

[MS-POINTER]:

Microsoft Edge / Internet Explorer Pointer Events Standards Support Document

Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation (“this documentation”) for protocols, file formats, data portability, computer languages, and standards support. Additionally, overview documents cover inter-protocol relationships and interactions.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you can make copies of it in order to develop implementations of the technologies that are described in this documentation and can distribute portions of it in your implementations that use these technologies or in your documentation as necessary to properly document the implementation. You can also distribute in your implementation, with or without modification, any schemas, IDLs, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications documentation.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that might cover your implementations of the technologies described in the Open Specifications documentation. Neither this notice nor Microsoft's delivery of this documentation grants any licenses under those patents or any other Microsoft patents. However, a given Open Specifications document might be covered by the Microsoft [Open Specifications Promise](#) or the [Microsoft Community Promise](#). If you would prefer a written license, or if the technologies described in this documentation are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting iplg@microsoft.com.
- **License Programs.** To see all of the protocols in scope under a specific license program and the associated patents, visit the [Patent Map](#).
- **Trademarks.** The names of companies and products contained in this documentation might be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights. For a list of Microsoft trademarks, visit www.microsoft.com/trademarks.
- **Fictitious Names.** The example companies, organizations, products, domain names, email addresses, logos, people, places, and events that are depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

Reservation of Rights. All other rights are reserved, and this notice does not grant any rights other than as specifically described above, whether by implication, estoppel, or otherwise.

Tools. The Open Specifications documentation does not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments, you are free to take advantage of them. Certain Open Specifications documents are intended for use in conjunction with publicly available standards specifications and network programming art and, as such, assume that the reader either is familiar with the aforementioned material or has immediate access to it.

Support. For questions and support, please contact dochelp@microsoft.com.

Revision Summary

Date	Revision History	Revision Class	Comments
7/7/2015	1.0	New	Released new document.
11/2/2015	1.0	None	No changes to the meaning, language, or formatting of the technical content.
7/19/2016	1.1	Minor	Clarified the meaning of the technical content.
11/2/2016	1.1	None	No changes to the meaning, language, or formatting of the technical content.
3/14/2017	1.1	None	No changes to the meaning, language, or formatting of the technical content.
10/3/2017	1.1	None	No changes to the meaning, language, or formatting of the technical content.

Table of Contents

1	Introduction	4
1.1	Glossary	4
1.2	References	4
1.2.1	Normative References	4
1.2.2	Informative References	4
1.3	Microsoft Implementations	4
1.4	Standards Support Requirements	5
1.5	Notation.....	5
2	Standards Support Statements	7
2.1	Normative Variations	7
2.1.1	[W3C-POINTER] Section 5.2.8 The pointercancel event	7
2.1.2	[W3C-POINTER] Section 5.2.9 The pointerout event	7
2.1.3	[W3C-POINTER] Section 5.2.10 The pointerleave event	7
2.1.4	[W3C-POINTER] Section 5.2.12 The lostpointercapture event.....	8
2.1.5	[W3C-POINTER] Section 10.1 Setting Pointer Capture	8
2.2	Clarifications	8
2.2.1	[W3C-POINTER] Section 9.1 The touch-action CSS property	8
2.2.2	[W3C-POINTER] Section 11.1 Mapping for devices that support hover.....	9
2.2.3	[W3C-POINTER] Section 11.2 Mapping for devices that do not support hover.....	9
2.3	Extensions	10
2.3.1	Additional Properties	10
2.3.1.1	Attributes.....	10
2.3.1.1.1	-ms-touch-action.....	10
2.3.1.1.2	-ms-touch-select	11
2.4	Error Handling	12
2.5	Security	12
3	Change Tracking.....	13
4	Index.....	14

1 Introduction

This document describes the level of support provided by Microsoft web browsers for the W3C *Pointer Events* specification [\[W3C-POINTER\]](#), published 23 February 2015. The [W3C-POINTER] specification defines events and related interfaces for handling hardware agnostic pointer input from devices including a mouse, pen, and touchscreen.

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-POINTER] World Wide Web Consortium, "Pointer Events", W3C Recommendation 24 February 2015, <http://www.w3.org/TR/2015/REC-pointerevents-20150224/>

1.2.2 Informative References

[MS-CSS21E] Microsoft Corporation, "[Internet Explorer Extensions to Cascading Style Sheets \(CSS\) 2.1 and DOM Level 2 Style Specifications](#)".

