

July 24rd, 2026

SCHC Architecture

Draft : draft-ietf-schc-architecture-06

Authors : A. Pelov , P. Thubert , A. Minaburo , Q. Lampin , M. Dumay

IETF 126, Vienna



How do the drafts work with the architecture?

SCHC Framework Core + Extensions

[RFC 8724 SCHC Framework](#)

[RFC 9441 Compound Ack](#)

[draft-ietf-schc-universal-option-01](#)

[draft-corneo-schc-compress-payload-02](#)

[draft-corneo-schc-ctx-mgmt-00](#)

[draft-westerlund-schc-compute-address-00](#)

SCHC Interoperability

[RFC 9363 YANG Data Model](#)

[draft-ietf-schc-schclet-00](#)

[draft-ietf-schc-protocol-numbers-06](#)

[draft-lampin-voici-02](#)

[draft-pelov-schc-header-format-00](#)

SCHC Adapted to

[RFC 9011 LoRaWAN](#)

[RFC 9391 NB-IOT](#)

[RFC 9442 Sigfox](#)

[draft-ietf-schc-over-networks-prone-to-disruptions-03](#)

[draft-ietf-schc-over-ppp-00](#)

SCHC Applied to

[RFC 8824 CoAP](#)

[draft-ietf-ipsecme-diet-esp-10](#)

[draft-ietf-schc-icmpv6-compression-02](#)

[draft-ietf-schc-8824-update-09](#)

[draft-sirohi-schc-quic-frame-compression-00](#)

[draft-ietf-schc-architecture-06](#)

Informational



- Document SCHC (Generic: LPWAN and other uses)
- Clear terminology
- Cover simple, as well as more advanced use-cases
- Entry-point for non-SCHC people (KISS)



The Contents

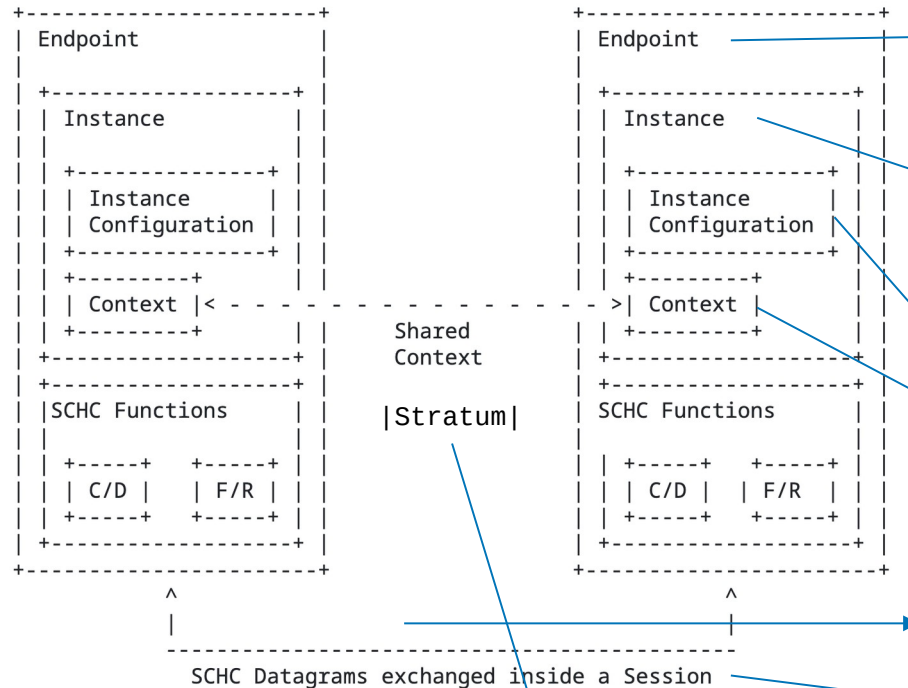
A new structure designed to introduce the main entities and their relationships right from the first few pages

