Pursuing a Software-Defined Information-Centric Network

Dimitris Syrivelis

CERTH-ITI and University of Thessaly
Contents

• An Information – Centric Network Architecture
  • An ICN Node Architecture
  • Using LIPSIN for packet forwarding
  • Designing a LIPSIN switch using Openflow and changes to ICN node architecture
  • Benefits of using SDN support
An ICN Network Architecture

• Proposed by FP7 PURSUIT Project
  http://www.fp7-pursuit.eu
• SDN support developed in FP7 OpenLab project
  http://www.ict-openlab.eu
• A modular design that supports publish/subscribe semantics with 3 discrete functions:
  – Rendezvous
  – Topology Management
  – Forwarding
A domain deployment example

Node 1

Node 2

Node 3

Node 4

Node 5
Rendezvous Node

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABCD</td>
<td>Pub</td>
<td>1</td>
</tr>
<tr>
<td>0xABCD</td>
<td>Sub</td>
<td>3</td>
</tr>
</tbody>
</table>
Topology Management/Forwarding

Topological diagram with nodes labeled 1 to 5, connected as follows: 1-2, 1-3, 2-3, 3-4. Each node is associated with an Fw Logic module, and there is a separate Topology Manager node.

EWSDN 2012, 25-26 October, Darmstadt, Germany
Example System Operation

Node 1

Node 2

Node 3

Node 4

Node 5

Topology Manager

Fw Logic

Information Identifier | Type | Node ID
--- | --- | ---

Rendezvous

Domain Network

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

Node 1
Node 2
Node 3
Node 4
Node 5

Domain Network

Topography Manager

Information Identifier
Type: Pub
Node ID: Node1

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

Node 1
Node 2
Node 3
Node 4
Node 5

Domain Network

Toplogy Manager

Information Identifier | Type | Node ID
0xABC1234 | Pub | Node1

Rendezvous

Node 4
Node 5

Fw Logic

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

Rendezvous

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABC1234</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABC1234</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

Domain Network

EWSDN 2012, 25-26 October, Darmstadt, Germany
Example System Operation

Topology Manager

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABC1234</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABC1234</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

Node 1
Fw Logic

Node 2
Fw Logic

Node 3
Fw Logic

Node 4
TM Req

Node 5
Fw Logic

Domain Network

Pub

Sub

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABC1234</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABC1234</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

Domain Network

Node 1
Fw Logic

Node 2
Fw Logic

Node 3
Fw Logic

Node 4
Fw Logic

Node 5
Fw Logic

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

Topology Manager

Information Identifier | Type | Node ID
--- | --- | ---
0xABC1234 | Pub | Node1
0xABC1234 | Sub | Node3

Start Publish Notification

EWSDN 2012, 25-26 October, Darmstadt Germany
Example System Operation

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABC1234</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABC1234</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Contents

• An Information – Centric Network Architecture
• An ICN Node Architecture
• Using LIPSIN for packet forwarding
• Designing a LIPSIN switch using Openflow and changes to ICN node architecture
• Benefits of using SDN support
An ICN node architecture (BlackAdder prototype)

The service model exports pure publish subscribe semantics, along with synchronization primitives for robust application development.

EWSDN 2012, 25-26 October, Darmstadt
Germany
Contents

• An Information – Centric Network Architecture
• An ICN Node Architecture
• Using LIPSIN for packet forwarding
• Designing a LIPSIN switch using Openflow and changes to ICN node architecture
• Benefits of using SDN support
LIPSIN for packet forwarding

- LIPSIN (Petri Jokela et al.) is a source-based routing system which uses bloom filters to encode routes to one or more destinations (multicast trees).
- LIPSIN encodes physical links by applying bloom filters on a fixed size, few-bytes long, identifier which is prepended on each packet.
- Once routes are encoded into a single forwarding identifier at the source, LIPSIN forwarding achieves line speed.
How LIPSIN works
How LIPSIN works

i) Assign fixed length deployment-unique identifiers to all physical links
How LIPSIN works

ii) For each set of destinations, you compute the route at the source as follows:
How LIPSIN works

ii) For each set of destinations, you compute the route at the source as follows:
iii) The forwarding identifier can then be used on each forwarder to choose local outgoing links.
How LIPSIN works

iii) The forwarding identifier can then be used on each forwarder to choose local outgoing links.
iii) The forwarding identifier can then be used on each forwarder to choose local outgoing links.
iii) The forwarding identifier can then be used on each forwarder to choose local outgoing links.

