A Media Type Structured Syntax Suffix for JSON Text Sequences

draft-wilde-json-seq-suffix-00

Abstract

Structured Syntax Suffixes for media types allow other media types to build on them and make it explicit that they are built on an existing media type as their foundation. This specification defines and registers "json-seq" as a structured syntax suffix for JSON Text Sequences.

Note to Readers

This draft should be discussed on the art mailing list [1].

Online access to all versions and files is available on github [2].

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on March 27, 2017.

Copyright Notice

Copyright (c) 2016 IETF Trust and the persons identified as the document authors. All rights reserved.

This document is subject to BCP 78 and the IETF Trust’s Legal Provisions Relating to IETF Documents (http://trustee.ietf.org/license-info) in effect on the date of
1. Introduction

Media Type Structured Syntax Suffixes [2] were introduced as a way of how a media type can signal that it is based on another media type as its foundation. Some structured syntax suffixes were registered initially [5], including "+json" for the widely popular JSON Format [4] format.

JSON Text Sequences [3] is a new specification in the JSON space that defines how a sequence of multiple JSON texts can be represented in one representation. Since this specification can be used as the foundation for other formats, this specification defines and registers the "+json-seq" structured syntax suffix.

2. Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1].

3. Using +json-seq

The use case for the "+json-seq" structured syntax suffix is the same as for "+json": It SHOULD be used by media types when parsing the JSON Text Sequence of a media type leads to a meaningful result, by simply using the generic JSON Text Sequence processing.

Applications encountering such a media type then can either simply use generic processing if all they need is a generic view of the JSON Text Sequence, or they can use generic JSON Text Sequence tools for
initial parsing, and then can implement their own specific processing on top of that generic parsing tool.

4. IANA Considerations

IANA has added the following "+json-seq" structured syntax suffix to its registry of structured syntax suffixes established by [2]:

Name: JSON Text Sequence
+suffix: +json-seq

References: [3]

Encoding considerations: See [3]

Fragment identifier considerations: The syntax and semantics of fragment identifiers specified for +json-seq SHOULD be as specified for "application/json-seq". (At publication of this document, there is no fragment identification syntax defined for "application/json-seq".)

The syntax and semantics for fragment identifiers for a specific "xxx/yyy+json-seq" SHOULD be processed as follows:

For cases defined in +json-seq, where the fragment identifier resolves per the +json-seq rules, then process as specified in +json-seq.

For cases defined in +json-seq, where the fragment identifier does not resolve per the +json-seq rules, then process as specified in "xxx/yyy+json-seq".

For cases not defined in +json-seq, then process as specified in "xxx/yyy+json-seq".

Interoperability considerations: n/a

Security considerations: See [3]

Contact: Applications and Real-Time Area Working Group (art@ietf.org)

Author/Change controller: The Applications and Real-Time Area Working Group. IESG has change control over this registration.
5. References

5.1. Normative References


5.2. Non-Normative References


Author’s Address

Erik Wilde
CA Technologies

Email: erik.wilde@dret.net
URI:  http://dret.net/netdret/