# **Shortpaper:** IoT-NDN: An IoT Architecture via Named Data Netwoking (NDN)

Leonard Boetefuer

***Abstract*—als Leztes**

## I. Introduction

als vorleztes

## II. Related Work

XYZTEST2

## III. Analysis of IoT and NDN

This section will talk about the limitations of IoT devices and the challenges of the current Internet architecture.

### *A. The connectivity of IoT devices:*

Currently IoT devices use server-client or host-to-host connection to connect. In the server-client architecture every client has to communicate to the server and with a billion devices the server will be a massive bottleneck. In the host-host architecture, every host has to communicate to every other host. This results in exponential resource consumption. The server-client and the host-to-host model both need IP addresses for every single device, which is not possible with a billion devices.

### *B. Technological Standards:*

The crucial standards are for the network protocols, the communication protocols, and the data aggregation. The challenge is that

### *C. Mobility:*

The amount of mobile devices is rising and so are the challenges. The technologies of the mobile devices are divers and the IoT systems need to keep that in mind.

### *D. Complexity and Integration Issues:*

IoT systems are composed of many different APIs (Application Programming Interfaces), protocols and platforms. The integration of new technologies in the system is very complicated because of all the different combinations. The IoT system should consider the resource limitation of its components.

### *E. NDN Packet Length:*

Packages in NDN are not bound to a specific length, this is helpful to expand further protocols by adding or subtracting from the overhead. IoT devices will send only small packages; because of the limited resources in memory. The overhead needs to be kept small, because the information proportion of small packages is way more influenced by a big overhead.

### *F. Caching in IoT/NDN:*

To keep the data up to date and reduce unnecessary package flow, we need to integrate caches in to the system. The problem is that the small devices don't have enough memory to keep an efficient cache. The solution is to use in-network caching, a feature of NDN. Even a small cache will dramatically increase the data availability [11].

### *G. Data Aggregation in Wireless Networks:*

If a user's query requests for data that includes multiple packages, every single package has to be requested separately and excluding the others. This results in a greater overhead and will result in a more unnecessary package flow. The new system should fix the request problem.

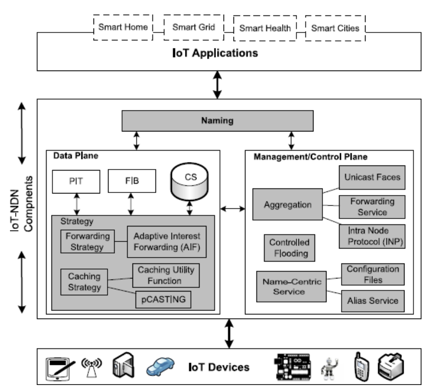
### *H. Naming Problems in Wireless Networks:*

NDN supports on name-centric services [17], which facilitates access without knowing their location. There is still a need to automate naming convention, because of the size of the Networks. These Names should be kept short to minimize storage usage.

### *I. Routing Scalability in NDN:*

In NDN routing is managed by names, instead of of usual number based systems. The scalability of routing is important to facilitate a large network.

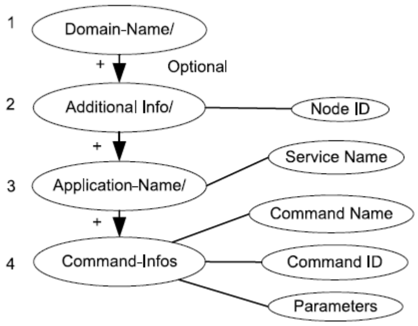
## IV. Architecture of IoT-NDN System




IoT-NDN has three main components. The **naming** component is made up of naming schemes and structure for wireless networks. The **Management and control plane** is made up of Unicast Faces; Forwardinng Services Intra Node Protocol, Controlled Flooding, Configuartion and Alias services. The **dataplan component** is made up of the caching and forwarding strategies. The IoT-NDN components are responsible for packages, caching, strategies and other. Devices in the IoT-NDN architecture have three tables: CS, PIT and FIB. NDN solves issues many issues discussed in III. For more information, see graphic 2.

| IoT-NDN Protocols & Algorithms | Implementation | Evaluation | References |
|---|---|---|---|
| Naming and NDN Protocols | ✓ | ✓ | [2], [6] |
| Name-Centric Services | ✓ | ✓ | [6] |
| Efficient Caching Algorithm | ✓ | ✓ | [4], [5] |
| Adaptive Internet Forwarding | ✓ | ✓ | [4] |
| Control Flooding | ✓ | ✓ | [4] |
| Data Aggregation Protocol | ✓ | ✓ | [3] |
| APIs for IoT-NDN | ✓ | ✓ | [7], [8] |