1.3 Microsoft Implementations

The following Microsoft web browser versions implement some portion of the [\[W3C-POINTER\]](#) specification:

- Windows Internet Explorer 10
- Internet Explorer 11
- Internet Explorer 11 for Windows 10
- 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
Internet Explorer 10	Quirks Mode

Browser Version	Document Modes Supported
	IE7 Mode IE8 Mode IE9 Mode IE10 Mode
Internet Explorer 11	Quirks Mode IE7 Mode IE8 Mode IE9 Mode IE10 Mode IE11 Mode
Internet Explorer 11 for Windows 10	Quirks Mode IE7 Mode IE8 Mode IE9 Mode IE10 Mode IE11 Mode
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 [\[W3C-POINTER\]](#), 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 [\[W3C-POINTER\]](#) and whether they are considered normative or informative.

Sections	Normative/Informative
1	Informative
2	Normative
3,4	Informative
5-11	Normative

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 [\[W3C-POINTER\]](#).

- 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

The following subsections describe normative variations from the MUST requirements of [\[W3C-POINTER\]](#).

2.1.1 [W3C-POINTER] Section 5.2.8 The pointercancel event

V0001: The pointercancel event returns null

The specification states:

After firing the pointercancel event, a user agent must also fire a pointer event named pointerout followed by firing a pointer event named pointerleave.

IE11 Mode (All versions)

The pointercancel event incorrectly returns a null value when pointerout fires.

2.1.2 [W3C-POINTER] Section 5.2.9 The pointerout event

V0002: The pointercancel event returns null

The specification states:

After firing the pointercancel event (see pointercancel).

IE11 Mode (All versions)

The pointercancel event incorrectly returns a null value when pointerout fires.

2.1.3 [W3C-POINTER] Section 5.2.10 The pointerleave event

V0003: The pointercancel event returns null

The specification states:

A user agent must fire a pointer event named `pointerleave` when a pointing device is moved out of the hit test boundaries of an element and all of its descendants, including as a result of a `pointerup` and `pointercancel` events from a device that does not support hover (see `pointerup` and `pointercancel`).

IE11 Mode (All versions)

The `pointercancel` event incorrectly returns a null value when `pointerleave` fires.

2.1.4 [W3C-POINTER] Section 5.2.12 The `lostpointercapture` event

V0004: The `lostpointercapture` event is not fired

The specification states:

A user agent **MUST** fire a pointer event named `lostpointercapture` after pointer capture is released for a pointer. This event **MUST** be fired prior to any subsequent events for the pointer after capture was released. This event is fired at the element from which pointer capture was removed.

IE11 Mode (All versions)

The `lostpointercapture` event is not fired from the element that fired the event.

2.1.5 [W3C-POINTER] Section 10.1 Setting Pointer Capture

V0005: `InvalidStateError` not thrown

The specification states:

If the Element on which this method is invoked does not participate in its `ownerDocument`'s tree, throw an exception with the name `InvalidStateError`.

IE11 Mode (All versions)

An `InvalidStateError` is not thrown when the element is not part of the `ownerDocument` and the `setPointerCapture` method is invoked.

2.2 Clarifications

The following subsections describe clarifications of the MAY and SHOULD requirements of [\[W3C-POINTER\]](#).

2.2.1 [W3C-POINTER] Section 9.1 The touch-action CSS property

C0001: Touch events are considered

The specification states:

auto
The user agent MAY determine any permitted touch behaviors, such as panning and zooming manipulations of the viewport, for touches that begin on the element.

none
Touches that begin on the element MUST NOT trigger default touch behaviors.

pan-x
The user agent MAY consider touches that begin on the element only for the purposes of horizontally scrolling the element's nearest ancestor with horizontally scrollable content.

pan-y
The user agent MAY consider touches that begin on the element only for the purposes of vertically scrolling the element's nearest ancestor with vertically scrollable content.

manipulation
The user agent MAY consider touches that begin on the element only for the purposes of scrolling and continuous zooming. Any additional behaviors supported by auto are out of scope for this specification.

IE11 Mode and EdgeHTML Mode (All versions)

Touches that begin on elements are considered for the specific manipulation type.