1.	Introduction	3
2.	Requirements Language	4
3.	Terminology	4
4.	Architecture	6
4.1.	Overview of a Basic Architecture	6
4.1.1.	SCHC: Quick Reminders	6
4.1.2.	Basic SCHC Architecture	6
4.2.	Focus on core components	9
4.2.1.	Instance	10
4.2.2.	Endpoint	11
4.2.3.	Session	13
4.2.4.	Domain	14
4.2.5.	Datagram Format	15
5.	Deployment Profiles	16
6.	Operational considerations	16
6.1.	Error handling	17
6.2.	Context consistency	17
7.	Security Considerations	17
8.	IANA Considerations	18
9.	References	18
9.1.	Normative References	18
9.2.	Informative References	18

Appendix A.	Examples - To Be Validated	20
A.1.	Control Header Examples	20
A.2.	Deployment Models	21
A.2.1.	LPWAN deployment	21
A.2.2.	PPP deployment	25
A.2.3.	Direct transport over Ethernet, IPv6, and UDP	26
A.3.	Compatible Partial Contexts	27
Appendix B.	Future Work - To Be Decided by the Working Group	28
B.1.	C/D Engine Interface	28
B.2.	Other Open Items	28



Basic SCHC Architecture



A logical entity that provides SCHC functionality by hosting the SCHC processing code.

One or more can be present on a physical device.

A logical component of an Endpoint that executes the actual SCHC operations (C/D, F&R)

One or more can be present on an endpoint.

Local parameters

SoR + Metadata

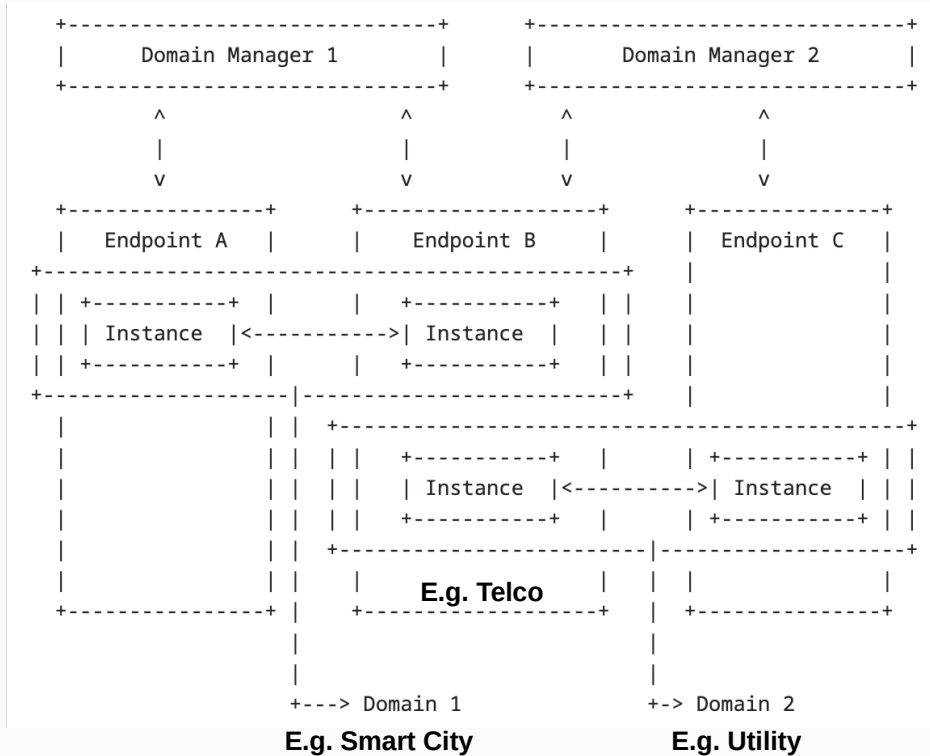
The unit exchanged between SCHC Instances.

Dynamic or implicit communication sessions

Background concept, portion of the network protocol stack targeted by SCHC



Multi-Tenants Scenario



Domain: A logical grouping of Instances that share a common Context.

Domain Manager: manages the **Domain**, including Context synchronization and configuration distribution.

TBD in future version: context management operation.

