



ウェブコンテンツ開発ガイド

[ブラウザ機能拡張(500k 対応)編]

Version 1.0.2 / Aug.18,2010

ソフトバンク モバイル 株式会社

本書は情報提供を目的として作成されたものです。ソフトバンクモバイル株式会社は本書の記載内容に関して明示的にも、黙示的にも何ら保証するものではありません。

本書に記載されている事柄は、予告なしに変更する可能性があります。

本書の使用、または本書を使用した結果については、ユーザ各位がその責任を負うものとししますのでご了承ください。

1. ドキュメントの一部または全部を改版、引用することを禁じます。
2. ドキュメントを第三者に複製し、頒布することを禁じます。
3. ドキュメントを運用した結果の影響については、いっさいの責任を負いかねますのでご了承ください。

[商標]

- Flash、Flash Lite は、Adobe Systems, Inc.の米国およびその他の国における登録商標または商標です。
- JavaScript は、米国 Sun Microsystems, Inc.の米国およびその他の国における商標または登録商標です。
- SOFTBANK およびソフトバンクの名称、ロゴは日本国およびその他の国におけるソフトバンク株式会社の登録商標または商標です。

その他、記載されている会社名、製品名は、各社の商標または登録商標です。

■ 修正履歴

Version	日付	内容
1.0.0	2010/4/1	新規文書
1.0.1	2010/5/28	Appendix.I を追記
1.0.2	2010/8/18	7. スクリーンキャプチャ 記載更新 8.1. ストレージへの保存禁止 記載更新

0. イントロダクション	6
0.1. 目的	6
0.2. 前提	7
0.3. 表記	8
0.4. 参考文献	9
0.5. 本書の構成	12
0.6. 本書の位置付け	12
1. 概要	13
1.1. 機能概要	13
1.2. USER-AGENT	14
2. 最大ページサイズ拡張	15
3. FLASH VIDEO再生	16
4. スクリプト対応	17
4.1. (X)HTML拡張	18
4.1.1. javascriptスキーム	18
4.1.2. メタ・モジュール	18
4.1.3. スクリプト・モジュール: <i>Script Module</i>	19
4.1.4. イベントハンドラ	22
4.2. JAVASCRIPT	24
4.2.1. 字句規約(<i>Lexical Conventions</i>)	24
4.2.2. トークン(<i>Tokens</i>)	26
4.2.3. 識別子(<i>Identifiers</i>)	27
4.2.4. 区切り子(<i>Punctuators</i>)	27
4.2.5. リテラル(<i>Literals</i>)	28
4.2.6. 自動セミコロン挿入規則(<i>Rules of Automatic Semicolon Insertion</i>)	28
4.2.7. 型(<i>Types</i>)	28
4.2.8. 型変換(<i>Types Conversion</i>)	29
4.2.9. 式(<i>Expressions</i>)	29
4.2.10. 正規表現(<i>Regular Expression</i>)	35
4.2.11. 文(<i>Statements</i>)	37
4.2.12. <i>ECMAScript</i> オブジェクト(<i>Native ECMAScript Objects</i>)	39
4.3. DOM	40
4.3.1. <i>DOM Level 0</i>	40

4.3.2.	<i>DOM Core Level 1,2 DOMException</i>	40
4.3.3.	<i>DOM Core Level 1,2</i>	41
4.3.4.	<i>DOM Event Level 1,2</i>	41
4.3.5.	<i>DOM HTML Level 1,2</i>	41
4.3.6.	<i>DOM Style Level 2 Document Object Model CSS</i>	41
4.3.7.	<i>Parsing and serializing XML</i>	41
5.	マルチウィンドウ	42
5.1.	スクリプトによるマルチウィンドウ制御	43
6.	QVGA/VGAモード指定.....	44
7.	スクリーンキャプチャ	46
8.	コンテンツ保存抑止	47
8.1.	ストレージへの保存禁止.....	47
8.2.	テキストコピーの禁止	48
APPENDIX.A.	メディア別サイズ一覧	49
APPENDIX.B.	ECMASCRIPTオブジェクト.....	50
APPENDIX.C.	DOM LEVEL 0	55
APPENDIX.D.	DOM CORE LEVEL 1,2	57
APPENDIX.E.	DOM EVENT LEVEL 1,2	58
APPENDIX.F.	DOM HTML LEVEL 1,2	59
APPENDIX.G.	DOM STYLE LEVEL 2 DOCUMENT OBJECT MODEL CSS.....	62
APPENDIX.H.	PARSING AND SERIALIZING XML	64
APPENDIX.I.	ブラウザ機能拡張端末向けコンテンツにおけるセキュリティ要件.....	65

0. イントロダクション

0.1. 目的

本書はブラウザ機能を拡張した端末向けにコンテンツ提供を行うコンテンツパートナー(以降、*CP*)様に、ご理解いただきたい内容を説明するものである。

本書は、機能の追加・変更に伴い予告なく随時改定される。

0.2. 前提

本書は以下の技術について熟知していることを前提とする。

- ❖ *HTTP/1.1*: Hyper Text Trasfer Protocol 1.1
- ❖ *HTML*: HyperText Markup Language
- ❖ *XML*: eXtensible Markup Language
- ❖ *XHTML*: The eXtensible HyperText Markup Language
- ❖ *CSS2*: Cascading Style Sheets, level 2
- ❖ *WCSS*: Wireless CSS
- ❖ *DOM2CORE*: Document Object Model(DOM) level2 Core Specification
- ❖ *DOM2EVENTS*: Document Object Model(DOM) level2 Events Specification
- ❖ *DOM2HTML*: Document Object Model(DOM) level2 HTML Specification
- ❖ *DOM2STYLE*: Document Object Model(DOM) level2 Style Specification
- ❖ *ECMASCRIPT*: ECMAScript Language Specification, 3rd Edition, ECMA-262
- ❖ *ESMP*: ECMAScript Mobile Profile

加えて、弊社提供のドキュメント

- ❖ ウェブコンテンツ開発ガイド[HTTP 編]
- ❖ ウェブコンテンツ開発ガイド[HTML 編]
- ❖ ウェブコンテンツ開発ガイド[XHTML 編]
- ❖ ウェブコンテンツ開発ガイド[メディア編]

については既読であること。

0.3. 表記

本書では以下の表記法を用いる。

表 0.3-1 本書で用いる表記法

表記	意味
Courier New	HTTP,HTML の構文要素
<i>Italic</i>	初出の用語、もしくは、強調したい用語
Gothic	強調したい用語

0.4. 参考文献

[HTTP]

"Hypertext Transfer Protocol – HTTP/1.1", IETF, RFC2616, June 1999

[HTML]

"HTML 4.01 Specification", W3C Recommendation, D.Raggett, A.Le Hors, I.Jacobs, eds., 24 December 1999

[XML]

"Extensible Markup Language (XML) 1.0 (Second Edition)", W3C Recommendation, T.Bray, J.Paoli, C.M.Sperberg-McQueen, E.Maler,eds., 6 October 2000

[XHTML Basic]

"XHTML Basic", W3C Recommendation, M.Baker et al., 19 December 2000

[XHTML1]

"XHTML 1.0 : The Extensible HyperText Markup Language – A Reformulation of HTML 4 in XML 1.0", W3C Recommendation, Steven Pemberton et al., 26 January 2000

[XHTML11]

"XHTML 1.1 Module-based XHTML", W3C Recommendation, Murray Altheim et al., 31 May 2001

[Modularization of XHTML]

"Modularization of XHTML", W3C Recommendation, M.Altheim et al., 10 April 2001

[XHTMLMP12]