2.2.2 [W3C-POINTER] Section 11.1 Mapping for devices that support hover

C0002: Mapping devices that support hover

The specification states:

Whenever a user agent is to dispatch a pointer event for a device that supports hover, it MAY run the following steps:

IE11 Mode and EdgeHTML Mode (All versions)

The steps are run to properly map pointer events to mouse events as appropriate.

2.2.3 [W3C-POINTER] Section 11.2 Mapping for devices that do not support hover

C0003: Mapping devices that do not support hover

The specification states:

This requires that user agents provide a different mapping for these types of input devices. Whenever a user agent is to dispatch a pointer event for a device that does not support hover, it MAY run the following steps:

IE11 Mode and EdgeHTML Mode (All versions)

The steps are run to properly map pointer events to mouse events as appropriate.

2.3 Extensions

The following subsections describe extensions to the requirements of [\[W3C-POINTER\]](#).

The extensions are as follows:

- [Additional Properties](#)

2.3.1 Additional Properties

This section lists CSS properties pertaining to touch events that are implemented by Microsoft web browsers in addition to those described in [\[W3C-POINTER\]](#).

2.3.1.1 Attributes

The CSSStyleDeclaration interface has been extended with the following attributes:

- [-ms-touch-action](#)
- [-ms-touch-select](#)

2.3.1.1.1 -ms-touch-action

IE10 Mode, IE11 Mode, and EdgeHTML Mode (All Versions)

msTouchAction of type `DOMString`, **read/write**

Sets or retrieves a value that specifies whether and how a given region can be manipulated by the user (for instance, by panning or zooming).

-ms-touch-action

Value:	auto none [[[pan-x pan-y pinch-zoom ?] manipulation] double-tap-zoom ?]
Initial:	auto
Applies to:	All elements
Inherited:	no
Percentages:	N/A
Media:	interactive
Computed value:	N/A

Values have the following meaning:

auto

Initial value. Indicates the browser will determine the permitted touch behaviors for the element.

none

The element permits no default touch behaviors.

pan-x

The element permits touch-driven panning on the horizontal axis. The touch pan is performed on the nearest ancestor with horizontally scrollable content.

pan-y

The element permits touch-driven panning on the vertical axis. The touch pan is performed on the nearest ancestor with vertically scrollable content.

pinch-zoom

The element permits pinch-zooming. The pinch-zoom is performed on the nearest ancestor with zoomable content. For more information about specifying content as zoomable, see the **-ms-content-zooming property** in [\[MS-CSS21E\]](#).

manipulation

The element permits touch-driven panning and pinch-zooming. This is the shorthand equivalent of "pan-x pan-y pinch-zoom".

double-tap-zoom

The element permits double-tap-zooming. The double-tap-zoom is performed on the full page. Double-tap-zoom is not available in Windows Store apps using JavaScript.

cross-slide-x

The element permits cross-sliding along the horizontal axis.

cross-slide-y

The element permits cross-sliding along the vertical axis.).

2.3.1.1.2 -ms-touch-select

IE10 Mode, IE11 Mode, and EdgeHTML Mode (All Versions)

msTouchSelect of type DOMString, **read/write**

Sets or retrieves a value that toggles the "gripper" visual elements that enable touch text selection.

-ms-touch-select

Value:	grippers none
Initial:	grippers
Applies to:	All elements
Inherited:	true
Percentages:	N/A
Media:	interactive
Computed value:	N/A

Values have the following meaning:

grippers

Initial value. Grippers are always on. In addition, selection will start even if an **onclick**, **onmsgesturetap**, or **onmouseup** handler is present.

none

Grippers are always off.

In addition to hiding the grippers, Windows Internet Explorer does not provide default touch selection functionality to the user.

2.4 Error Handling

There are no additional error handling considerations.

2.5 Security

There are no additional security considerations.

3 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

4 Index

A

Attributes

filter ([section 2.3.1.1.1](#) 10, [section 2.3.1.1.2](#) 11)
layout-grid-mode ([section 2.3.1.1.1](#) 10, [section 2.3.1.1.2](#) 11)

C

[Change tracking](#) 13

G

[Glossary](#) 4

I

[Informative references](#) 4

Interfaces

[CSS2Properties](#) 10
[Introduction](#) 4

N

[Normative references](#) 4

R

References

[informative](#) 4
[normative](#) 4

T

[Tracking changes](#) 13