EWSDN 2012, 25-26 October, Darmstadt Germany
iv) Multihop routing is implemented also by the same operation on the Forwarding identifier on each local forwarder.
iv) Multihop routing is implemented also by the same operation on the Forwarding identifier on each local forwarder.
Contents

• An Information – Centric Network Architecture
• An ICN Node Architecture
• Using LIPSIN for packet forwarding
• Designing a LIPSIN switch using Openflow and changes to ICN node architecture
• Benefits of using SDN support
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

<table>
<thead>
<tr>
<th>LIPSIN IDENTIFIER</th>
<th>INFORMATION IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Payload</td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

<table>
<thead>
<tr>
<th>LIPSIN IDENTIFIER</th>
<th>INFORMATION IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Payload</td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

EWSDN 2012, 25-26 October, Darmstadt Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

LIPSIN IDENTIFIER

INFORMATION IDENTIFIER

Payload

EWSDN 2012, 25-26 October, Darmstadt
Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

LIPSIN IDENTIFIER

INFORMATION IDENTIFIER

Payload

EWSDN 2012, 25-26 October, Darmstadt Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

<table>
<thead>
<tr>
<th>LIPSIN IDENTIFIER</th>
<th>INFORMATION IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Payload</td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Why using LIPSIN with SDN and not directly information identifiers?

ICN ethernet frame

```
<table>
<thead>
<tr>
<th>LIPSIN IDENTIFIER</th>
<th>INFORMATION IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Payload</td>
</tr>
</tbody>
</table>
```

EWSDN 2012, 25-26 October, Darmstadt Germany
Modifying LIPSIN functionality for Openflow datapaths

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Moving LIPSIN functionality to Openflow datapaths

Topology Manager

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Rendezvous

Node 4
Fw Logic

Controller

OF Datapath

Node 1
Fw Logic

Node 2
Fw Logic

Node 3
Fw Logic

Node 5
Fw Logic

Node 1

Node 2

Node 3

EWSDN 2012, 25-26 October, Darmstadt Germany
Moving LIPSIN functionality to Openflow datapaths

Topography Manager

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Rendezvous

Node 4

Node 1

Node 2

Node 3

EWSDN 2012, 25-26 October, Darmstadt
Germany
Example for 5-port OpenFlow datapath

Each switch port is assigned a deployment unique Identifier by the Topology Manager which is kept at the local controller

Port 1: 1000 0000
Port 2: 0100 0000
Port 3: 0010 0000
Port 4: 0001 0000
Port 5: 0000 1000
Example for 5-port OpenFlow datapath

Openflow datapath is configured to match forwarding identifiers on each packet with respective delivery ports,

<table>
<thead>
<tr>
<th>Port</th>
<th>LIPSIN IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>1000 0000</td>
</tr>
<tr>
<td>2</td>
<td>0100 0000</td>
</tr>
<tr>
<td>3</td>
<td>0010 0000</td>
</tr>
<tr>
<td>4</td>
<td>0001 0000</td>
</tr>
<tr>
<td>5</td>
<td>0000 1000</td>
</tr>
</tbody>
</table>

EWSDN 2012, 25-26 October, Darmstadt Germany
Example for 5-port OpenFlow datapath

Openflow datapath sends to the local controller packets with forwarding identifiers that don’t match any entry.
Example for 5-port OpenFlow datapath

Local openflow controller uses the LIPSIN bloom-filter approach to decode the identifier and find the local datapath ports where the packet should be delivered and installs the rule.

