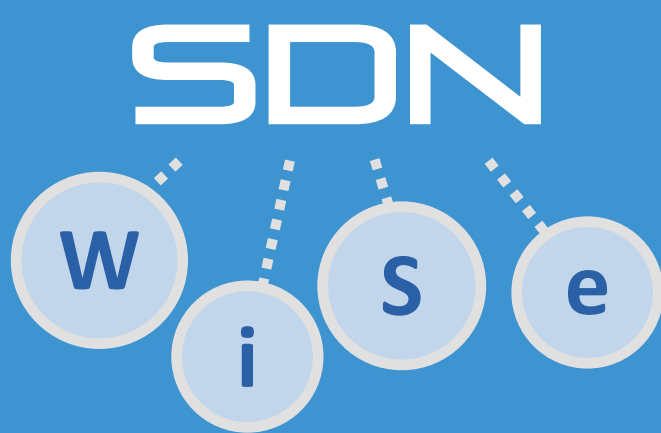


Reprogramming Wireless Sensor Networks by Using SDN-WISE: a Hands-On Demo.



Laura Galluccio¹, Sebastiano Milardo², Giacomo Morabito¹, and Sergio Palazzo¹.

¹Electric, Electronics and Computer Engineering Department, University of Catania, Italy
²CNIT Research Unit, Catania, Italy

Abstract

SDN-WISE is a Software Defined Networking solution for Wireless Sensor Networks (WSNs). This work shows the peculiarities of SDN-WISE, and how easy it is programming the behavior of a WSN by exploiting the SDN-WISE features.

Features

- SDN-WISE introduces some novel functions which are particularly convenient in WSNs:
- Statefulness
 - Flexible definition of rules
 - Support of duty cycles
 - Support of multitenancy
 - Lots of deployment options and programming languages
 - Integration with simulation environments