1 Gateway, 2 IoT devices, different Tenants with different Contexts

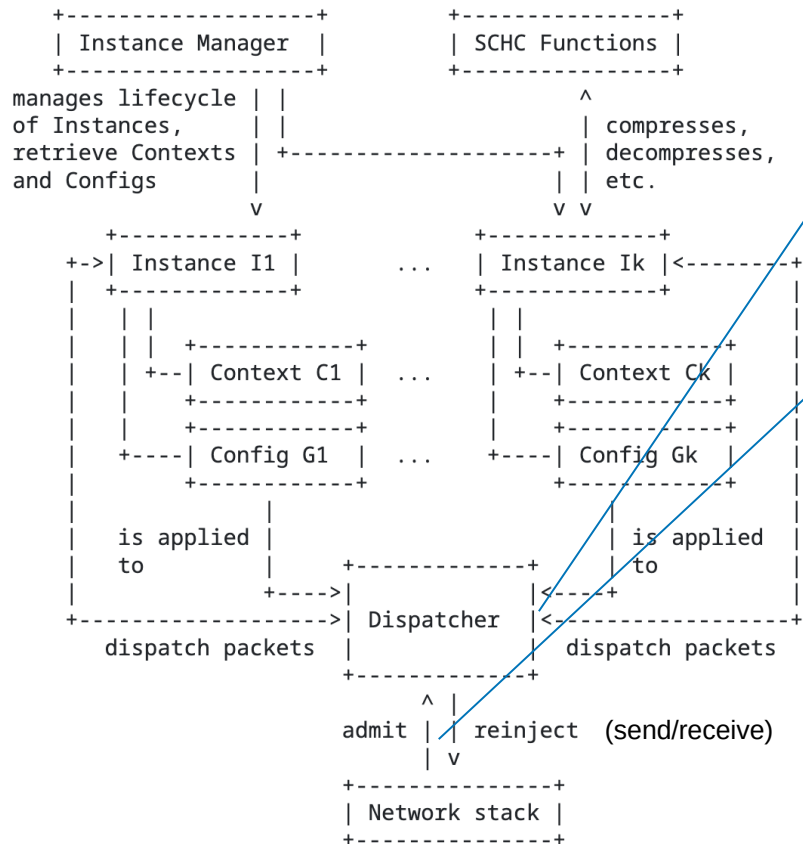
A single SCHC Instance runs on each IoT Endpoint (A, C).

Two Instances run on the Gateway (B)

- Single Domain per IoT Endpoint
- Two Domains on Gateway
- Discriminator is provided by L2, e.g., DevEUI.



Multiple Instances on an Endpoint



Dispatcher: A logical component of the Endpoint that routes Datagrams to the appropriate Instances based on defined admission rules.

It can be integrated into the network stack or implemented as a separate component.

Discriminator: An optional information element used by the Dispatcher to route SCHC Datagrams to the appropriate Instance. The discriminator can be a combination of several criteria.

From the Side Meeting:

- *may be constituted by one or more elements.*
- *should not require knowledge of the SCHC context to be interpreted.*

(otherwise, chicken-and-egg problem)

SCHC Datagram Data Unit Format

Datagram Data Unit Format

```
+-----+-----+
| RuleID | Rule-dependent fields and payload |
+-----+-----+
```

A Data Unit starts with a RuleID. The Rule identified by that RuleID determines the format and interpretation of the remaining bits. A Data Unit may be an unfragmented SCHC Packet or a SCHC F/R message, such as a SCHC Fragment, SCHC ACK, SCHC ACK Request, SCHC Sender-Abort, or SCHC Receiver-Abort.

Control Header for Advanced Use Cases

```
+-----+-----+
| RuleID | Control Header | Rule-dependent fields and payload |
+-----+-----+
```

or

```
+-----+-----+
| Control Header | RuleID | Rule-dependent fields and payload |
+-----+-----+
```

The presence, placement, and format of the Control Header must be clearly identified, e.g., by a SCHC profile document or other specification that defines the framing used by the deployment.