"XHTML Mobile Profile 1.2", Open Mobile Alliance, OMA-TS-XHTMLMP-V1_2-20080331-A, 31 March 2008

[CSS2]

"Cascading Style Sheets, level 2", W3C Recommendation, H.W.Lie, B.Bos, C.Lilley, I.Jacobs, 12 May 1998

[WCSS11]

"Wireless CSS Specification 1.1", Open Mobile Alliance, OMA-WAP-WCSS-V1_1-20061020-A, 20 Oct 2006

[DOM1]

"Document Object Model (DOM) Level 1 Specification Version 1.0", W3C Recommendation, 1 October 1998

[DOM2CORE]

"Document Object Model (DOM) Level 2 Core Specification Version 1.0", W3C Recommendation, 13 November 2000

[DOM2EVENTS]

"Document Object Model (DOM) Level 2 Events Specification Version 1.0", W3C Recommendation, 13 November 2000

[DOM2HTML]

"Document Object Model (DOM) Level 2 HTML Specification Version 1.0", W3C Recommendation, 9 January 2003

[DOM2STYLE]

"Document Object Model (DOM) Level 2 Style Specification Version 1.0", W3C Recommendation, 13 November 2000

[JavaScript13]

"JavaScript 1.3 Client-Side Guide", Netscape Communications Corporation., 1999
"JavaScript 1.3 Client-Side Reference", Netscape Communications Corporation., 1999

[ECMAScript]

"ECMAScript Language Specification, 3rd Edition", Standard ECMA-262, December 1999

[ESMP]

"ECMAScript Mobile Profile A Wireless Markup Scripting Language Approved Version 1.0", Open Mobile Alliance, 20 October 2006

[DLOTA]

"Generic Content Download Over The Air Specification Version 1.0", Open Mobile Alliance, OMA-Download-OTA-v1_0

[DRM]

"Digital Rights Management", Open Mobile Alliance, OMA-Download-DRM-v1_0

0.5. 本書の構成

本書は以下の構成である。

- 1章 概要：本機能の概要を説明する。
- 2章 最大ページサイズ拡張：ページサイズの拡張について説明する。
- 3章 Flash Video再生：Flash Video再生について説明する。
- 4章 スクリプト対応：利用可能なJavaScriptについて説明する。
- 5章 マルチウィンドウ：タブブラウザ機能について説明する。
- 6章 QVGA/VGAモード指定：コンテンツ表示モード切替機能について説明する。
- 7章 スクリーンキャプチャ：スクリーンキャプチャ機能について説明する。
- 8章 コンテンツ保存抑止：コンテンツ保存を禁止する方法について説明する。

0.6. 本書の位置付け

本書では、本機能向けに新規追加された仕様、および、変更された仕様について説明する。また、変更がない仕様についても一部、説明する。本書で説明の無い事項は弊社提供の各種開発ガイドを参照すること。

1. 概要

1.1. 機能概要

ブラウザ機能拡張(500k 対応)端末は、従来の端末と比較して、以下の機能を拡張している。

- ・ 最大ページサイズ拡張
- ・ Flash Video プログレッシブ再生
- ・ Ajax を含むスクリプト言語対応
- ・ マルチウィンドウ(タブブラウザ機能)対応
- ・ VGA/QVGA モード指定機能
- ・ スクリーンキャプチャ機能
- ・ コンテンツ保存抑止機能

1. 2. User-Agent

ブラウザ機能拡張(500k 対応)に対応する端末は、User-Agent リクエストヘッダフィールドの UE-Generation-Version を"2.0"とする。

User-Agent 例:

User-Agent: SoftBank/2.0/111XX/XXJ001 Browser/XX-Browser/1.0

User-Agent の詳細は「ウェブコンテンツ開発ガイド[HTTP 編]」を参照すること。

2. 最大ページサイズ拡張

- リクエストは 500kbytes 未満
- エンティティサイズは 500kbytes 未満(ただし、テキストおよびマークアップ言語を表す Content-Type の場合は 48kbytes 未満)
- ページサイズ(ルートドキュメント+インラインデータ)は 500kbytes 未満

個別の最大サイズについては、Appendix.Aを参照すること。

3. **Flash Video** 再生

端末は、500kbytes の Flash Video 再生に対応している。詳細はウェブコンテンツ開発ガイド[メディア編]を参照すること。

4. スクリプト対応

端末は、ソフトバンク携帯電話向け JavaScript を利用できる。

ソフトバンク携帯電話向け JavaScript は、Standard ECMA-262 ECMAScript Language Specification 3rd edition[ECMAScript]に準拠した JavaScript に DOM(DOM Level0[JavaScript13], W3C DOM Level1[DOM1], W3C DOM Level2[DOM2CORE][DOM2EVENT][DOM2HTML][DOM2STYLE])+XMLHttpRequest で定義されたインターフェイスを追加したものである。

4. 1. (X)HTML 拡張

4.1.1. javascript スキーム

端末は、スクリプトを実行する為の javascript スキームに対応する。
javascript スキームは、a 要素の href 属性と form 要素の action 属性にて使用できる。

(X)HTML 記述例:

```
<a href="javascript:history.back()">戻る</a>
```

4.1.2. メタ・モジュール

ドキュメント中にスクリプト言語を使用する場合、meta 要素でスクリプト言語を明示すること。

XHTML 記述例:

```
<meta http-equiv="content-script-type" content="text/javascript" />
```

4.1.3. スクリプト・モジュール: Script Module

スクリプト・モジュールは、実行可能なスクリプトに属する情報、実行可能なスクリプトのサポートの欠如に属する情報を含むときに用いる要素を定義する。

4.1.3.1. Script 要素

定義

```

<!ENTITY % script.element "INCLUDE" >
<![%script.element;[
<!ENTITY % script.content "( #PCDATA )" >
<!ENTITY % script.qname "script" >
<!ELEMENT %script.qname; %script.content; >
<!-- end of script.element -->]]>

<!ENTITY % script.attlist "INCLUDE" >
<![%script.attlist;[
<!ATTLIST %script.qname;
    type           %ContentType.datatype; #REQUIRED
    src            %URI.datatype;       #IMPLIED
    defer         ( defer )             #IMPLIED
>
<!-- end of script.attlist -->]]>

```

要素	属性	最小内容モデル
script	type, src, defer	PCDATA

属性名	必須	case	デフォルト値
type	○	CS	スクリプト言語の MIME 型を指定する。
src	×	CT	スクリプトを外部リソースより取得する際の URI を指定する。
defer	×	CS	スクリプトの実行を待たずにドキュメントの表示を行う。

説明

文章中にスクリプトを組み込むための要素である。スクリプトはこの要素の内容として記述することも、外部リソースから読み込むことも可能である。

属性

type ::= "type" "=" %ContentType

スクリプトの MIME 型を指定する。値は"text/javascript"を指定すること。

src ::= "src" "=" %URI

外部スクリプトの URI を指定する。URI はスキームから始まる絶対 URI で記述すること。

defer ::= "defer" "=" "defer"

スクリプトの実行を待たずにドキュメントの表示を行う。ドキュメント表示の変更を伴わないスクリプトである場合、この属性を指定することにより、スクリプトの解析・実行を待つことなくドキュメントを表示することができる。

4.1.3.2. Noscript 要素

定義

```
-----  
<!ENTITY % noscript.element "INCLUDE" >  
<![%noscript.element;[  
<!ENTITY % noscript.content  
      "( %Block.mix; )+"  
>  
<!ENTITY % noscript.qname "noscript" >  
<!ELEMENT %noscript.qname; %noscript.content; >  
<!-- end of noscript.element -->]]>  
  
<!ENTITY % noscript.attlist "INCLUDE" >  
<![%noscript.attlist;[  
<!ATTLIST %noscript.qname;  
      %Common.attrib;  
>  
<!-- end of noscript.attlist -->]]>
```

