Network Working Group
Internet-Draft
Intended status: Standards Track
Expires: December 30, 2010

P-A. LaFayette
Chromium Project

June 28, 2010

# The 'icon' URI scheme
# draft-lafayette-icon-uri-scheme-01

## Abstract

This document specifies the "icon" URI scheme. Icons have a fundamental role in user interaction with computer operating systems. In a graphical user interface, they may represent a file, folder, application, or device. They live on the desktop, in toolbars, and in menus. Most operating systems provide a standard set of icons that match the look and feel of the system. The "icon" URI scheme lets web page designers leverage these existing platform icons.

## Editorial Note (To be removed by RFC Editor)

Discussion of this draft should take place on the URI Review mailing list (uri-review@ietf.org).

## URI Scheme versus URN Namespace

It has been suggested that a URN namespace may be more appropriate for icon resolution than the new URI scheme. Discussion [1] on these matters has been ongoing in the uri-review@ietf.org mailing list. The arguments will be taken into consideration for the final decision as whether this draft should be considered for permanent registration.

## 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 December 30, 2010.

Copyright Notice



## 1. Introduction

An "icon" URI is designed to be a web accessible scheme used to resolve operating system icon resources. "Icon" URIs use either a file extension or an Internet media type to specify the filetype of a platform icon. Web pages can use "icon" URIs in any place that an image is specifiable by a URI. In particular, "icon" URIs can be used with "<img>" tags and JavaScript "Image" objects.

Similar icon URI schemes are currently being used in some modern browsers. The Mozilla project has a "moz-icon://" URI scheme that is web accessible and provides native and browser themed icons. The Chromium project has a "chrome://fileicon" scheme that is not web accessible, but does provide native icons for internal pages. This new scheme is an attempt to standardize icon URIs so that their core functionality may become available to all web pages.

## 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 [RFC2119].

## 3. URI Syntax

The general syntax of an "icon" URI is defined below using ABNF [RFC5234]:

```
iconuri = "icon:" ( fextension / mediatype / "unknown" /
                    "directory" / "parentdir" ) [ ";" size ]
fextension = "." 1*token ; File extension
mediatype = [ type ":" subtype ] ; Internet media type
size = pixels / "small" / "medium" / "large"
pixels = 1*digit ; Size of icon in square pixels
```

where "token", "digit", "type" and "subtype" rules are defined in [RFC2045]. The sections of an "icon" URI are ALWAYS case-insensitive.

## 4. Encoding Considerations

The encoding of the "type" and "subtype" rules is defined in [RFC2045]. The "fextension" rule may include characters from the Unicode Character Set [UCS], as suggested by URI [RFC3986], by first encoding those characters as octets to the UTF-8 character encoding [RFC3629]. Only those octets that do not correspond to characters in the unreserved set should be percent-encoded.

By using UTF-8 encoding, there are no known compatibility issues with mapping Internationalized Resource Identifiers to "icon" URIs according to [RFC3987]. Since "icon" URIs do not use domain names, "ireg-name" conversion is unnecessary.

## 5. Resolving "icon" URIs

An "icon" URI MUST resolve to an image resource representing an icon. The "directory" and "parentdir" keywords SHOULD resolve to the system icons for a directory and a parent directory (i.e. the "up" icon). The resolved icon SHOULD be the platform icon for a the specified

"fextension" or "mediatype". If an icon is not available for the specified filetype, applications MUST return a default icon which SHOULD be the system icon for files of unknown type. The icon resource's dimensions MUST match the size indicated in the URI or use the default size.

## 5.1. File extension

The filetype for an icon is specifiable by the "fextension" rule. An acceptable value for this section is a file extension for which a platform icon can be retrieved. Applications are REQUIRED to support "icon" URI resolution by file extension.





In the event that a platform icon is not available for the provided file extension, the application SHOULD return a default "unknown file" icon in the appropriate size.

## 5.2. Internet media type

The filetype for an icon is also specifiable by the "mediatype" rule. An acceptable value for this section is an Internet media type, as defined in [RFC2045] and [RFC2046], for which a platform icon can be retrieved. Applications are REQUIRED to support "icon" URI resolution by Internet media type.

In the event that a platform icon is not available for the provided media type, the application SHOULD return a default "unknown file" icon in the appropriate size.

## 5.3. Icon Size

An "icon" URI MAY specify the size of a requested icon through the "size" rule. The "size" MUST be either an integer value representing the width/height of the square icon in pixels or one of: "small", "medium", or "large". Applications are REQUIRED to support both user-provided integer values and the 3 keyword values.

If the "size" is specified as "small" the application MUST return a 16x16 pixels icon.

If the "size" is specified as "medium" the application MUST return a 64x64 pixels icon.

If the "size" is specified as "large" the application MUST return a 256x256 pixels icon.

If the "size" is not specified in the URI, the application MUST return a 16x16 pixels icon.

In the event that the platform does not support one of the keyword icon sizes or a user-provided size, the application SHOULD scale the icon appropriately.

## 5.4. Examples

icon: -- Displays a 16x16 pixels platform icon for an unknown filetype.

icon:;medium -- Displays a 64x64 pixels platform icon for an unknown filetype.





icon:.zip --Displays a 16x16 pixels platform icon for the .zip file extension.

icon:.MP3;32 --Displays a 32x32 pixels platform icon for the .mp3 file extension.

