



OpenNICTA Open Source Software Information and Video Transcript

Software name:	CAMkES
Software description:	CAMkES (Component Architectures for microkernel-based Embedded Systems) is a software development and runtime framework for quickly and reliably building microkernel-based multiserver (operating) systems.
Project description:	The <u>CAMkES Project</u> focused on creating a quick catalogue solution for embedded systems. Creating new component technology increases the reliability of embedded systems and decreases the cost.
Link to Software:	http://www.ertos.nicta.com.au/software/camkes/

Video Transcript

Interview with Ihor Kuz

Tell us about yourself and the research you are currently working on

Ihor: My name is Ihor Kuz and I'm a Senior Researcher at NICTA. I work with the ERTOS group – we're working on embedded systems and operating systems for embedded systems. My main interest is on componentisation, using component-based software engineering techniques for building these systems – in particular for building the operating systems. So the work started with the CAMkES project and essentially I'm continuing that work, making it work on the new platform sel4 that we are working on, and integrating that into a larger process.

Tell us about your open source software

Ihor: So, what CAMkES is it's a component-based development platform for building embedded systems software on top of the L4 microkernel. What CAMkES provides is a set of tools that essentially make it easier to build this kind of embedded systems software. So the tools allow you to design and construct the systems using higher level abstractions, in particular components and connectors, and this means you don't have to, when you are building these systems, worry about the intricacies and details of the underlying microkernel. The difference is similar to building a model using Lego bricks that you can easily snap together. It's easy to use Lego bricks compared to doing it from scratch and worrying about metals and woods and nuts and bolts and nails. So, it's easy and simple to build a model using Lego and CAMkES does a similar thing for L4, for building systems on L4.

Why open source?

Ihor: So there are a couple of reasons for going with open source. Firstly because CAMkES is quite closely tied to or associated with the OKL4 microkernel – the OKL4 version – and that's open source, so they have a community built around this open source microkernel and the users are used to using open source, and so if we want adoption within that community, it makes sense to go with open source. Also by going open source, it makes it possible to involve the users of CAMkES

more, in particular in improving the framework and making a better framework. And then lastly, because we are targeting embedded systems and a property of embedded systems is that most of devices and platforms used are slightly different – they all differ from each other. Making the framework open source means that the users of CAMkES that are targeting specific embedded systems and embedded devices are free to tweak it and tinker with it to make it actually work on their platforms.