要素	属性	最小内容モデル
noscript	Common	(Block)+

説明

スクリプトが実行できない場合に、代わりに表示させる内容を示す。

制約

body 要素の内容モデルとして記載しなければならない。

4.1.4. イベントハンドラ

要素と利用可能なイベントハンドラの対応表を以下に示す。

表 4.1.4-1 イベントハンドラ

ハンドラ 要素	on blur	on change	on click	on focus	key events [*1]	on load	on unload	mouse events [*2]	on reset	on submit
body			○[*3]			○	○	○		
address			○[*3]					○		
blockquote			○[*3]					○		
cite			○[*3]					○		
dfn			○[*3]					○		
div			○[*3]					○		
em			○[*3]					○		
h1 - h6			○[*3]					○		
kbd			○[*3]					○		
p			○[*3]					○		
pre			○[*3]					○		
q			○[*3]					○		
span			○[*3]					○		
strong			○[*3]					○		
var			○[*3]					○		
a	○		○	○	○			○		
dl			○[*3]					○		
dt			○[*3]					○		
dd			○[*3]					○		
ol			○[*3]					○		
ul			○[*3]					○		
li			○[*3]					○		
b			○[*3]					○		
big			○[*3]					○		
hr			○[*3]					○		
i			○[*3]					○		
small			○[*3]					○		
form			○[*3]					○	○	○
input	○	○	○	○	○			○		

ハンドラ 要素	on blur	on change	on click	on focus	key events [*1]	on load	on unload	mouse events [*2]	on reset	on submit
label			○[*3]					○		
select	○	○	○	○				○		
option			○		○			○		
textarea	○	○	○	○	○			○		
fieldset			○[*3]					○		
optgroup			○[*3]					○		
table			○[*3]					○		
caption			○[*3]					○		
tr			○[*3]					○		
td			○[*3]					○		
th			○[*3]					○		
img			○[*3]			○		○		
object			○[*3]					○		
link			○[*3]					○		
Window object						○	○			

[*1]: keyevents は、onkeydown, onkeypress を示す

[*2]: mouseevents は、onmousedown, onmousemove, onmouseout, onmouseover, onmouseup を示す

[*3]: マウスポインタおよびタッチパネルのカーソル操作のみ

4. 2. JavaScript

ソフトバンク携帯電話向け JavaScript は、Standard ECMA-262 ECMAScript Language Specification 3rd edition[ECMAScript]に準拠する。

4.2.1. 字句規約(Lexical Conventions)

ECMAScript 完全互換。詳細は[ECMAScript]7章を参照すること。

4.2.1.1. Unicode 制御文字(Unicode Format-Control Characters)

ECMAScript 完全互換。詳細は[ECMAScript]7.1章を参照すること。

4.2.1.2. 空白(White Space)

ECMAScript 完全互換。以下を空白文字とする。

表 4.2.1.2-1 空白文字

Code Point Value	Name	Formal Name
¥u0009	Tab	<TAB>
¥u000B	Vertical Tab	<VT>
¥u000C	Form Feed	<FF>
¥u0020	Space	<SP>
¥u00A0	No-break space	<NBSP>
¥u1680	Ogham space mark	<USP>
¥u180E	Mongolian vowel separator	
¥u2000	En quad	
¥u2001	Em quad	
¥u2002	En space	
¥u2003	Em space	
¥u2004	Three-per-em space	
¥u2005	Four-per-em space	
¥u2006	Six-per-em space	
¥u2007	Figure space	
¥u2008	Punctuation space	
¥u2009	Thin space	
¥u200A	Hair space	
¥u202F	Narrow no-break space	
¥u205F	Medium mathematical space	
¥u3000	Ideographic space	

4.2.1.3. 行終端文字(Line Terminators)

ECMAScript 完全互換。以下を行終端文字とする。

表 4.2.1.3-1 行終端文字

Code Point Value	Name	Formal Name
¥u000A	Line Feed	<LF>
¥u000D	Carriage Return	<CR>
¥u2028	Line separator	<LS>
¥u2029	Paragraph separator	<PS>

4.2.1.4. コメント(Comments)

ECMAScript 完全互換。

JavaScript 記述例:

```
// 一行コメント
/* 範囲コメント */
```

4.2.2. トークン(Tokens)

4.2.2.1. 予約語(Reserved Words)

ECMAScript 完全互換。予約語を識別子として使用することはできない。

4.2.2.2. キーワード(Keywords)

ECMAScript 完全互換。以下のキーワードを識別子として使用することはできない。

break, case, catch, continue, default, delete, do, else, finally, for, function, if, in, instanceof, new, return, switch, this, throw, try, typeof, var, void, while, with

4.2.2.3. 将来の予約語(Future Reserved Words)

ECMAScript 完全互換。以下のワードを予約する。

abstract, boolean, byte, char, class, const, debugger, double, enum, export, extends, final, float, goto, implements, import, int, interface, long, native, package, private, protected, public, short, static, super, synchronized, throws, transient, volatile

4.2.3. 識別子(Identifiers)

ECMAScript 完全互換。詳細は[ECMAScript]7.6 章を参照すること。

4.2.4. 区切り子(Punctuators)

ECMAScript 完全互換。詳細は[ECMAScript]7.7 章を参照すること。

4.2.5. リテラル(Literals)

ECMAScript 完全互換。リテラルには以下の種類がある。
リテラルの詳細は[ECMAScript]7.8章を参照すること。

- ・空リテラル(Null Literals)
- ・真理値リテラル(Boolean Literals)
- ・数値リテラル(Numeric Literals)
- ・文字列リテラル(String Literals)
- ・正規表現リテラル(Regular Expression Literals)
- ・配列リテラル(Array Literals)
- ・オブジェクトリテラル(Object Literals)

4.2.6. 自動セミコロン挿入規則(Rules of Automatic Semicolon Insertion)

ECMAScript 完全互換。詳細は[ECMAScript]7.9章を参照すること。

4.2.7. 型(Types)

ECMAScript 完全互換。型には以下の種類がある。
型の詳細は[ECMAScript]8章を参照すること。

- ・未定義型(The Undefined Type)
- ・空型(The Null Type)
- ・真理値型(The Boolean Type)
- ・文字列型(The String Type)
- ・数値型(The Number Type)
- ・オブジェクト型(The Object Type)

4.2.8. 型変換(Types Conversion)

ECMAScript 完全互換。詳細は[ECMAScript]9章を参照すること。

4.2.9. 式(Expressions)

4.2.9.1. 演算子(Operators)

ソフトバンク携帯電話向け JavaScript は、以下の演算子を使用できる。
演算子の詳細は[ECMAScript]11章を参照すること。

表 4.2.9.1-1 算術演算子

Name	Syntax
後置インクリメント	x++
前置インクリメント	++x
単項プラス	+x
後置デクリメント	x--
前置デクリメント	--x
単項マイナス	-x
加算	x + y
減算	x - y
乗算	x * y
除算	x / y
剰余	x % y

表 4.2.9.1-2 比較演算子

Name	Syntax
等価	<code>x == y</code>
非等価	<code>x != y</code>
厳密等価	<code>x === y</code>
厳密非等価	<code>x !== y</code>
大なり	<code>x > y</code>
大なりイコール	<code>x >= y</code>
小なり	<code>x < y</code>
小なりイコール	<code>x <= y</code>

