699	4092	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810855	639	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810857	1220	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:10	1427449810959	1433	127.0.0.1:36780	RealObj	
27/03/2015-10:50:10	1427449810963	1898	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811094	530	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811096	1283	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811194	900	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811196	926	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811304	464	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811313	919	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811468	459	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811470	1048	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811603	535	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811605	1326	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811700	896	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811703	2060	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:11	1427449811878	9648	127.0.0.1:36780	RealObj	
27/03/2015-10:50:11	1427449811879	7926	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:12	1427449812084	53470	127.0.0.1:36780	RealObj	
27/03/2015-10:50:12	1427449812084	52288	127.0.0.1:36780	RobotView1	*****
27/03/2015-10:50:12	1427449812268	33112	127.0.0.1:36780	RealObj	

Figura 3 - CKCEditor, visor de monitorización

La monitorización pretende cubrir diferentes aspectos como son:

- Calidad de servicio
- Monitorización de las configuraciones
- Monitorización de los datos

Dentro de las calidades se puede diferenciar entre calidad de servicio y calidad de contexto. La calidad de servicio ofrece información relativa a la comunicación y se evalúan parámetros como por ejemplo los plazos en la recepción de los datos. Por otro lado, la calidad de contexto es un factor que evalúa el desempeño de las tareas a ejecutar a partir del chequeo de la cantidad de recursos de sistema utilizado y una predicción de calidad basada en el previo estudio del comportamiento del mismo.

La monitorización de configuraciones tiene la finalidad de asegurar que las configuraciones aplicadas a los diferentes dispositivos están validadas respecto al modelo. Si bien CKCEditor ya genera automáticamente ficheros xml validados, se dispone la funcionalidad de hacer una carga manual, por tanto, esta comprobación sigue siendo necesaria.

La monitorización de los datos analiza el payload de los mensajes y visualiza su contenido. Los tipos de datos asociados al mensaje varía en función del propósito de la aplicación ejecutada, así pues se puede estar trabajando con valores numéricos, imágenes, etc. Una visualización de los datos en crudo no siempre es la solución más acertada. Se tiene previsto la implementación de un plugin para CKCEditor que tenga en cuenta el tipo de dato para realizar la visualización correspondiente.

3. Configuración de sistemas particionados

Se ha desarrollado una aplicación de configuración de sistemas particionados que permite la definición y edición gráfica de configuraciones de ejecución para el hypervisor de código abierto "XtratuM". La configuración de un sistema particionado es una operación delicada y compleja que involucra la definición de la plataforma de ejecución, la asignación de recursos hardware a cada partición, la asignación de áreas de memoria a las particiones e hypervisor, la definición del tratamiento de eventos de error (HealthMonitoring), la definición de los canales de comunicación entre particiones y la asignación de tiempos de ejecución para cada una de las particiones.

La herramienta desarrollada es muy versátil y está basada en los modelos definidos para cada una de las entidades: plataforma, asignación de recursos y carga computacional (ver figura 4).

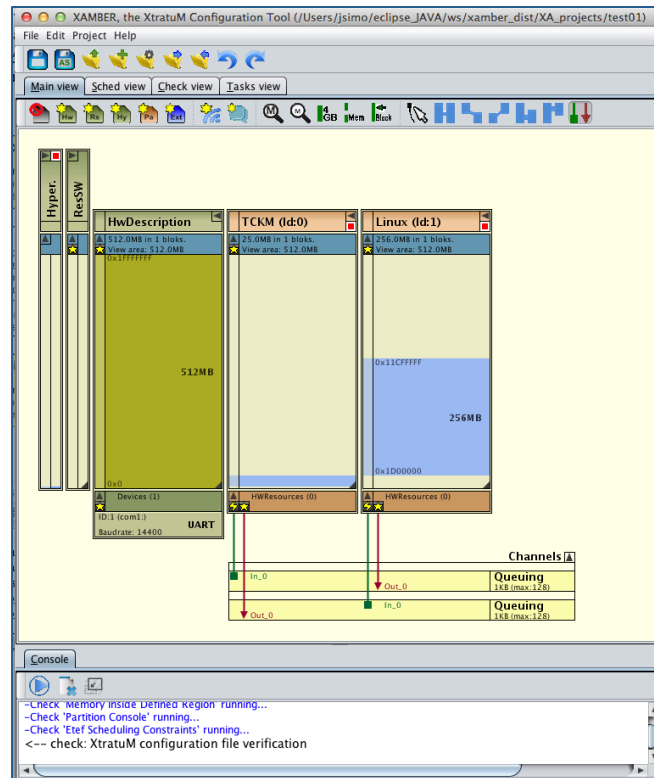


Figura 4: Panel de asignación de memoria y canales de comunicación

Aunque la configuración de un sistema particionado se realiza comúnmente considerando cada partición como una caja negra (sin tener en cuenta qué tareas se ejecutan en su interior) a la que se la asignan recursos temporales y espaciales, la aplicación soporta una visión de caja blanca en la que se pueden definir flujos de ejecución entre particiones, dependencias de recursos etc. (ver figura 5). Esta definición de caja blanca, en el entorno de la aplicación se usa para invocar a una herramienta externa de planificación que elabora una asignación temporal para cada partición siguiendo un esquema de planificación plan de un solo nivel. El resultado de la planificación se incorpora al modelo de configuración dentro del mismo entorno de la aplicación.

