[MS-WAI-ARIA-11]:

Microsoft Edge Accessible Rich Internet Applications (WAI-ARIA) 1.1 Standards Support Document

Revision Summary

Date	Revision History	Revision Class	Comments
6/5/2018	1.0	New	Released new document.
8/28/2018	1.1	Minor	Clarified the meaning of the technical content.

Table of Contents

1 Introduction

This document describes the level of support provided by Microsoft Edge for the *Accessible Rich Internet Applications (WAI-ARIA) 1.1* specification [W3C-WAI-ARIA-11], published 14 December 2017. The [W3C-WAI-ARIA-11] specification provides an ontology of roles, states, and properties that define accessible user interface elements and can be used to improve the accessibility and interoperability of web content and applications.

1.1 Glossary

MAY, SHOULD, MUST, SHOULD NOT, MUST NOT: These terms (in all caps) are used as defined in [RFC2119]. All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

1.2 References

Links to a document in the Microsoft Open Specifications library point to the correct section in the most recently published version of the referenced document. However, because individual documents in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact dochelp@microsoft.com. We will assist you in finding the relevant information.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, http://www.rfc-editor.org/rfc/rfc2119.txt

[W3C-WAI-ARIA-11] World Wide Web Consortium, "Accessible Rich Internet Applications (WAI-ARIA) 1.1", W3C Recommendation 14 December 2017, https://www.w3.org/TR/2017/REC-wai-aria-1.1-20171214/

1.2.2 Informative References

None.

1.3 Microsoft Implementations

The following Microsoft web browsers implement some portion of the [W3C-WAI-ARIA-11] specification:

- Microsoft Edge

Each browser version may implement multiple document rendering modes. The modes vary from one to another in support of the standard. The following table lists the document modes supported by each browser version.

Browser Version	Document Modes Supported
Microsoft Edge	EdgeHTML Mode

For each variation presented in this document there is a list of the document modes and browser versions that exhibit the behavior described by the variation. All combinations of modes and versions that are not listed conform to the specification. For example, the following list for a variation indicates that the variation exists in three document modes in all browser versions that support these modes:

Quirks Mode, IE7 Mode, and IE8 Mode (All Versions)

1.4 Standards Support Requirements

To conform to [W3C-WAI-ARIA-11], a user agent must implement all required portions of the specification. Any optional portions that have been implemented must also be implemented as described by the specification. Normative language is usually used to define both required and optional portions. (For more information, see [RFC2119].)

The following table lists the sections of [W3C-WAI-ARIA-11] and whether they are considered normative or informative.

Sections	Normative/Informative
1-2	Informative
3	Normative
4	Informative
5-7,	Normative
All appendices	Informative

1.5 Notation

The following notations are used in this document to differentiate between notes of clarification, variation from the specification, and points of extensibility.

Notation	Explanation
C####	This identifies a clarification of ambiguity in the target specification. This includes imprecise statements, omitted information, discrepancies, and errata. This does not include data formatting clarifications.
V####	This identifies an intended point of variability in the target specification such as the use of MAY, SHOULD, or RECOMMENDED. (See [RFC2119].) This does not include extensibility points.
E####	Because the use of extensibility points (such as optional implementation-specific data) can impair interoperability, this profile identifies such points in the target specification.

For document mode and browser version notation, see also section 1.3.

2 Standards Support Statements

This section contains all variations, clarifications, and extensions for the Microsoft implementation of [W3C-WAI-ARIA-11].

- Section 2.1 describes normative variations from the MUST requirements of the specification.
- Section 2.2 describes clarifications of the MAY and SHOULD requirements.
- Section 2.3 describes extensions to the requirements.
- Section 2.4 considers error handling aspects of the implementation.
- Section 2.5 considers security aspects of the implementation.

2.1 Normative Variations

There are no normative variations from the MUST requirements of [W3C-WAI-ARIA-11].

2.2 Clarifications

The following subsections describe clarifications of the MAY and SHOULD requirements of [W3C-WAI-ARIA-11].

2.2.1 [W3C-WAI-ARIA-11] Section 5. The Roles Model

C0001: The user agent maps the role attribute to the appropriate value in the accessibility API and updates the mapping when the attribute changes.

