

# EXHIBIT 1

Internet Engineering Task Force (IETF)  
Request for Comments: 8259  
Obsoletes: 7159  
Category: Standards Track  
ISSN: 2070-1721

T. Bray, Ed.  
Textuality  
December 2017

## The JavaScript Object Notation (JSON) Data Interchange Format

### Abstract

JavaScript Object Notation (JSON) is a lightweight, text-based, language-independent data interchange format. It was derived from the ECMAScript Programming Language Standard. JSON defines a small set of formatting rules for the portable representation of structured data.

This document removes inconsistencies with other specifications of JSON, repairs specification errors, and offers experience-based interoperability guidance.

### Status of This Memo

This is an Internet Standards Track document.

This document is a product of the Internet Engineering Task Force (IETF). It represents the consensus of the IETF community. It has received public review and has been approved for publication by the Internet Engineering Steering Group (IESG). Further information on Internet Standards is available in Section 2 of RFC 7841.

Information about the current status of this document, any errata, and how to provide feedback on it may be obtained at <https://www.rfc-editor.org/info/rfc8259>.

## Copyright Notice

Copyright (c) 2017 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 (<https://trustee.ietf.org/license-info>) in effect on the date of publication of this document. Please review these documents carefully, as they describe your rights and restrictions with respect to this document. Code Components extracted from this document must include Simplified BSD License text as described in Section 4.e of the Trust Legal Provisions and are provided without warranty as described in the Simplified BSD License.

This document may contain material from IETF Documents or IETF Contributions published or made publicly available before November 10, 2008. The person(s) controlling the copyright in some of this material may not have granted the IETF Trust the right to allow modifications of such material outside the IETF Standards Process. Without obtaining an adequate license from the person(s) controlling the copyright in such materials, this document may not be modified outside the IETF Standards Process, and derivative works of it may not be created outside the IETF Standards Process, except to format it for publication as an RFC or to translate it into languages other than English.

## Table of Contents

|  |    |
|--|----|
| 1. Introduction . . . . .                        | 3  |
| 1.1. Conventions Used in This Document . . . . . | 4  |
| 1.2. Specifications of JSON . . . . .            | 4  |
| 1.3. Introduction to This Revision . . . . .     | 5  |
| 2. JSON Grammar . . . . .                        | 5  |
| 3. Values . . . . .                              | 6  |
| 4. Objects . . . . .                             | 6  |
| 5. Arrays . . . . .                              | 7  |
| 6. Numbers . . . . .                             | 7  |
| 7. Strings . . . . .                             | 8  |
| 8. String and Character Issues . . . . .         | 9  |
| 8.1. Character Encoding . . . . .                | 9  |
| 8.2. Unicode Characters . . . . .                | 10 |
| 8.3. String Comparison . . . . .                 | 10 |
| 9. Parsers . . . . .                             | 10 |
| 10. Generators . . . . .                         | 10 |
| 11. IANA Considerations . . . . .                | 11 |
| 12. Security Considerations . . . . .            | 12 |
| 13. Examples . . . . .                           | 12 |
| 14. References . . . . .                         | 14 |
| 14.1. Normative References . . . . .             | 14 |
| 14.2. Informative References . . . . .           | 14 |
| Appendix A. Changes from RFC 7159 . . . . .      | 16 |
| Contributors . . . . .                           | 16 |
| Author's Address . . . . .                       | 16 |

**1. Introduction**

JavaScript Object Notation (JSON) is a text format for the serialization of structured data. It is derived from the object literals of JavaScript, as defined in the ECMAScript Programming Language Standard, Third Edition [ECMA-262].

JSON can represent four primitive types (strings, numbers, booleans, and null) and two structured types (objects and arrays).

A string is a sequence of zero or more Unicode characters [UNICODE]. Note that this citation references the latest version of Unicode rather than a specific release. It is not expected that future changes in the Unicode specification will impact the syntax of JSON.

An object is an unordered collection of zero or more name/value pairs, where a name is a string and a value is a string, number, boolean, null, object, or array.

An array is an ordered sequence of zero or more values.

The terms "object" and "array" come from the conventions of JavaScript.

JSON's design goals were for it to be minimal, portable, textual, and a subset of JavaScript.

### 1.1. Conventions Used in This Document

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

The grammatical rules in this document are to be interpreted as described in [RFC5234].

### 1.2. Specifications of JSON

This document replaces [RFC7159]. [RFC7159] obsoleted [RFC4627], which originally described JSON and registered the media type "application/json".

JSON is also described in [ECMA-404].

The reference to ECMA-404 in the previous sentence is normative, not with the usual meaning that implementors need to consult it in order to understand this document, but to emphasize that there are no inconsistencies in the definition of the term "JSON text" in any of its specifications. Note, however, that ECMA-404 allows several practices that this specification recommends avoiding in the interests of maximal interoperability.

The intent is that the grammar is the same between the two documents, although different descriptions are used. If there is a difference found between them, ECMA and the IETF will work together to update both documents.

If an error is found with either document, the other should be examined to see if it has a similar error; if it does, it should be fixed, if possible.

If either document is changed in the future, ECMA and the IETF will work together to ensure that the two documents stay aligned through the change.

# Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

## Real-Time Litigation Alerts



Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

## Advanced Docket Research



With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

## Analytics At Your Fingertips



Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

## API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

## LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

## FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

## E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.