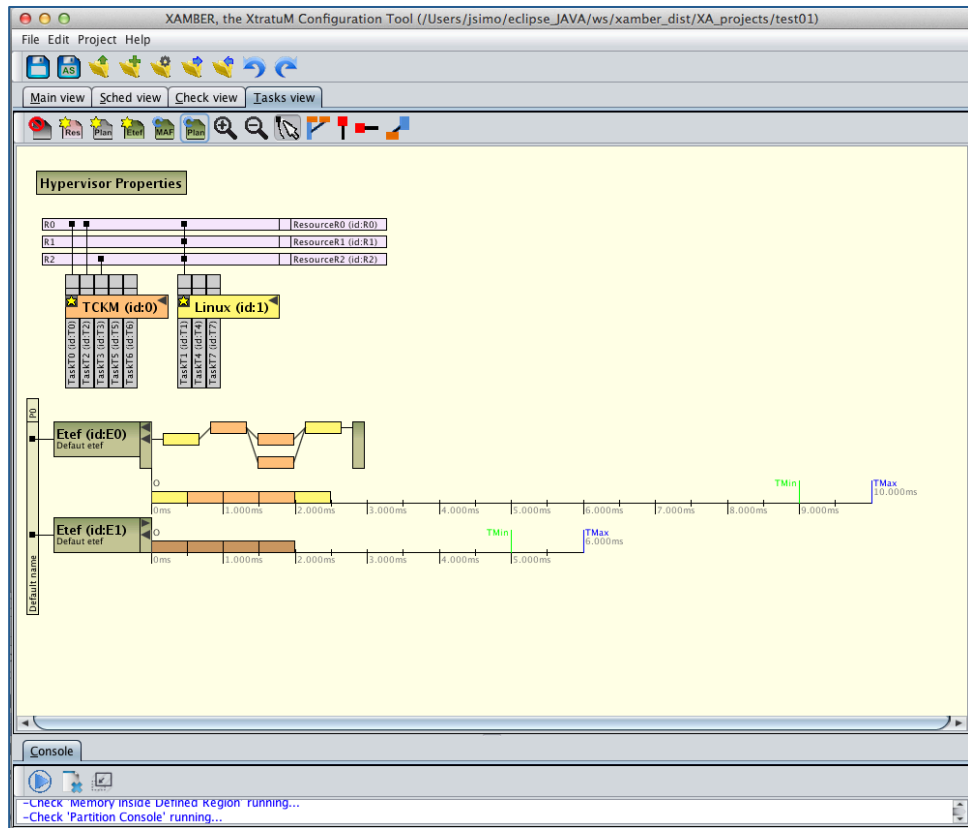


Figura 4: Panel de definición de la carga computacional (caja blanca)

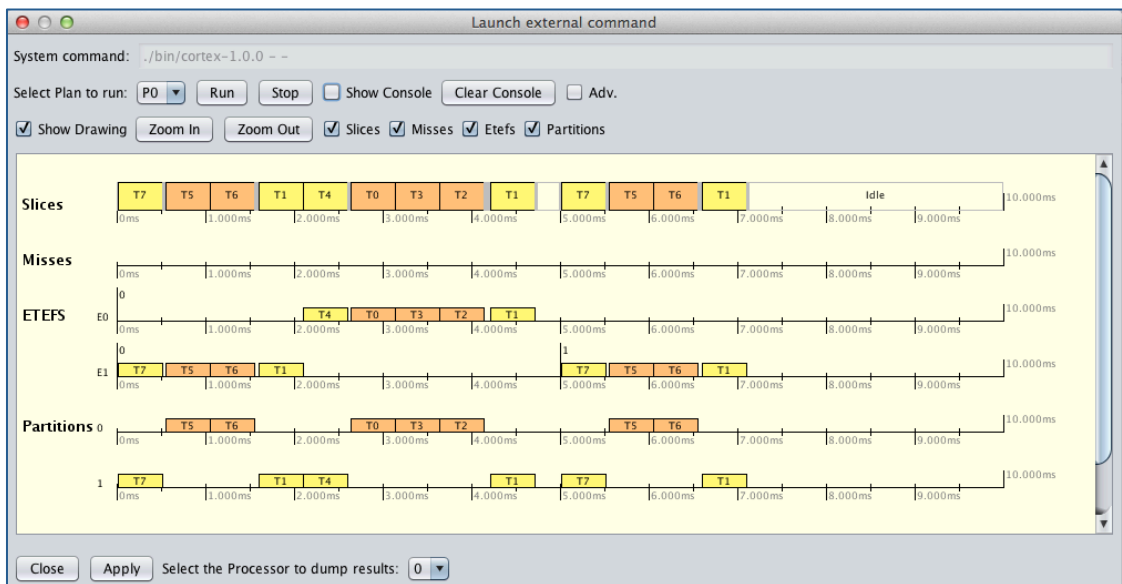


Figura 5: Resultado de la planificación de actividades