<table>
<thead>
<tr>
<th>Port</th>
<th>INFORMATION IDENTIFIER</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0010 1000</td>
</tr>
<tr>
<td>2</td>
<td>0000 1000</td>
</tr>
<tr>
<td>3</td>
<td>0010 0000</td>
</tr>
<tr>
<td>4</td>
<td>0001 0000</td>
</tr>
<tr>
<td>5</td>
<td>0000 1000</td>
</tr>
</tbody>
</table>

FW Logic

Controller

OF Datapath

EWSDN 2012, 25-26 October, Darmstadt Germany
ICN architecture using SDN

Node 1
Node 2
Node 3
Node 4
Node 5

Topology Manager

Information Identifier
Node ID

FW Logic
Controller

OF Datapath 1

OF Datapath 2

OF1
OF2

1 2 5
3 4
ICN architecture using SDN Example

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Rendezvous

Node 4

Node 5

Topology Manager

OF Datapath 1
- Controller
- FW Logic

Node 1
- Port 1: 1000 0000
- Port 2: 0100 0000
- Port 3: 0010 0000
- Port 4: 0001 0000

Node 2

Node 3
- Port 1: 0000 1000
- Port 2: 0000 0100
- Port 3: 0000 0010

OF Datapath 2
- Controller
- FW Logic

Node 4

Node 5
ICN architecture using SDN Example

Node 1
Node 3
Node 5
Node 2
Node 4

Topology Manager

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

FW Logic
Controller

OF Datapath 1

Port 1: 1000 0000
Port 2: 0100 0000
Port 3: 0010 0000
Port 4: 0001 0000

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

0001 1000
Rv Req

OF1
OF2

1 2 5
3 4

Node 4

Node 2

Node 3
ICN architecture using SDN Example

Node 1
Node 2
Node 3
Node 4
Node 5

Topology Manager

OF Datapath 1
FW Logic
Controller

OF Datapath 2
FW Logic
Controller

Port 1: 1000 0000
Port 2: 0100 0000
Port 3: 0010 0000
Port 4: 0001 0000

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Rendezvous

0xABCD123
Pub
Node 1

Information
Identifier
Type
Node ID

0xABCD123
Pub
Node 1
ICN architecture using SDN Example

Topology Manager

Node 5

Information Identifier | Type | Node ID
--- | --- | ---
0xABCD123 | Pub | Node1
0xABCD123 | Sub | Node3

Node 4

0010 0010
TM Req

Node 1

Node 2

Node 3

OF1 - OF2

OF Datapath 1

FW Logic

Controller

Port 1: 1000 0000
Port 2: 0100 0000
Port 3: 0010 0000
Port 4: 0001 0000

OF Datapath 2

FW Logic

Controller

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Rendezvous

MATCH!
ICN architecture using SDN Example

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABCD123</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABCD123</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

Rendezvous: Node 4

Start Publish: 0001 0100

OF Datapath 1
- Port 1: 1000 0000
- Port 2: 0100 0000
- Port 3: 0010 0000
- Port 4: 0001 0000

OF Datapath 2
- Port 1: 0000 1000
- Port 2: 0000 0100
- Port 3: 0000 0010

FW Logic
- Controller

Node 1
- Pub

Node 2
- Sub

Node 3
- Sub
ICN architecture using SDN Example

**Node 1**: Node 1

**Node 2**: Node 2

**Node 3**: Node 3

**Node 4**: Node 4

**Node 5**: Node 5

**Controller**: Controller

**Rendezvous**: Rendezvous

**OF Datapath 1**: OF Datapath 1

**OF Datapath 2**: OF Datapath 2

**FW Logic**: FW Logic

**Topology Manager**: Topology Manager

**Information Identifier**

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>0xABCD123</td>
<td>Pub</td>
<td>Node1</td>
</tr>
<tr>
<td>0xABCD123</td>
<td>Sub</td>
<td>Node3</td>
</tr>
</tbody>
</table>

**Flow tables**

<table>
<thead>
<tr>
<th>FID</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000 0000</td>
<td>1</td>
</tr>
</tbody>
</table>

**Controller**

**Ports**

- Port 1: 1000 0000
- Port 2: 0100 0000
- Port 3: 0010 0000
- Port 4: 0001 0000

**Ports**

- Port 1: 1000 0000
- Port 2: 0100 0000
- Port 3: 0010 0000
- Port 4: 0001 0000

**Ports**

- Port 1: 0000 1000
- Port 2: 0000 0100
- Port 3: 0000 0010
ICN architecture using SDN Example

Node 1
Node 2
Node 3
Node 4
Node 5

Topology Manager

Information Identifier Type Node ID
0xABCD123 Pub Node1
0xABCD123 Sub Node3

Rendezvous

Node 4

OF Datapath 1
FW Logic
Controller

OF Datapath 2
FW Logic
Controller

OF 1 Flow tables

<table>
<thead>
<tr>
<th>FID</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000 0000</td>
<td>1</td>
</tr>
</tbody>
</table>

