Question 2	What are the key ideas (design choices, main components) of their proposed architecture ? (You should, amongst, address the following points: The southbound interface used? Are RPL,CoAP,etc still used and for what? How does the topology discovery work?)
Pas encore répondu	

SD6WSN Proposed Architecture

Design choices and Main components

Please find below the main components that make a SD6WSN architecture. The design choice for each component is determined by answering those questions. Collectively, the answers for all the components determines the design of the SD6WSN architecture.

1. Control layer – Set of controllers

- how many controllers? And, what is the strategy for adoption of Virtualization techniques?

2. Forwarding layer – Set of switching devices/nodes/WSN nodes

- How many nodes in the WSN network? What is the topology used?
- Which nodes have the function of border routing + data plane forwarding + control plane routing?
- In the SD6WSN, the border routing node is called as 6LoWPAN Border Router (6LBR).

3. Networking applications & North-bound interface (NBI)

- what networking applications & algorithms are used in the SDN controller?
- What implementation model/language is used – OpenFlow, Yang etc?
- What is the NBI protocol used?

4. SD6WSN agent

- The agent which is part of the SD6WSN nodes that co-ordinates the communication between the sensing applications and the controller.
- What are the sensing applications in a WSN node and what are their hardware/software/battery requirements?

5. Communication protocol & South bound Interface (SBI)

- SD6WSNP – The control plane protocol which is used by the SDN controllers and the SD6WSN agents to communicate to each other and to exchange information.
- South bound interface – What is the interface technology (example - Netflow/Yang etc) used to implement the SD6WSNP between the controller co-ordinator and the SD6WSN agent

6. Protocol stack

- In SD6WSN architecture, the protocol stack used is – IEEE 802.15.4 at MAC, 6LoWPAN and RPL are Network layer, UDP at transport layer, CoAP at application layer.
- What are the other protocol stack choices and combinations?

Role of Southbound Interface

The south bound interface (SBI) is a set of software Interface APIs used for communication between the SDN controller and the SD6WSN nodes using the SD6WSNP protocol.

The SD6WSNP defines four messages: node-mod, info-get, flow-mod and packet-in. All these messages use the SBI API's for communication.

OpenFlow API's are an example for SouthBound Interface (SBI) APIs.

Role of CoAP

The role of CoAP as an application layer protocol in the SD6WSN architecture are as below,

- X`CoAP messages are used in the SD6WSNP protocol for exchanging communication between the controller co-ordinator and the SD6WSN agent.
- The CoAP messages **from SD6WSN agent to the SDN controller**, provides the required information that helps the controller in,
    - Managing the entries in the flow tables
    - Obtaining local information from nodes that includes neighbourhood info, wireless link quality info, geo-location and power transmission level
    - Modifying node behaviour attributes, setting up power level and transmission levels
- The CoAP messages **from the SDN controller to SD6WSN agent** of the SD6WSN nodes, helps the agent to perform following actions,
    - Installs rules in the flow table
    - Drives packet forwarding according to the flow table
- The CoAP messages **from the SDN controller to SD6WSN agent of the 6LBR** Border routing nodes, helps the agent to perform following actions,
    - Topology management related actions
    - Conversion of IPV6 to 6LoWPAN

Role of RPL

I have mentioned earlier, regarding what I understood about the role of the SD6WSNP protocol and CoAP protocols in communication exchange between the SDN controller and the SD6WSN nodes.

Now, I will present my understanding about the role of the RPL protocol as a routed protocol in the SD6WSN architecture,

- RPL + 6LoWPAN together make a reliable protocol stack to transport the SD6WSNP messages from each node in the WSN topology to the SDN controller. This way, RPL helps to ensure that the SDN controller remains always updated about the **latest** status of each node. Thus, RPL helps the SDN controller to have a unified view of the network as close as possible to reality.
- RPL does the first job in node discovery. When ever the RPL identifies a new node it informs the Border 6LBR node. The border 6LBR node inturn sends a notification to the controller.
- RPL helps in updating the status change in network due to following events,
    - Node failure, significant improvement of the quality of a link and deterioration of the quality of a link
- With the updated knowledge of the topology & the unified view of the WSN network and based on the decisions of the networking applications, the SDN controller is enabled to

prepare the rules for packet forwarding and populate those rules in the 'flow table' in each of the WSN switches/nodes.

- Thus, RPL enables the SDN controller in computing the rules and flow tables as demanded by the WSN network. So, I understand that, RPL's role is crucial for the operation of SD6WSN.

Working of Topology discovery and topology management

SD6WSNP defines four messages: node-mod, info-get, flow-mod and packet-in. Node-mod, info-get and flow-mod are used for topology discovery and maintenance.

The SD6WSN controller consists of three modules: coordinator; topology discovery and management; and flow control. The topology discovery and management module is triggered by flow-mod notifications and info-get notifications received from nodes. These notifications inform the SDN controller about significant changes in neighbourhood wireless channels.

As I understand, these messages are used by the SDN controller in the order of steps as mentioned below to discover the topology and then to manage the discovered topology:

STEP-1

As mentioned earlier, whenever a new node is identified by RLP during node inclusion case or when the node is removed in the exclusion case, the 6LBR gets notified. The Controller uses **node-mod** to request 6LBR for notification regarding new node identification. Thus, when there is new node identification, the border router 6LBR sends a notification to the controller.

STEP-2

a) Just after receiving a notification as discussed in step-1, the controller sends **info-get** to the nodes to obtain the neighbours of the discovered node and the quality of respective wireless links. A networking application can also ask the controller to send an info-get, whenever it needs information from a node.
b) Moreover, the SDN controller, sends the **info-get** 'with an 'observe' option being activated. This registers **request for notification** at SD6WSN nodes. The SD6WSN nodes, then, act as a CoAP server and notifies whenever a relevant change state occurs. **Three events** are considered relevant changes: node failure, significant improvement of the quality of a link and deterioration of the quality of a link. This way, the SDN controller learns the changes in the topology and keeps itself updated with latest topology at field.
c) After registering the node information in its database, the controller sends a **packet-in** message instructing the node to send back a notification when it receives data plane packets that do not match any entry in its flow table. The flow control module in the SDN controller receives and processes the packet-in notifications. The state of the SD6WSN is stored in the networking database (DB).

STEP-3

The SD6WSNP protocol uses the information sent in **flow-mod notifications** to include in and exclude nodes from the SD6WSN topology. **Flow-mod** is used to insert and remove entries in flow tables, allowing the networking applications to install flows according to their traffic needs.