Design and Implementation

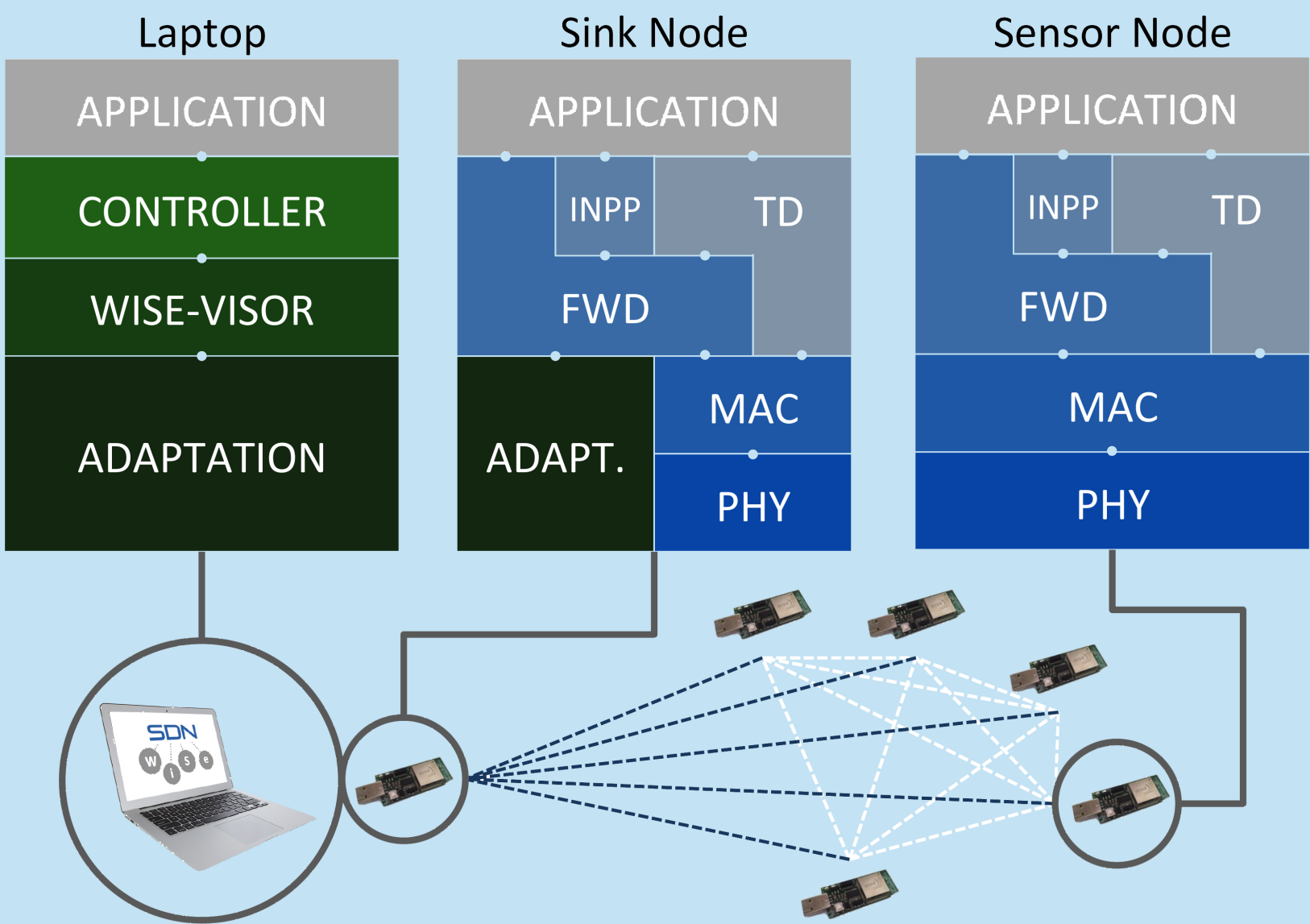


Figure: SDN-WISE@INFOCOM testbed and protocol stack.

Downloads

The source code of the controller and the firmware are available under GPL 3.0 licence on GitHub. For more information please visit <http://sdn-wise.dieei.unict.it>.

Use Case

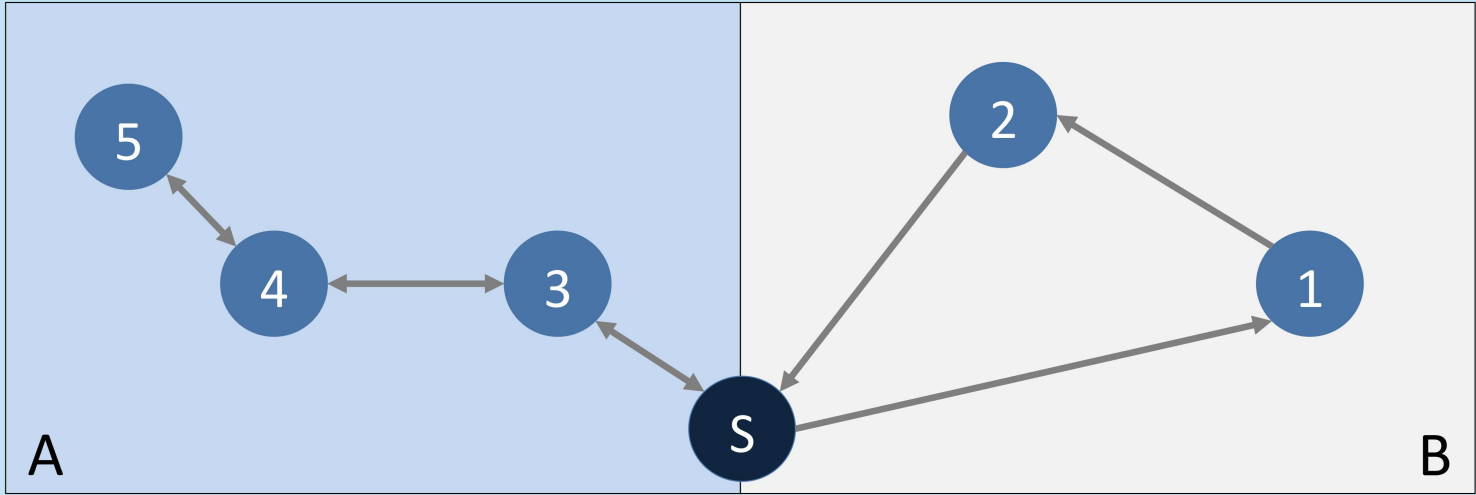


Figure: Two controllers operating on the same network.