表 4.2.9.1-3 ビット演算子

Name	Syntax
ビット積	<code>x & y</code>
ビット和	<code>x y</code>
ビット排他的論理和	<code>x ^ y</code>
ビット否定	<code>~x</code>
左シフト	<code>x << y</code>
符号付き右シフト	<code>x >> y</code>
符号無し右シフト	<code>x >>> y</code>

表 4.2.9.1-4 論理演算子

Name	Syntax
論理積	<code>expr1 && expr2</code>
論理和	<code>expr1 expr2</code>
論理否定	<code>!expr</code>

表 4.2.9.1-5 代入演算子

Name	Syntax
単純代入	<code>x = y</code>
加算代入	<code>x += y</code>
減算代入	<code>x -= y</code>
乗算代入	<code>x *= y</code>
除算代入	<code>x /= y</code>
剰余代入	<code>x %= y</code>
左シフト代入	<code>x <<= y</code>
符号付き右シフト代入	<code>x >>= y</code>
符号無し右シフト代入	<code>x >>>= y</code>
ビット積代入	<code>x &= y</code>
ビット排他的論理和代入	<code>x ^= y</code>
ビット和代入	<code>x = y</code>

表 4.2.9.1-6 文字列演算子

Name	Syntax
文字列結合	<code>string + string</code>
文字列結合代入	<code>string += string</code>

表 4.2.9.1-7 特殊演算子

Name	Syntax
条件演算子	condition ? expr1 : expr2
カンマ演算子	expr1, expr2
delete 演算子[*1]	delete objectName delete objectName.property delete objectName[index] delete property
new 演算子	objectName = new objectType (param1 [, param2] ...[,paramN])
typeof 演算子	typeof operand typeof (operand)
void 演算子	void (expression) void expression
instanceof 演算子	objectName instanceof objectType
in 演算子	propNameOrNumber in objectName
this 演算子	this[.propertyName]
配列添え字	[]
関数呼び出し	()
メンバアクセス	.

[*1]: 「delete property」は、with 文内でのみ有効

4.2.9.2. 優先順位と結合規則(Precedence and Associativity)

ソフトバンク携帯電話向け JavaScript で利用可能な演算子の評価優先順位と結合規則を以下に示す。

表 4.2.9.2-1 演算子の優先順位と結合規則

Precedence	Operator	Description	Associativity
1	.	メンバアクセス	左から右
	[]	配列添え字	
	new	new 演算子	右から左
2	()	関数呼び出し	左から右
3	++	インクリメント	両方
	--	デクリメント	
4	!	論理否定	右から左
	~	ビット否定	
	+	単項プラス	
	-	単項マイナス	
	typeof	typeof 演算子	
	void	void 演算子	
	delete	delete 演算子	
5	*	乗算	左から右
	/	除算	
	%	剰余	
6	+	加算	左から右
	-	減算	
7	<<	左シフト	左から右
	>>	符号付き右シフト	
	>>>	符号無し右シフト	
8	<	小なり	左から右
	<=	小なりイコール	
	>	大なり	
	>=	大なりイコール	
	in	in 演算子	
	instanceof	instanceof 演算子	
9	==	等価	左から右

	!=	非等価	
	===	厳密等価	
	!==	厳密非等価	
10	&	ビット積	左から右
11	^	ビット排他的論理和	左から右
12		ビット和	左から右
13	&&	論理積	左から右
14		論理和	左から右
15	?:	条件演算子	右から左
16	=	単純代入	右から左
	+=	加算代入	
	-=	減算代入	
	*=	乗算代入	
	/=	除算代入	
	%=	剰余代入	
	<<=	左シフト代入	
	>>=	符号付き右シフト代入	
	>>>=	符号無し右シフト代入	
	&=	ビット積代入	
	^=	ビット排他的論理和代入	
	=	ビット和代入	
17	,	カンマ演算子	左から右

4.2.10. 正規表現(Regular Expression)

ソフトバンク携帯電話向け JavaScript で利用可能な正規表現の構文を以下に示す。
構文の詳細は[ECMAScript]15.10 章を参照すること。

- Disjunction = Alternative | Disjunction
- Alternative = Empty
- Alternative = Term | Alternative
- Term = Atom [?*] [?]
- = ^
- = \$
- = ¥b
- = ¥B
- Atom = (Disjunction)
- = (? Disjunction)
- = (?= Disjunction)
- = (! Disjunction)
- = PatternCharacter
- = .
- = ¥ 0
- = DecimalEscape
- = ¥ [fnrtv]
- = ¥ c [a-zA-Z]
- = ¥ x hex hex
- = ¥ u hex hex hex hex
- = ¥ [dDsSwW]
- = IdentityEscape
- = ¥ [^] [] (ClassAtom · ClassAtom)* ¥]
- ClassAtom
- = ¥b
- = ¥ c [a-zA-Z]
- = ¥ x hex hex
- = ¥ u hex hex hex hex
- = ¥ [dDsSwW]
- = SourceCharacter · "¥] ·"
- PatternCharacter = SourceCharacter but not any of ^ \$ ¥ . * + ? () [] { } |

- `ControlEscape` = [fnrtv]
- `ControlLetter` = [a-zA-Z]
- `CharacterClassEscape` = [dDsSwW]

4.2.11. 文(Statements)

ソフトバンク携帯電話向け JavaScript で利用可能な「文」の構文を以下に示す。
構文の詳細は[ECMAScript]12 章を参照すること。

表 4.2.11-1 文

Statement	Syntax	Remark
Variable Statement	<code>var varname [= value] [..., varname [= value]]</code>	
Empty Statement	<code>;</code>	
The if Statement	<code>if (condition) { statements1 } [else { statements2 }]</code>	
Iteration Statements	<code>do statements while (condition);</code> <code>while (condition) { statements }</code> <code>for ([initial-expression]; [condition]; [increment-expression]) { statements }</code> <code>for (variable in object) { statements }</code>	最大ループ 回数は 1000000 回
The continue Statement	<code>continue [label]</code>	
The break Statement	<code>break [label]</code>	
The return Statement	<code>return expression</code>	
The with Statement	<code>with (object) { statements }</code>	
The label Statement	<code>label : statements</code>	
The switch Statement	<code>switch (expression) { case label : statements; break; case label : statements;</code>	

	<pre>break; ... default : statements; }</pre>	
The throw Statement	throw expression	
The try Statement	<pre>try { statements } [catch (catchID) { statements }] [finally { statements }]</pre>	
The function Statement	<pre>function name([param][,param] [..., param]) { statements }</pre>	
Comments	// comment text	
	/* multiple line comment text */	

4.2.12. ECMAScript オブジェクト(Native ECMAScript Objects)

ソフトバンク携帯電話向け JavaScript で利用可能な ECMAScript オブジェクトを Appendix.B に示す。詳細は[ECMAScript]15 章を参照すること。

4.3. DOM

ソフトバンク携帯電話向け JavaScript で利用可能な DOM を以下に示す。

4.3.1. DOM Level 0

ソフトバンク携帯電話向け JavaScript で利用可能な DOM Level 0 仕様を Appendix.C に示す。詳細は[JavaScript13]を参照すること。

4.3.2. DOM Core Level 1,2 DOMException

ソフトバンク携帯電話向け JavaScript で利用可能な DOMException コードを以下に示す。詳細は[DOM2CORE]1.2 章を参照すること。

表 4.3.2-1 DOM Core Level 1,2 DOMException

ExceptionCode
INDEX_SIZE_ERR
HIERARCHY_REQUEST_ERR
WRONG_DOCUMENT_ERR
NO_MODIFICATION_ALLOWED_ERR
NOT_FOUND_ERR
NOT_SUPPORTED_ERR
INUSE_ATTRIBUTE_ERR
INVALID_STATE_ERR
SYNTAX_ERR
INVALID_MODIFICATION_ERR
NAMESPACE_ERR
INVALID_ACCESS_ERR

