                    ECMAScript Media Types Updates
                   draft-ietf-dispatch-javascript-mjs-04

Abstract

   This document proposes updates to the ECMAScript media types,
   superseding the existing registrations for "application/javascript"
   and "text/javascript" by adding an additional extension and removing
   usage warnings. This document updates RFC4329, "Scripting Media
   Types".

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 https://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 December 21, 2019.

Copyright Notice

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.

Table of Contents

1. Introduction

This document updates the existing media types for the ECMAScript
programming language. It supersedes the media types registrations in
[RFC4329] for "application/javascript" and "text/javascript".

2. Background

In order to formalize support for modular programs [ECMA-262] now
defines two top-level goal symbols for the ECMAScript grammar. This
means that (in the absence of additional information) there are two
possible interpretations for any given ECMAScript Source Text. The
TC39 standards body for ECMAScript has determined that media types
are outside of their scope of work [TC39-MIME-ISSUE].

It is not possible to fully determine if a Source Text of ECMAScript
is meant to be parsed in the Module or Script grammar goals based
upon content alone. Therefore, scripting environments must use out
of band information in order to determine what goal a Source Text
should be treated as. To this end some scripting environments have
chosen to adopt a new file extension of .mjs for determining the goal
of a given Source Text.

3. IANA Considerations

The media type registrations herein are divided into two major
categories: the sole media type "text/javascript" which is now in
common usage, and all of the media types that are obsolete.

For both categories, The ECMAScript media types are to be updated to
point to a non-vendor specific standard undated specification of
ECMAScript. In addition, a new file extension of .mjs is to be added
to the list of file extensions with the restriction that it must
correspond to the Module grammar of [ECMA-262]. Finally, the [HTML]
specification is using "text/javascript" as the default media type of
ECMAScript when preparing script tags; therefore, "text/javascript"
has been moved intended usage from OBSOLETE to COMMON.

3.1. Common Javascript Media Types

3.1.1. text/javascript

Type name: text

Subtype name: javascript

Required parameters: none

Optional parameters: charset, see section 4.1 of [RFC4329].

Encoding considerations: Encoding is host dependent with differences
    in byte order marks, the charset parameter, and text
    preprocessing.

Security considerations: See section 5 of [RFC4329].

Interoperability considerations: See notes in various sections of
    [RFC4329].

Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
    discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: COMMON

   Restrictions on usage: The .mjs file extension signals that the file
      represents a JavaScript module. Execution environments that rely
      on file extensions to determine how to process inputs parse .mjs
      files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2. Historic Javascript Media Types

   The following media types are added or updated for historical
   purposes. All herein have an intended usage of OBSOLETE, and are not
   expected to be in use with modern implementations.

3.2.1. application/ecmascript

   Type name: application

   Subtype name: ecmascript

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
    discussed in [RFC4329].

Additional information:

    Magic number(s): n/a

    File extension(s): .es, .mjs

    Macintosh File Type Code(s): TEXT

Person & email address to contact for further information: See
    Author's Address section.

Intended usage: OBSOLETE

Restrictions on usage: This media type is obsolete; current
    implementations should use text/javascript as the only JavaScript/
    ECMAScript media type. The .mjs file extension signals that the
    file represents a JavaScript module. Execution environments that
    rely on file extensions to determine how to process inputs parse
    .mjs files using the Module grammar of [ECMA-262].

Author: See Author's Address section.

Change controller: IESG <iesg@ietf.org>

3.2.2. application/javascript

Type name: application

Subtype name: javascript

Required parameters: none

Optional parameters: charset, see section 4.1 of [RFC4329].

Encoding considerations: Encoding is host dependent with differences
    in byte order marks, the charset parameter, and text
    preprocessing.

Security considerations: See section 5 of [RFC4329].

Interoperability considerations: See notes in various sections of
    [RFC4329].

Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
   discussed in [RFC4329].

Additional information:

   Magic number(s): n/a

   File extension(s): .js, .mjs

   Macintosh File Type Code(s): TEXT

Person & email address to contact for further information: See
   Author's Address section.

Intended usage: OBSOLETE

Restrictions on usage: This media type is obsolete; current
   implementations should use text/javascript as the only JavaScript/
   ECMAScript media type. The .mjs file extension signals that the
   file represents a JavaScript module. Execution environments that
   rely on file extensions to determine how to process inputs parse
   .mjs files using the Module grammar of [ECMA-262].

Author: See Author's Address section.

Change controller: IESG <iesg@ietf.org>.

3.2.3. application/x-ecmascript

Type name: application

Subtype name: x-ecmascript

Required parameters: none

Optional parameters: charset, see section 4.1 of [RFC4329].

Encoding considerations: Encoding is host dependent with differences
   in byte order marks, the charset parameter, and text
   preprocessing.

Security considerations: See section 5 of [RFC4329].

Interoperability considerations: See notes in various sections of
   [RFC4329].

Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .es, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.4. application/x-javascript

   Type name: application

   Subtype name: x-javascript

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.5. text/ecmascript

   Type name: text

   Subtype name: ecmascript

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .es, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.6. text/javascript1.0

   Type name: text

   Subtype name: javascript1.0

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.7. text/javascript1.1

   Type name: text

   Subtype name: javascript1.1

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.8. text/javascript1.2

   Type name: text

   Subtype name: javascript1.2

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.9. text/javascript1.3

   Type name: text

   Subtype name: javascript1.3

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.10. text/javascript1.4

   Type name: text

   Subtype name: javascript1.4

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.11. text/javascript1.5

   Type name: text

   Subtype name: javascript1.5

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: This media type is obsolete; current
      implementations should use text/javascript as the only JavaScript/
      ECMAScript media type. The .mjs file extension signals that the
      file represents a JavaScript module. Execution environments that
      rely on file extensions to determine how to process inputs parse
      .mjs files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.12. text/jscript

   Type name: text

   Subtype name: jscript

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
    discussed in [RFC4329].