### A. Naming




In IoT-NDN the structure in which data is addressed is hierarchical. As shown in graphic 3. The first component is a global Domain-Name. The second is optional and, if used, stores additional information, like node ID. The third component contains the application and service name. The fourth component contains information about the command name, command ID and additional parameters. The marker component would be the additional fifth component of the name, specific to IoT-NDN. It contains application, service or device resource information. The marker component can

### B. Managment and Control Plane

1) **Aggregation:** Data aggregation is improtant to reduce the memory size of CS and the energy consumption, by combining similar information into one package. Data aggregation is achieved by three components. The components are: Forwarding Service, Unicast Faces and Intra Node Protocol (INP); they are explained in greater detail in [5]. Unicast Face needs every package, in the radio layer, to include the source and destination address. When a device receives a package, it will build a connection to the sender of the package. This enables devices to learn their neighbors. If a connection is lost, for this connection allocated resources will be released.

2) **Controlled Flooding:** IoT-devices are unreliable in power or connectivity. As a result the Forwarding Information Base (FIB) tables wont usually be populated in advance by routing information. Packages will transfer from one device to another and to mitigate flooding, controlled flooding is used. To reduce overhead and redundancy, devices will delay sending out packages. This time can be random or based on network topology. While a package is delayed and arrives again at the same device, the package is not send out twice. The path is selected

3) **Name-Centric Services:** The use of a Gateway enables IoT-Devices to be reachable from any internet device. The Gateway allows via protocol conversion wireless and wired connection. The Gateway provides all important configurations of names and IoT-NDN devices. Services on the gateway can be implemented as IoT-NDN applications and communicate to each other via the IoT-NDN deamon and face. To reduce the workload on the IoT devices the gateway will use aliases, that are shorter, to reduce the size of the packages. Names form the recived the internet will be mapped to names used in the Iot-NDN network.

### C. Data Plane

1) **Strategy-In-Network Caching:** The CS is a caching place in IoT-NDN devices, unlike Routers in the IP protocol it can send cached packages more than once. If receives a package request it, will first check the CS, by checking of matching prefixs. This is done because the same data will probably be requested many times in IoT-NDN networks. The standard replacement strategy is LRU, but others can be implemented. Iot-NDN devices have a special probabilistic CAching STrategy (pCASTING), that considers data freshness and the charge and storage of devices. This strategy is used when a data package with a matching PIT is received by a device.

2) **Strategy-Forwarding:** The forwarding Path is selected from the FIB to forward Interest packets, by the forwarding strategy component. by remembering the number of unsatisfied Interests, the forwarding strategy could be used to control the traffic. The forwarding component the path of the Interest by using data such as delay and throughput. Forwarding steps on devices are supported by IoT-NDN and the forwarding strategy allows the requesting of lost packages in network, by using its metrics. These metrics can be used to study the performance of every face. Fining missing packages can also be achieved with the InterestLifeTime parameter. For more info see [4]

## V. Conclusion

## VI. References

## References

[1] M. A. Heil, "IoT-NDN: An IoT Architecture via Named Data Netwoking (NDN)" in Software Embedded System Department Euroimmun AG, a PerkinElmer Company

[2] Z. Ren, M. Hail, and H. Hellbruck, "Ccn-wsn - a lightweight, flexible content-centric networking protocol for wireless sensor networks," in Intelligent Sensors, Sensor Networks and Information Processing, 2013.

[3] T. Teubler, M. A. M. Hail, and H. Hellbr¨uck, "Efficient Data Aggregation with CCNx in Wireless Sensor Networks," in 19th EUNICE Workshop on Advances in Communication Networking (EUNICE 2013), Germany.

[4] M. A. Hail, M. Amadeo, A. Molinaro, and S. Fischer, "On the performance of caching and forwarding in information centric networking for the iot," in 13th International Conference on Wired and Wireless Internet Communications, April 2015.

[5] ——, "Caching in named data networking for the wireless internet of things," in Recent Advances in Internet of Things (RIoT), 2015 International Conference on, April 2015, pp. 1–6.

[6] T. Teubler, M. A. M. Hail, and H. Hellbr¨uck, “A solution for the naming problem for name-centric services,” in Wired/Wireless Internet Communications, A. Mellouk, S. Fowler, S. Hoceini, and B. Daachi, Eds. Cham: Springer International Publishing, 2014, pp. 214–227.
[7] S. Ebers, M. A. Hail, S. Fischer, and H. Hellbr¨uck, “Api for data dissemination protocols - evaluation with autocast,” in The Third World Congress on Nature and Biologically Inspired Computing (NaBIC 2011). Salamanca, Spain: IEEE, Oct. 2011, pp. 534–539.
[8] M. A. Hail and S. Fischer, “Flexible API for IoT Services with Named Data Networking,” in 2016 IEEE International Conference on Emerging Technologies and Innovative Business Practices for the Transformation of Societies (IEEE EmergiTech 2016), Mauritius, Mauritius, Aug 2016.