4.3.3. DOM Core Level 1,2

ソフトバンク携帯電話向け JavaScript で利用可能な DOM Level 1,2 Core 仕様を Appendix.D に示す。詳細は[DOM1][DOM2CORE]を参照すること。

4.3.4. DOM Event Level 1,2

ソフトバンク携帯電話向け JavaScript で利用可能な DOM Level 1,2 Event 仕様を Appendix.E に示す。詳細は[DOM1][DOM2EVENT]を参照すること。

4.3.5. DOM HTML Level 1,2

ソフトバンク携帯電話向け JavaScript で利用可能な DOM Level 1,2 HTML 仕様を Appendix.F に示す。詳細は[DOM1][DOM2HTML]を参照すること。

4.3.6. DOM Style Level 2 Document Object Model CSS

ソフトバンク携帯電話向け JavaScript で利用可能な DOM Level 2 CSS 仕様を Appendix.G に示す。詳細は[DOM1][DOM2STYLE]を参照すること。

4.3.7. Parsing and serializing XML

ソフトバンク携帯電話向け JavaScript で利用可能な XMLHttpRequest 仕様を Appendix.H に示す。

5. マルチウィンドウ

端末は複数のウィンドウを制御するタブブラウザ機能を有する。本機能はユーザの任意でコンテンツを別ウィンドウに表示するものであるが、スクリプトを利用した場合、コンテンツ側で制御を行うことも可能である。

5.1. スクリプトによるマルチウィンドウ制御

本章では、JavaScript の `window.open()` の第 2 引数および Flash® ActionScript の `getURL()` の第 2 引数を `target` 値と表記する。

`target` 値が `"_blank"` の場合は、新しいウィンドウに表示する。

`target` 値が `"_top"`、`"_self"`、`"_parent"` のいずれかである場合は、スクリプトを実行したウィンドウに表示する。

上記 4 つの予約名称以外の任意 `target` 値である場合は、新しいウィンドウに表示する。

但し、任意 `target` 値でも既に開いているウィンドウ名称と重複する場合は、該当する既存ウィンドウにて表示する。

また、ウィンドウ数が端末で表示可能な上限に達している等、新規ウィンドウで表示できない場合は、既存ウィンドウにて表示する。

上記を表にしたものが、表 5.1-1 となる。

表 5.1-1 マルチウィンドウ制御

target 値 新規可/不可	<code>_blank</code>	<code>_top</code> <code>_self</code> <code>_parent</code>	重複なし	重複あり
可	A	B	A	C
不可	B	B	B	C

A: 新規ウィンドウで表示する

B: スクリプトを実行したウィンドウで表示する

C: 名称が重複する既存ウィンドウで表示する

なお、マルチウィンドウに対応しない端末の場合、`target` 値は無視して全てのウィンドウに表示する。

6. QVGA/VGA モード指定

QVGA/VGA モードの指定は、ルートドキュメント中に `meta` 要素で記述する。子フレーム等に記載しても、有効としない。

`name` 属性値に `"disparea"`、`content` 属性値に `"qvga"` を指定した場合、端末は QVGA モードが指定されたものと解釈する。

`name` 属性値に `"disparea"`、`content` 属性値に `"vga"` を指定した場合、端末は VGA モードが指定されたものと解釈する。

XHTML 記述例:

```
<!-- 下記のいずれか一方を記述 -->
<meta name="disparea" content="qvga" />
<meta name="disparea" content="vga" />
```

ブラウザでコンテンツを表示する際に VGA 以上の解像度を持つ端末で、QVGA モードが指定されていた場合、コンテンツ中の対象となるテキストやイメージを縦横 2 倍角で表示する。同端末で VGA モードが指定されるか、何も指定が無い場合、コンテンツのリサイズを行わずに表示する

ブラウザでコンテンツを表示する際に VGA 未満の解像度を持つ端末(表示ベースが QVGA となる端末)で、VGA モードが指定されていた場合、コンテンツ中の対象となるテキストやイメージを縦横 1/2 倍角で表示する。同端末で QVGA モードが指定されるか、何も指定が無い場合、コンテンツのリサイズを行わずに表示する。

なお、以下についてはリサイズの対象外とする。

- CSS データタイプ: `<percentage>`、`<length>` の `em` と `ex`
- CSS プロパティ: `font-size` の `<absolute-size>` と `<relative-size>`
- (X)HTML 要素: `h1-h6`、`big`、`small`

XHTML 記述例:

```
<h3>テスト</h3><br />
<font style="font-size:medium;">あいうえおかきくけこ</font><br />
<font style="font-size:24px;">あいうえおかきくけこ</font><br />
<br />
<br />
```

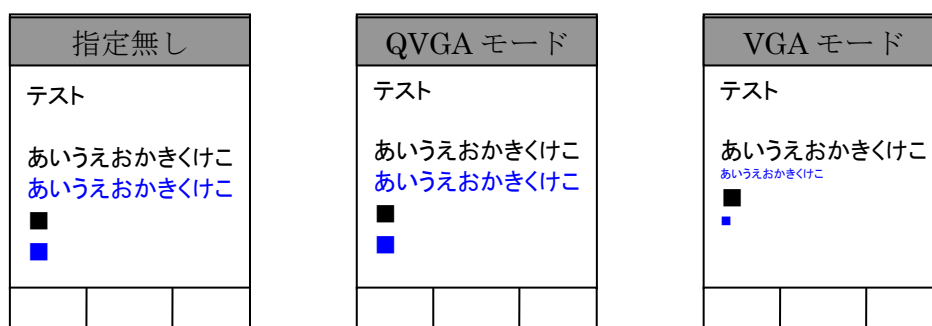


図 6-1 QVGA 端末描画モード表示例

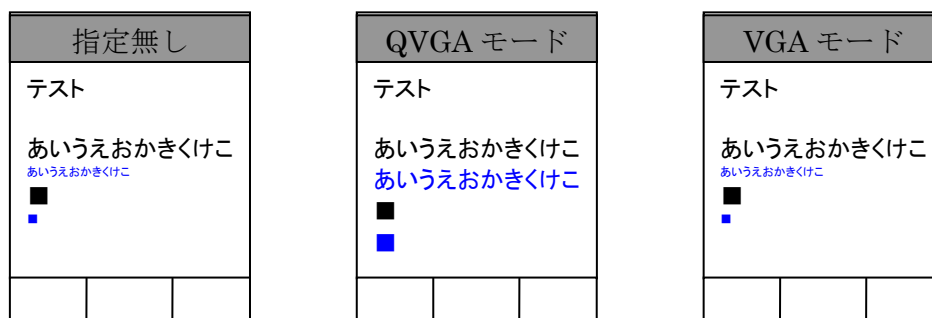


図 6-2 VGA 端末描画モード表示例

7. スクリーンキャプチャ

端末は、ブラウザ表示画面のスクリーンキャプチャ機能を有する。スクリーンキャプチャで保存された画像に対しては『ブラウザで表示』のみできる。なお、本機能はコンテンツ側の指定で無効とすることが可能である。詳細は「8.1 ストレージへの保存禁止」を参照すること。

8. コンテンツ保存抑止

8.1. ストレージへの保存禁止

ストレージへの保存禁止の指定は、**meta** 要素で記述する。子フレームに記載した場合、同一ウィンドウ内の全てのフレームで保存を禁止する。

name 属性値に"**prohibition**"、**content** 属性値に"**image**"を指定した場合、端末はストレージへの保存禁止が指定されたものと解釈する。