Port 1 1000 0000
Port 2 0100 0000
Port 3 0010 0000
Port 4 0001 0000

Node 2
Node 3

Sub

0001 0100 DATA

Pub

Port 1 0000 1000
Port 2 0000 0100
Port 3 0000 0010
ICN architecture using SDN Example

Node 1
Node 2
Node 3
Node 4
Node 5

Topology Manager

Information Identifier | Type | Node ID
---------------------|------|--------
0xABCD123            | Pub  | Node1  
0xABCD123            | Sub  | Node3  

Rendezvous

Node 4

OF Datapath 1

Controller

Port 1: 1000 0000
Port 2: 0100 0000
Port 3: 0010 0000
Port 4: 0001 0000

Data

0001 0100

OF 1 Flow tables

<table>
<thead>
<tr>
<th>FID</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000 0000</td>
<td>1</td>
</tr>
<tr>
<td>0001 0100</td>
<td>4</td>
</tr>
</tbody>
</table>

Part 1

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Part 2

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Part 3

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Part 4

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010

Part 5

Port 1: 0000 1000
Port 2: 0000 0100
Port 3: 0000 0010
ICN architecture using SDN Example

- **Node 1**
  - **Information Identifier**: 0xABCD123
  - **Type**: Pub
  - **Node ID**: Node1

- **Node 2**
  - **Information Identifier**: 0xABCD123
  - **Type**: Sub
  - **Node ID**: Node3

- **Node 3**
  - **Information Identifier**: 0xABCD123
  - **Type**: Pub
  - **Node ID**: Node1

- **Node 4**
  - **Information Identifier**: 0xABCD123
  - **Type**: Sub
  - **Node ID**: Node3

**Controller**
- **FW Logic**
- **OF Datapath 1**
  - Port 1: 1000 0000
  - Port 2: 0100 0000
  - Port 3: 0010 0000
  - Port 4: 0001 0000

**OF 1 Flow tables**

<table>
<thead>
<tr>
<th>FID</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>1000 0000</td>
<td>1</td>
</tr>
<tr>
<td>0001 0100</td>
<td>4</td>
</tr>
</tbody>
</table>

**OF Datapath 2**
- **Controller**
- **0001 0100 DATA**
- Port 1: 0000 1000
- Port 2: **0000 0100**
- Port 3: 0000 0010
Contents

• An Information – Centric Network Architecture
• An ICN Node Architecture
• Using LIPSIN for packet forwarding
• Designing a LIPSIN switch using Openflow and changes to ICN node architecture
• Benefits of using SDN support
Benefits of using SDN

- ICN node architecture gets simplified and forwarding is carried by the network and is completely decoupled from the nodes.
Benefits of using SDN

• Network bootstrap gets very simplified
Benefits of using SDN

- Network bootstrap gets very simplified
Benefits of using SDN

- Network bootstrap gets very simplified
Benefits of using SDN

• Network bootstrap gets very simplified
Benefits of using SDN

- Topology Management internal structures get simpler and response is improved
Future Work

• Use Multi-stage Bloom filters to avoid having different FID labels for the same delivery ports within a datapath
The problem

Topology Manager

<table>
<thead>
<tr>
<th>Information Identifier</th>
<th>Type</th>
<th>Node ID</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rendezvous</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Node 1

Node 2

Node 3

Node 4

Node 5

OF1

OF2

1 2 5

3 4

FW Logic

Controller

OF Datapath 1

Port 1 1000 0000
Port 2 0100 0000
Port 3 0010 0000
Port 4 0001 0000

OF 2 Flow tables

<table>
<thead>
<tr>
<th>FID</th>
<th>PORT</th>
</tr>
</thead>
<tbody>
<tr>
<td>0000 1000</td>
<td>1</td>
</tr>
<tr>
<td>0001 1000</td>
<td>1</td>
</tr>
</tbody>
</table>

OF Datapath 2

Port 1 0000 1000
Port 2 0000 0100
Port 3 0000 0010
Future Work

• Future work
  – Use SDN to simplify handover in ICN mobility
Thank You!

Pursuit BlackAdder Prototype:
https://github.com/fp7-pursuit/blackadder

Questions ?