Testbed

- Sensor nodes:
 - ▷ Embit IEEE 802.15.4 boards (EMB-Z2530PA)
 - ▷ Memory: 8 kB RAM + 256 kB Flash memory
 - ▷ 40 kB of memory used for MAC (TIMAC v.1.4.0)
 - ▷ 10 kB of memory used for SDN-WISE
- Control plane:
 - ▷ WISEVisor + Controllers hosted in the same PC
 - ▷ Controllers implemented in Java

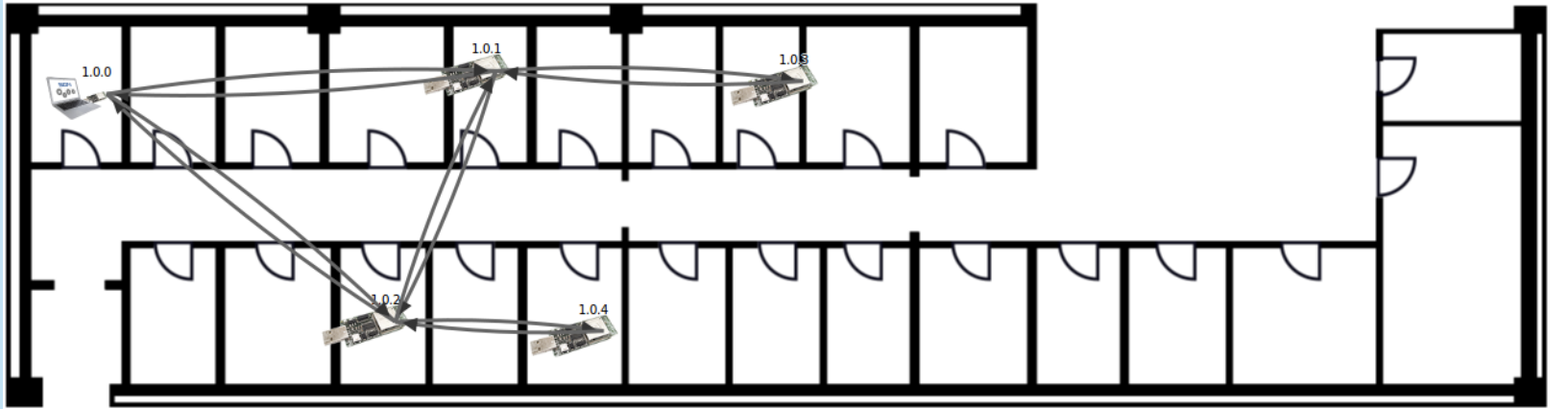


Figure: SDN-WISE@University of Catania testbed.

Conclusions

SDN-WISE reduces the amount of information exchanged between sensors and the SDN controllers by using a stateful approach and provides a new level of abstraction which allows several logical WSNs to coexist over the same set of sensor nodes.

Acknowledgement

This work was supported by the 7th Framework Program of the European Commission within the NEWCOM# project and by MIUR under the SIGMA contract.

Matching Rule					Matching Rule					Matching Rule					Action					Stats	
Operator	Size	Location	Offset	Value	Operator	Size	Location	Offset	Value	Operator	Size	Location	Offset	Value	Type	Multimatch	Location	Offset	Value	TTL	Count
=	2	PACKET	SRC_ADDR	B	>=	2	PACKET	DATA	Xthr	=	1	STATE_ARR	0	0	MODIFY	TRUE	STATE_ARRAY	0	1	255	0
=	2	PACKET	SRC_ADDR	B	<	2	PACKET	DATA	Xthr	=	1	STATE_ARR	0	1	MODIFY	TRUE	STATE_ARRAY	0	0	255	0
=	2	PACKET	SRC_ADDR	B	-	-	-	-	-	-	-	-	-	-	FORWARD	FALSE	PACKET	0	D	255	0
=	2	PACKET	SRC_ADDR	A	=	1	STATE_ARRAY	0	0	-	-	-	-	-	DROP	FALSE	PACKET	0	100%	255	0
=	2	PACKET	SRC_ADDR	A	=	1	STATE_ARRAY	0	1	-	-	-	-	-	FORWARD	FALSE	PACKET	0	D	255	0

Figure: WISE Table for use case 1.