XHTML 記述例:

```
<meta name="prohibition" content="image" />
```

保存禁止の対象となる操作および機能を以下に記述する。

- ・スクリーンキャプチャ操作
- ・インライン画像/インライン **Flash**®コンテンツの保存

スクリーンキャプチャ操作は本機能でのみ制御が可能となる。なお、インライン画像/インライン **Flash**®コンテンツの保存については、従来の著作物保護制御も有効である。詳細は「ウェブコンテンツ開発ガイド[HTTP 編]」を参照すること。

また、本機能は後述する「テキストコピーの禁止」と両立可能である。その際は、属性値をカンマ区切りで記述するか、両方の**meta**要素を別々に記述すること。

8.2. テキストコピーの禁止

テキストコピー禁止の指定は、**meta** 要素で記述する。子フレームに記載した場合、同一ウィンドウ内の全てのフレームでテキストコピーを禁止する。

name 属性値に"**prohibition**"、**content** 属性値に"**text**"を指定した場合、端末はテキストコピーの禁止が指定されたものと解釈する。

XHTML 記述例:

```
<meta name="prohibition" content="text" />
```

なお、本機能は前述の「ストレージへの保存禁止」と両立可能である。その際は、属性値をカンマ区切りで記述するか、両方の**meta**要素を別々に記述すること。

XHTML 記述例(カンマ区切り):

```
<!-- ストレージ保存/テキストコピーの両方を禁止する -->  
<meta name="prohibition" content="image,text" />
```

XHTML 記述例(別々):

```
<!-- ストレージ保存/テキストコピーの両方を禁止する -->  
<meta name="prohibition" content="image" />  
<meta name="prohibition" content="text" />
```


Appendix.A. メディア別サイズ一覧

表 A-1 メディア別サイズ一覧

カテゴリ	フォーマット	メディア型(MIME)	サイズ
ページ記述	HTML	text/html	48k
	XHTML	text/html application/xhtml+xml application/vnd.wap.xhtml+xml	48k
	CSS	text/css	48k
	JavaScript	text/javascript	48k
Java™	JAD	text/vnd.sun.j2me.app-descriptor	6k
	JAR	application/java application/java-archive	1M
モバイル ウィジェット	WGT	application/widget	500k
	SWGTT	application/x-s-widget	500k
メディア	PNG	image/png	500k
	JPEG	image/jpeg	500k
	GIF	image/gif	500k
	WBMP	image/vnd.wap.wbmp	-
	SMAF	application/x-smaf	500k
	SMF	audio/midi	500k
	SP-MIDI	audio/midi	500k
	XMF	audio/xmf0 audio/xmf1	-
	MP4	video/3gpp	500k
	SVG	image/svg+xml	-
	Flash	application/x-shockwave-flash	500k
	Flash Video	video/x-flv	500k
デコレメール	テンプレート	application/x-htmlmail-template	300k
	マイ絵文字パッケージ	application/x-s-pictogram-multipart	300k
DRM	Forward Lock	application/vnd.oma.drm.message	500k
OMA Download	Download Descriptor	application/vnd.oma.dd+xml	48k
その他	text	text/plain	48k
	vCard	text/x-vcard	48k
	vBookmark	text/x-vbookmark	48k
	vCalender	text/x-vcalender	48k
	vMessage	text/x-vmessage	48k
	vNote	text/x-vnote	48k

Appendix.B. ECMAScriptオブジェクト

Object	Constructor/Property/Method	Remark	
Global	(Value Property)		
	NaN		
	Infinity		
	undefined		
	(Functions Property)		
	eval(x)		
	parseInt(string [, radix])		
	parseFloat(string)		
	isNaN(number)		
	isFinite(number)		
	(URI Handling Function Property)		
	decodeURI(encodedURI)		
	decodeURIComponent(encodedURIComponent)		
	encodeURI(uri)		
	encodeURIComponent(uriComponent)		
	escape(string)		
	unescape(string)		
	Objects	(as a Function)	
		Object([value])	
(Constructor)			
new Object([value])			
(Property of the Object Constructor)			
length			
prototype			
(Property of the Object Prototype Object)			
constructor			
toString()			
valueOf()			
Function	(as a Function)		
	Function([p1, p2, ... , pN,] body)		
	(Constructor)		
	new Function([p1, p2, ... , pN,] body)		
	(Property of the Function Constructor)		
	length		
	prototype		
	(Property of Function Instances)		
	length		
	prototype		
	(Property of the Function Prototype Object)		
	constructor		
	toString()		
	apply(thisArg [, argArray])		
call(thisArg [, arg1, arg2, ... , argN])			
Arguments	(Property)		
	callee		
	length		
Array	(as a Function)		
	Array([item1, item2, ... , itemN])		
	(Constructor)		
	new Array([item0, item1, ... , itemN])		
	new Array(len)		

	(Property of the Array Constructor)		
	length		
	prototype		
	(Property of the Array Prototype Object)		
	constructor		
	toString()		
	concat([item1, item2, ... , itemN])		
	join(separator)		
	pop()		
	push([item1, item2, ... , itemN])		
	reverse()		
	shift()		
	slice(start [, end])		
	sort(comparefn)		
	splice(start, deleteCount [, item1, item2, ... , itemN])		
	unshift([item1, item2, ... , itemN])		
	(Property of Array Instance)		
	length		
String	(as a Function)		
	String([value])		
	(Constructor)		
	new String([value])		
	(Property of the String Constructor)		
	length		
	prototype		
	fromCharCode([code0, code1, ... , codeN])		
	(Property of the String Prototype Object)		
	constructor		
	toString()		
	valueOf()		
	charAt(pos)		
	charCodeAt(pos)		
	concat([item1, item2, ... , itemN])		
	indexOf(pattern [, pos])		
	lastIndexOf(pattern [, pos])		
	localeCompare(that)		
	match(regexp)		
	replace(searchValue, replaceValue)		
	search(regexp)		
	slice(start [, end])		
	split(separator [, lim])		
	substring(start, end)		
	substr(start [, length])		
	toLowerCase()		
	toLocaleLowerCase()		
	toUpperCase()		
	toLocaleUpperCase()		
	(Property of String Instance)		
	length		
	Boolean	(as a Function)	
		Boolean(value)	
		(Constructor)	
		new Boolean(value)	
		(Property of the Boolean Constructor)	

	length	
	prototype	
	(Property of the Boolean Prototype Object)	
	constructor	
	toString()	
	valueOf()	
Number	(as a Function)	
	Number([value])	
	(Constructor)	
	new Number([value])	
	(Property of the Number Constructor)	
	length	
	prototype	
	MAX_VALUE	
	MIN_VALUE	
	NaN	
	NEGATIVE_INFINITY	
	POSITIVE_INFINITY	
	(Property of the Number Prototype Object)	
	constructor	
	toString([radix])	
	toLocaleString()	
	valueOf()	
	toFixed(fractionDigits)	
	toExponential(fractionDigits)	
	toPrecision(precision)	
Math	(Value Property)	
	E	
	LN10	
	LN2	
	LOG2E	
	LOG10E	
	PI	
	SQRT1_2	
	SQRT2	
	(Function Property)	
	abs(x)	
	acos(x)	
	asin(x)	
	atan(x)	
	atan2(y, x)	
	ceil(x)	
	cos(x)	
	exp(x)	
	floor(x)	
	log(x)	
	max([value1, value2, ... , valueN])	
	min([value1, value2, ... , valueN])	
	pow(x, y)	
	random()	
	round(x)	
	sin(x)	
	sqrt(x)	
	tan(x)	