ICON:unknown --Displays a 16x16 pixels default icon for an unknown filetype.

icon:directory --Displays a 16x16 pixels platform icon for a directory.

icon:parentdir;32 --Displays a 32x32 pixels platform icon for a parent directory.

icon:image:jpeg;128 --Displays a 128x128 pixels platform icon for the image/jpeg media type.

icon:AUDIO:MPEG;large --Displays a 256x256 platform icon for the audio/mpeg media type.

icon:.doc;SMALL --Displays a 16x16 platform icon for the .doc file

extension.

icon:.pdf;medium --Displays a 64x64 platform icon for the .pdf file extension.

## 6. Normalization

"Icon" URIs adhere to the standard URI normalization rules [RFC3986]; specifically Simple String Comparison, Case Normalization, and Percent-Encoding Normalization. Due to the structure of "icon" URIs, the Syntax-Based, Scheme-Based, and Protocol-Based Normalization rules do not apply. The sections of an "icon" URI are ALWAYS case-insensitive.

## 7. Security Considerations

There are potential privacy risks that need to be taken into consideration with this new scheme.

"Icon" URIs specify private icon resources located on a user's machine. Web pages MUST NOT be able to retrieve the platform icons that this scheme provides to the user. Further, web pages MUST NOT be able to retrieve information about the platform icons that might expose the applications installed on a user's machine. The following sections explore specific issues that applications MUST address in their respective implementations.





### 7.1. Canvas Images

HTML5 introduces the canvas element which allows for dynamic scriptable rendering of 2D bitmap images.

Images can be imported onto the canvas by first retrieving a reference to a JavaScript "Image" object, then drawing the image on the canvas using the drawImage. Another option involves setting the 2D context's "fillStyle" or "strokeStyle" to a "CanvasPattern" object created from an "Image" object using the createPattern method. There are four ways one can retrieve a handle to an "Image" object:

1. Retrieve "<img>" elements already on the page using the DOM.

2. Retrieve other canvas elements using the document.getElementsById

method or the document.getElementsByTagName method.

3. Create an image from scratch, i.e. var img = new Image().

4. Embed an image using its data URI.

As a result of #1, an "<img>" tag, whose src attribute is an "icon" URI, can be retrieved as a JavaScript "Image" object. Considering #3; an "Image" can be created in JavaScript and have its src field is set to an "icon" URI.

Drawing the "Image" to the canvas provides web pages with two potential means of accessing the images: the getImageData and toDataURL methods of the canvas's 2D context. The canvas's 2D drawing API is defined in [Canvas2D].

An "Image" object with an "icon" URI as its src attribute MUST NOT be considered as same-origin as any other origin. If an "Image" with an "icon" URI src is drawn to a canvas, the canvas MUST be considered tainted and have its "origin-clean" flag set to false. As such, if "getImageData" or "toDataURL" is called on a canvas that has been tainted by "icon" URI "Image" data, the method MUST raise a SECURITY_ERR exception. See Security with canvas elements in the HTML5 Draft Standard [HTML5] for further details.

## 7.2. Image Size

Because the width and height of an "Image" object are always accessible, applications MUST use the default size (16x16) for platform icons if the size has not been provided by the "icon" URI. This is a requirement because evil web pages may be able to use the width and height of icons to infer details about the applications a user has installed.





# 8. IANA Considerations

This specification requests the IANA provisionally register the "icon" URI scheme as specified in this document and summarized in the following template, per [RFC4395]:

URI scheme name: icon

Status: Provisional

URI scheme syntax: See Section 3

URI scheme semantics: See Section 1

Encoding considerations: See Section 4

Intended usage: See Section 1 and Section 5.4

Applications and/or protocols that use this URI scheme name: Any applications that use URIs as identifiers for private resources, such as web browsers.

Interoperability considerations: See Section 6

Security considerations: See Section 7

Relevant publications: None

Contact: Pierre-Antoine LaFayette (pierre@alumni.utoronto.ca)

Author/Change controller: Pierre-Antoine LaFayette

## 9. Acknowledgements

This document was made possible thanks to the input of Adam Barth, Lachlan Hunt, Maciej Stachowiak, Jonas Sicking, Robert O'Callahan, Josh Soref, and Ian Fette.

## 10. References

### 10.1. Normative References

[RFC2045] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies", RFC 2045, November 1996.

[RFC2046] Freed, N. and N. Borenstein, "Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types", RFC 2046,





November 1996.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3629] Yergeau, F., "UTF-8, a transformation format of ISO 10646", STD 63, RFC 3629, November 2003.

[RFC3986] Berners-Lee, T., Fielding, R., and L. Masinter, "Uniform Resource Identifier (URI): Generic Syntax", STD 66, RFC 3986, January 2005.

[RFC3987] Duerst, M. and M. Suignard, "Internationalized Resource Identifiers (IRIs)", RFC 3987, January 2005.

[RFC5234] Crocker, D. and P. Overell, "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008.

[UCS] International Organization for Standardization, "Information Technology - Universal Multiple-Octet Coded Character Set (UCS)", ISO/IEC Standard 10646, December 2003.

## 10.2. Informative References

[Canvas2D] Hickson, I., Hyatt, D., Schepers, D., and E. Graff, "Canvas 2D API Specification 1.0".

[HTML5] Hickson, I., "HTML5 Draft Standard".

[RFC4395] Hansen, T., Hardie, T., and L. Masinter, "Guidelines and Registration Procedures for New URI Schemes", BCP 35, RFC 4395, February 2006.

URIs

[1] <http://www.ietf.org/mail-archive/web/uri-review/current/msg01161.html>

Author's Address

Pierre-Antoine Benoit LaFayette
Chromium Project

EMail: pierre@alumni.utoronto.ca