The specification states:

```
5. The Roles Model

    This section defines the WAI-ARIA role taxonomy and describes the characteristics and
    properties of all roles. ...
    ...
    In order to reflect the content in the DOM, user agents SHOULD map the role attribute
    to the appropriate value in the implemented accessibility API, and user agents SHOULD
    update the mapping when the role attribute changes.
```

EdgeHTML Mode

The user agent maps the role attribute to the appropriate value in the accessibility API and updates the mapping when the attribute changes.

2.2.2 [W3C-WAI-ARIA-11] Section 5.2.8. Presentational Children

C0002: The user agent does not expose the descendants of the element through the platform accessibility API

The specification states:

```
5.2.8 Presentational Children
```

```
        RDF Property
            role:childrenArePresentational

        Values
            Boolean (true | false)

        The DOM descendants are presentational. User agents SHOULD NOT expose descendants of
        this element through the platform accessibility API. If user agents do not hide the
        descendant nodes, some information may be read twice.
```

EdgeHTML Mode

The user agent does not expose the descendants of the element through the platform accessibility API.

2.2.3 [W3C-WAI-ARIA-11] Section 5.4. Definition of Roles

C0008: The user agent treats elements with the role of main as navigational landmarks

The specification states:

```
    5.4 Definition of Roles
        ...
        main (role)

        The main content of a document.
        ...
        User agents SHOULD treat elements with the role of main as navigational landmarks.
```

EdgeHTML Mode

The user agent treats elements with the role of main as navigational landmarks.

C0011: The user agent treats element with the role of search as navigational landmarks

The specification states:

```
    5.4 Definition of Roles
        ...
        search (role)

        A landmark region that contains a collection of items and objects that, as a whole,
        combine to create a search facility. ...
        ...
        User agents SHOULD treat elements with the role of search as navigational landmarks.
```

EdgeHTML Mode

The user agent treats element with the role of search as navigational landmarks.

C0003: The user agent treats elements with the roll of banner as navigational landmarks

The specification states:

```
5.4 Definition of Roles
    ...
    banner (role)

    A region that contains mostly site-oriented content, rather than page-specific
    content.
    ...
    User agents SHOULD treat elements with the role of banner as navigational landmarks.
```

EdgeHTML Mode

The user agent treats elements with the roll of banner as navigational landmarks.

C0006: The user agent treats elements with the role of contentinfo as navigational landmarks

The specification states:

```
5.4 Definition of Roles
    ...
    contentinfo (role)

    A large perceivable region that contains information about the parent document.
    ...
    User agents SHOULD treat elements with the role of contentinfo as navigational
    landmarks.
```

EdgeHTML Mode

The user agent treats elements with the role of contentinfo as navigational landmarks.

C0005: The user agent treats elements with the role of complementary as navigation landmarks

The specification states:

```
5.4 Definition of Roles
    ...
    complementary (role)

    A supporting section of the document, designed to be complementary to the main
    content at a similar level in the DOM hierarchy, but remains meaningful when
    separated from the main content.
    ...
    User agents SHOULD treat elements with the role of complementary as navigational
    landmarks.
```

EdgeHTML Mode

The user agent treats elements with the role of complementary as navigation landmarks.

C0004: The user agent does not expose aria-required or aria-readonly in a columnheader to assistive technologies

The specification states:

```
5.4 Definition of Roles
    ...
    columnheader (role)

    A cell containing header information for a column.
    ...
    ... Therefore, authors SHOULD NOT use aria-required or aria-readonly in a
    columnheader that descends from a table, and user agents SHOULD NOT expose either
    property to assistive technologies unless the columnheader descends from a grid.
```

EdgeHTML Mode

The user agent does not expose `aria-required` or `aria-readonly` in a columnheader to assistive technologies.

C0007: The user agent treats elements with the role of form as navigational landmarks

The specification states:

```
5.4 Definition of Roles

    form (role)

    A landmark region that contains a collection of items and objects that, as a whole,
    combine to create a form. See related search.
    ...
    User agents SHOULD treat elements with the role of form as navigational landmarks.
```

EdgeHTML Mode

The user agent treats elements with the role of form as navigational landmarks.