Date	(as a Function)	
	Date ([year, month, date, hours, minutes, seconds, ms])	
	(Constructor)	
	new Date(year, month [, date, hours, minutes, seconds, ms])	
	new Date()	
	new Date(milliseconds)	
	new Date(dateString)	
	(Property of the Date Constructor)	
	length	
	prototype	
	parse(string)	
	UTC(year, month [, date, hours, minutes, seconds, ms])	
	(Property of the Date Prototype Object)	
	constructor	
	toString()	
	toDateString()	
	toTimeString()	
	toLocaleString()	
	toLocaleDateString()	
	toLocaleTimeString()	
	valueOf()	
	getTime()	
	getFullYear()	
	getUTCFullYear()	
	getMonth()	
	getUTCMonth()	
	getDate()	
	getUTCDate()	
	getDay()	
	getUTCDay()	
	getHours()	
	getUTCHours()	
	getMinutes()	
	getUTCMinutes()	
	getSeconds()	
	getUTCSeconds()	
	getMilliseconds()	
	getUTCMilliseconds()	
	getTimezoneOffset()	
	setTime(time)	
	setMilliseconds(ms)	
	setUTCMilliseconds(ms)	
	setSeconds(sec [, ms])	
	setUTCSeconds(sec [, ms])	
	setMinutes(minute [, sec, ms])	
	setUTCMinutes(minute [, sec, ms])	
	setHours(hour [, minute, sec, ms])	
	setUTCHours(hour [, minute, sec, ms])	
	setDate(date)	
	setUTCDate(date)	
setMonth(month [, date])		
setUTCMonth(month [, date])		
setFullYear(year [, month, date])		
setUTCFullYear(year [, month, date])		

	toUTCString()	
	toGMTString()	
	getFullYear()	
	setYear(year)	
RegExp	(as a Function)	
	RegExp(pattern [, flags])	
	(Constructor)	
	new RegExp(pattern [, flags])	
	(Property of the RegExp Constructor)	
	length	
	prototype	
	(Property of the RegExp Prototype Object)	
	constructor	
	exec([string])	
	test([string])	
	toString()	
	valueOf()	
	(Property of RegExp Instance)	
	source	
	global	
	ignoreCase	
multiline		
lastIndex		
Error	(as a Function)	
	Error(message)	
	(Constructor)	
	new Error(message)	
	(Property of the Error Constructor)	
	length	
	prototype	
	(Property of the Error Prototype Object)	
	constructor	
	name	
	message	
	toString()	
code	実装依存	
EvalError	The same as "Error Object"	実装依存
RangeError	The same as "Error Object"	実装依存
ReferenceError	The same as "Error Object"	実装依存
SyntaxError	The same as "Error Object"	実装依存
TypeError	The same as "Error Object"	実装依存
URIError	The same as "Error Object"	実装依存

Appendix.C. DOM Level 0

Object	Attribute/Method/Event Handler/Constructor	Type	Read Only	Remark	
Navigator	(Attribute)				
	appName	DOMString	Yes	“Mozilla”	
	appVersion	DOMString	Yes	“5”以上	
	cookieEnabled	boolean	Yes	戻り値は端末設定に従う。 ON: true、OFF: false、都度選択: true	
	language	DOMString	Yes	“ja-JP”	
	mimeType	MimeTypeArray	Yes		
	platform	DOMString	Yes	“SoftBank”	
	plugins	PluginArray	Yes		
	userAgent	DOMString	Yes		
	MimeType	(Attribute)			
description		DOMString	Yes		
enabledPlugin		Plugin	Yes		
suffixes		DOMString	Yes		
type		DOMString	Yes		
Plugin	(Attribute)				
	description	DOMString	Yes		
	filename	DOMString	Yes	実装依存	
	length	unsigned long	Yes		
	name	DOMString	Yes		
Screen	(Attribute)				
	availHeight	long	Yes		
	availWidth	long	Yes		
	colorDepth	long	Yes		
	height	long	Yes		
	width	long	Yes		
Window	(Attribute)				
	closed	boolean	Yes	値はfalseのみサポート	
	document	HTMLDocument	Yes		
	frames	Window	Yes		
	history	History	Yes		
	location	Location	No		
	name	DOMString	No		
	navigator	Navigator	No		
	opener	Window	No		
	pageXOffset	long	Yes		
	pageYOffset	long	Yes		
	parent	Window	Yes		
	self	Window	Yes		
	top	Window	Yes		
	window	Window	Yes		
	XMLHttpRequest	XMLHttpRequest	No		
	(Method)				
	alert(in DOMString message)	void			最大文字数: 半角128
	clearInterval(in long intervalID)	void			
	clearTimeout(in long timeoutID)	void			
	close()	void			
	confirm(in DOMString message)	boolean			
	focus()	void			最大文字数: 半角100
	home()	void			
	open(in DOMString URL, in DOMString windowName, in DOMString windowFeatures)	Window			
	prompt(in DOMString message, in DOMString inputDefault / in DOMString message, in long inputDefault)	DOMString			最大文字数: メッセージ半角80、入力512
	scrollBy(in long horizontal, in long vertical)	void			
	scrollTo (in long x-coordinate, in long y-coordinate)	void			
	setInterval(in DOMString expression, in long msec, in Function function, in long msec, in arg1, argN)	long			
	setTimeout(in DOMString expression, in long msec, in Function function, in long msec, in arg1, argN)	long			
	(Event Handlers)				
	onload				
	onunload				
	History	(Attribute)			
		length	unsigned long	Yes	
		(Method)			
		back()	void		
		forward()	void		
	go(in long delta / in DOMString location)	void			
	Location	(Attribute)			
		hash	DOMString	No	
		hostname	DOMString	No	
		href	DOMString	No	
		pathname	DOMString	No	

protocol	DOMString	No	
search	DOMString	No	
(Method)			
assign(sURL)	void		
reload([in boolean forceGet])	void		
replace(in DOMString URL)	void		