Additional information:

    Magic number(s): n/a

    File extension(s): .js, .mjs

    Macintosh File Type Code(s): TEXT

Person & email address to contact for further information: See
    Author's Address section.

Intended usage: OBSOLETE

Restrictions on usage: The .mjs file extension signals that the file
    represents a JavaScript module. Execution environments that rely
    on file extensions to determine how to process inputs parse .mjs
    files using the Module grammar of [ECMA-262].

Author: See Author's Address section.

Change controller: IESG <iesg@ietf.org>

3.2.13. text/livescript

Type name: text

Subtype name: livescript

Required parameters: none

Optional parameters: charset, see section 4.1 of [RFC4329].

Encoding considerations: Encoding is host dependent with differences
    in byte order marks, the charset parameter, and text
    preprocessing.

Security considerations: See section 5 of [RFC4329].

Interoperability considerations: See notes in various sections of
    [RFC4329].

Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
    discussed in [RFC4329].

   Additional information:

      Magic number(s): n/a

      File extension(s): .js, .mjs

      Macintosh File Type Code(s): TEXT

   Person & email address to contact for further information: See
      Author's Address section.

   Intended usage: OBSOLETE

   Restrictions on usage: The .mjs file extension signals that the file
      represents a JavaScript module. Execution environments that rely
      on file extensions to determine how to process inputs parse .mjs
      files using the Module grammar of [ECMA-262].

   Author: See Author's Address section.

   Change controller: IESG <iesg@ietf.org>

3.2.14. text/x-ecmascript

   Type name: text

   Subtype name: x-ecmascript

   Required parameters: none

   Optional parameters: charset, see section 4.1 of [RFC4329].

   Encoding considerations: Encoding is host dependent with differences
      in byte order marks, the charset parameter, and text
      preprocessing.

   Security considerations: See section 5 of [RFC4329].

   Interoperability considerations: See notes in various sections of
      [RFC4329].

   Published specification: [[RFCXXXX]]

   Applications which use this media type: Script interpreters as
      discussed in [RFC4329].

   Additional information:

Magic number(s): n/a

File extension(s): .es, .mjs

Macintosh File Type Code(s): TEXT

Person & email address to contact for further information: See
Author's Address section.

Intended usage: OBSOLETE

Restrictions on usage: The .mjs file extension signals that the file
represents a JavaScript module. Execution environments that rely
on file extensions to determine how to process inputs parse .mjs
files using the Module grammar of [ECMA-262].

Author: See Author's Address section.

Change controller: IESG <iesg@ietf.org>

3.2.15. text/x-javascript

Type name: text

Subtype name: x-javascript

Required parameters: none

Optional parameters: charset, see section 4.1 of [RFC4329].

Encoding considerations: Encoding is host dependent with differences
in byte order marks, the charset parameter, and text
preprocessing.

Security considerations: See section 5 of [RFC4329].

Interoperability considerations: See notes in various sections of
[RFC4329].

Published specification: [[RFCXXXX]]

Applications which use this media type: Script interpreters as
discussed in [RFC4329].

Additional information:

Magic number(s): n/a

   File extension(s): .js, .mjs

   Macintosh File Type Code(s): TEXT

Person & email address to contact for further information: See
   Author's Address section.

Intended usage: OBSOLETE

Restrictions on usage: The .mjs file extension signals that the file
   represents a JavaScript module. Execution environments that rely
   on file extensions to determine how to process inputs parse .mjs
   files using the Module grammar of [ECMA-262].

Author: See Author's Address section.

Change controller: IESG <iesg@ietf.org>

4. References

4.1. Normative References

[ECMA-262]
            Ecma International, "Standard ECMA-262: ECMAScript
            Language Specification", August 2017, <https://ecma-
            international.org/publications/standards/Ecma-262.htm>.

[RFC4329] Hoehrmann, B., "Scripting Media Types", RFC 4329,
            DOI 10.17487/RFC4329, April 2006,
            <https://www.rfc-editor.org/info/rfc4329>.

4.2. Informative References

[HTML]   WHATWG, "HTML Living Standard", August 2017,
            <https://html.spec.whatwg.org/multipage/
            scripting.html#prepare-a-script>.

[TC39-MIME-ISSUE]
            TC39, "Add `application/javascript+module` mime to remove
            ambiguity", August 2017, <https://web.archive.org/web/2017
            0814193912/https://github.com/tc39/ecma262/issues/322>.

Appendix A. Acknowledgements

   The authors would like to thank Suresh Krishnan, Alexey Melnikov,
   Mark Nottingham, James Snell, Adam Roach, and Allen Wirfs-Brock for
   their guidance throughout this process.

Authors' Addresses

    Myles Borins
    Google

    Email: mylesborins@google.com

    Mathias Bynens
    Google

    Email: mths@google.com

    Matthew A. Miller
    Mozilla

    Email: linuxwolf+ietf@outer-planes.net

    Bradley Farias

    Email: bradley.meck@gmail.com