Example of a control header for generic multiplexing use-cases is given in: [draft-lampin-voici-02](#)

It is expected there to be several such standard-track documents that specify different control headers.



SCHC Header and SCHC Header Format (not in -06 !)

- **SCHC Header:** The portion of a SCHC Data Unit that carries SCHC processing information, including the RuleID and, when present, Control Header or F/R control fields. The SCHC Header enables the receiver to identify the Rule and interpret the remaining **SCHC Payload**, fragment, residue, or control information. Its exact encoding is defined by the applicable Rule, profile, or encapsulation.
- **SCHC Header Format:** A profile-, encapsulation-, or deployment-specific description of the SCHC Header encoding, including the presence, order, and encoding of the RuleID and any Control Header or F/R control fields.

Example of how this can be implemented: [draft-pelov-schc-header-format-00](#)



Take-away from today

How -06 impacts existing documents

- Automatically processed 7 RFCs and 17 drafts against -06 version (Claude, Gemini, meta-analysis with Codex)

For all 7 RFCs - all concepts, roles, communication patterns, rule structures, and fragmentation mechanisms map directly or compositely to SCHC architecture -06

All 17 drafts map onto the architecture with High or Very High conceptual equivalence

14 Easy, 3 Medium, 0 Hard - the bulk is terminology alignment, not technical rework

- With diff suggestions

- <https://github.com/lp-wan/archi/tree/main/archi-06-document-analysis>

- The work continues

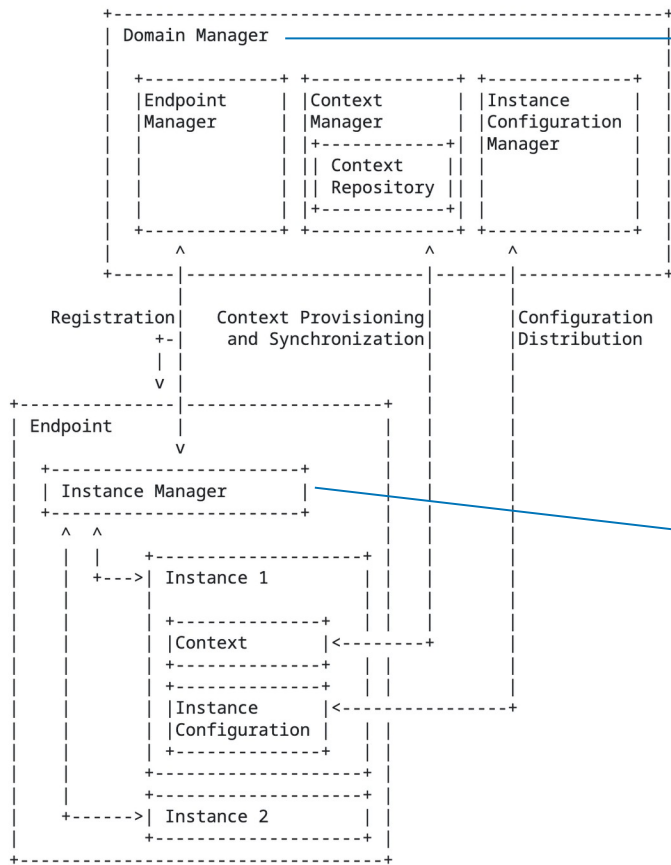


Thank you!

- End of the work of the Design Team?
 - Produce -07 with minimal changes (as presented)
- Questions ?



Domain



Domain: A logical grouping of Instances that share a common Context.

Domain Manager: manages the **Domain**, including Context synchronization and configuration distribution.

It uses:

Endpoint Manager to manage the **Endpoints**

Context Manager to manage the **Contexts**

Instance Conf Manager to manage **Instance Configuration**

Domain vs. Session

The Domain defines which Instances share Contexts, while the Session identifies which Instances are actually communicating.

A logical component that manages the lifecycle and configuration of Instances within an Endpoint.

E.g. creating, updating, and deleting Instances as needed.

Note: These are logical components.

They may be implemented in a single software for example.

The domain may be peer-to-peer.

➔ Next revision needs to specify this use-case