C0010: The user agent does not expose aria-required or aria-readonly in a rowheader to assistive technologies

The specification states:

```
5.4 Definition of Roles
    ...
    rowheader (role)

    A cell containing header information for a row in a grid.
    ...
    ... Therefore, authors SHOULD NOT use aria-required or aria-readonly in a rowheader
    that descends from a table, and user agents SHOULD NOT expose either property to
    assistive technologies unless the rowheader descends from a grid.
```

EdgeHTML Mode

The user agent does not expose `aria-required` or `aria-readonly` in a rowheader to assistive technologies.

C0009: The user agent treats elements with the role of navigation as navigational landmarks

The specification states:

```
5.4 Definition of Roles
    ...
    navigation (role)

    A collection of navigational elements (usually links) for navigating the document or
    related documents.
    ...
    User agents SHOULD treat elements with the role of navigation as navigational
    landmarks.
```

EdgeHTML Mode

The user agent treats elements with the role of navigation as navigational landmarks.

2.2.4 [W3C-WAI-ARIA-11] Section 6.6. Definitions of States and Properties (all aria-* attributes)

C0014: The aria-colcount is not properly set

The specification states:

```
6.6 Definitions of States and Properties (all aria-* attributes)
    ...
    aria-colcount (property)

    Defines the total number of columns in a table, grid, or treegrid. ...
    ...
    Authors MUST set the value of aria-colcount to an integer equal to the number of
    columns in the full table. If the total number of columns is unknown, authors MUST
    set the value of aria-colcount to -1 to indicate that the value should not be
    calculated by the user agent.
```

EdgeHTML Mode

The `aria-colcount` is 0 and not set to -1.

C0012: The search for the first element with aria-atomic set is done, and all appropriate behavior is applied

The specification states:

```
6.6 Definitions of States and Properties (all aria-* attributes)
    ...
    aria-atomic (property)

    Indicates whether assistive technologies will present all, or only parts of, the
    changed region based on the change notifications defined by the aria-relevant
    attribute.
    ...
    When the content of a live region changes, user agents SHOULD examine the changed
    element and traverse the ancestors to find the first element with aria-atomic set,
    and apply the appropriate behavior for the cases below.

    1.  If none of the ancestors have explicitly set aria-atomic, the default is that
```

```
       aria-atomic is false, and assistive technologies will only present the
       changed node to the user.

    2.  If aria-atomic is explicitly set to false, assistive technologies will stop
        searching up the ancestor chain and present only the changed node to the user.

    3. If aria-atomic is explicitly set to true, assistive technologies will present the
       entire contents of the element, including the author-defined live region label if
       one exists.
```

EdgeHTML Mode

The changed element is examined and ancestors are traversed to find the first element with `aria-atomic` set, and all appropriate behavior is applied for the three cases.

2.2.5 [W3C-WAI-ARIA-11] Section 7.6. State and Property Attribute Processing

C0013: State and property attributes that are absent return a zero-length string

The specification states:

```
7.6 State and Property Attribute Processing
    ...
    Sometimes states and properties are present in the DOM but have a zero-length string
    ("") as their value. This is equivalent to their absence. User agents SHOULD treat
    state and property attributes with a value of "" the same as they treat an absent
    attribute. For supported states and properties, this corresponds to the default
    value, but if it is a required attribute, it signals an author error, and the
    implicit value for the role is used.
```

EdgeHTML Mode

State and property attributes that are absent return a zero-length string.

2.3 Extensions

There are no extensions to the requirements of [W3C-WAI-ARIA-11].

2.4 Error Handling

There are no additional error handling considerations.

2.5 Security

There are no additional security considerations.

3 Change Tracking

This section identifies changes that were made to this document since the last release. Changes are classified as Major, Minor, or None.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements.
- A document revision that captures changes to protocol functionality.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **None** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the relevant technical content is identical to the last released version.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	Description	Revision class
2.1 Normative Variations	Variation V0001 of the prior version is now clarification C0014.	Minor

4 Index