Appendix.D. DOM Core Level 1,2

Interface	Attribute / Method / Event Handler / Constructor	Exception	Type	Read Only	Remark
DOMException	(Attribute)				
	code		unsigned short	No	実装依存
DOMImplementation	(Method)				
	createDocument(in DOMString namespaceURI, in DOMString qualifiedName, in DocumentType doctype)	raises(DOMException)	Document		
DocumentFragment					
Document	(Attribute)				
	documentElement		Element	Yes	
	implementation		DOMImplementation	Yes	
	(Method)				
	createAttribute(in DOMString name)	raises(DOMException)	Attr		
	createComment(in DOMString data)		Comment		
	createDocumentFragment()		DocumentFragment		
	createElement(in DOMString tagName)	raises(DOMException)	Element		
	createTextNode(in DOMString data)		Text		
	getElementById(in DOMString elementId)		Element		
	getElementsByName(in DOMString tagName)		HTMLCollection		
	(Attribute)				
	attributes		NamedNodeMap	Yes	
	childNodes		NodeList	Yes	
	firstChild		Node	Yes	
lastChild		Node	Yes		
nextSibling		Node	Yes		
nodeName		DOMString	Yes		
nodeValue	raises(DOMException) on setting raises(DOMException) on retrieval	DOMString	No		
nodeType		unsigned short	Yes		
ownerDocument		Document	Yes		
parentNode		Node	Yes		
prefix	raises(DOMException) on setting	DOMString	No	書き込みのみサポート	
previousSibling		Node	Yes		
(Method)					
appendChild(in Node newChild)	raises(DOMException)	Node			
cloneNode(in boolean deep)		Node		実装依存	
hasAttributes()		boolean		実装依存	
hasChildNodes()		boolean			
insertBefore(in Node newChild, in Node refChild)	raises(DOMException)	Node			
replaceChild(in Node newChild, in Node oldChild)	raises(DOMException)	Node			
removeChild(in Node oldChild)	raises(DOMException)	Node			
NodeList	(Attribute)				
	length		unsigned long	Yes	
	(Method)				
item(in unsigned long index)		Node			
NamedNodeMap	(Attribute)				
	length		unsigned long	Yes	
	(Method)				
	getNamedItem(in DOMString name)		Node		
	item(in unsigned long index)		Node		
	removeNamedItem(in DOMString name)	raises(DOMException)	Node		
setNamedItem(in Node arg)	raises(DOMException)	Node			
CharacterData	(Attribute)				
	data	raises(DOMException) on setting raises(DOMException) on retrieval	DOMString	No	
	length		unsigned long	Yes	
	(Method)				
	appendData(in DOMString arg)	raises(DOMException)	void		
	deleteData(in unsigned long offset, in unsigned long count)	raises(DOMException)	void		
	insertData(in unsigned long offset, in DOMString arg)	raises(DOMException)	void		
	replaceData(in unsigned long offset, in unsigned long count, in DOMString arg)	raises(DOMException)	void		
substringData(in unsigned long offset, in unsigned long count)	raises(DOMException)	DOMString			
Attr(attribute)	(Attribute)				
	name		DOMString	Yes	
	specified		boolean	Yes	
Element	value		DOMString	No	
	(Attribute)				
Element	tagName		DOMString	Yes	
	(Method)				
	getAttribute(in DOMString name)		DOMString		
	getElementsByName(in DOMString name)		HTMLCollection		
	hasAttribute(name)		boolean		実装依存
	removeAttribute(in DOMString name)	raises(DOMException)	void		実装依存
	removeAttributeNode(in Attr oldAttr)	raises(DOMException)	Attr		
	setAttribute(in DOMString name, in DOMString value)	raises(DOMException)	void		
	setAttributeNode(in Attr newAttr)	raises(DOMException)	Attr		
	(Method)				
	splitText(in unsigned long offset)	raises(DOMException)	Text		
Comment					

Appendix.E. DOM Event Level 1,2

Interface	Attribute/Method/Event Handler/Constructor	Type	Read Only	Remark
EventTarget	(Method)			
	addEventListener()	void		
	removeEventListener()	void		
Event	(Attribute)			
	target		Yes	
	timeStamp	DOMTimeStamp	Yes	
	type	String	Yes	
	keyCode	integer	Yes	値を以下に示す。 キー名称: キーコード(Decimal表記) 0: 48 1: 49 2: 50 3: 51 4: 52 5: 53 6: 54 7: 55 8: 56 9: 57 決定: 13 ↑: 38 ↓: 40 ←: 37 →: 39 *: 42 #: 35

Appendix.F. DOM HTML Level 1,2

Interface	Attribute/Method/Event Handler/Constructor	Type	Read Only	Remark
HTMLCollection	(Attribute)			
	length	unsigned long	Yes	
	(Method)			
	item(in unsigned long index)	Node		
HTMLDocument	namedItem(in DOMString name)	Node		
	(Attribute)			
	anchors	HTMLCollection	Yes	
	applets	HTMLCollection	Yes	
	body	HTMLElement	No	
	cookie	DOMString	No	
	domain	DOMString	Yes	
	forms	HTMLCollection	Yes	
	images	HTMLCollection	Yes	
	lastModified		Yes	
	links	HTMLCollection	Yes	
	referrer	DOMString	Yes	
	title	string	No	
	URL	DOMString	Yes	
	(Method)			
	close()	void		
	getElementsByName(in DOMString elementName)	NodeList		
open()	void			
write(in DOMString text)	void			
writeln(in DOMString text)	void			
HTMLElement	(Attribute)			
	className	DOMString	No	
	id	DOMString	No	
	innerHTML	String	No	
	innerText	String	No	
	offsetParent		Yes	
	lang	DOMString	No	
	style	CSS2Properties	Yes	
	title	DOMString	No	
	HTMLHtmlElement	(Attribute)		
version		DOMString	No	
HTMLHeadElement	(Attribute)			
	profile	DOMString	No	
HTMLLinkElement	(Attribute)			
	href	DOMString	No	
	rel	DOMString	No	
HTMLTitleElement	(Attribute)			
	text	DOMString	No	
HTMLMetaElement	(Attribute)			
	content	DOMString	Yes	
	httpEquiv	DOMString	Yes	
	name	DOMString	Yes	
HTMLBaseElement	(Attribute)			
	href	DOMString	No	
HTMLStyleElement	(Attribute)			
	type	DOMString	No	
HTMLFormElement	(Attribute)			
	action	DOMString	No	
	elements	HTMLCollection	Yes	
	enctype	DOMString	No	
	length	long	Yes	
	method	DOMString	No	
	(Method)			
	reset()	void		
	submit()	void		
HTMLSelectElement	(Attribute)			
	form	HTMLFormElement	Yes	
	length	Integer	Yes	
	multiple	boolean	No	
	name	DOMString	No	
	options	HTMLCollection	Yes	
	selectedIndex	long	No	
	size	long	No	
	type	DOMString	Yes	
	value	DOMString	No	
	(Method)			
	add(in HTMLElement element, in HTMLElement before)	void		
	blur()	void		
focus()	void			
remove(in long index)	void			
HTMLOptionElement	(Attribute)			
	defaultSelected	boolean	Yes	
	form	HTMLFormElement	Yes	
	selected	boolean	No	
	text	DOMString	Yes	
	value	DOMString	No	

HTMLInputElement	(Attribute)			
	accessKey	DOMString	No	
	checked	boolean	No	
	defaultChecked	boolean	Yes	
	defaultValue	String	Yes	
	form	HTMLFormElement	Yes	
	maxLength	long	No	
	name	String	No	
	size	unsigned long	No	
	type	DOMString	Remark参照	Document.createElement() の直後のみ書き換えが可能
	value	DOMString	No	
	(Method)			
	blur()	void		
	click()	void		
focus()	void			
HTMLTextAreaElement	(Attribute)			
	accessKey	DOMString	No	
	cols	long	No	
	defaultValue	DOMString	No	
	form	HTMLFormElement	Yes	
	name	DOMString	No	
	rows	long	No	
	type	DOMString	Yes	
	value	DOMString	No	
	(Method)			
blur()	void			
focus()	void			
select()	void		実装依存	
HTMLUListElement	(Attribute)			
type	DOMString	No		
HTMLOListElement	(Attribute)			
	start	long	No	
type	DOMString	No		
HTMLLIElement	(Attribute)			
	type	DOMString	No	
value	long	No		
HTMLDivElement	(Attribute)			
align	DOMString	No		
HTMLParagraphElement	(Attribute)			
	align	DOMString	No	
HTMLHeadingElement	(Attribute)			
	align	DOMString	No	
HTMLQuoteElement	(Attribute)			
	cite	DOMString	No	
HTMLBRElement	(Attribute)			
	clear	DOMString	No	
HTMLFontElement	(Attribute)			
	color	DOMString	No	
HTMLHRElement	(Attribute)			
	align	DOMString	No	
	noShade	boolean	No	
	size	DOMString	No	
	width	DOMString	No	
HTMLAnchorElement	(Attribute)			
	accessKey	DOMString	No	
	hash	DOMString	No	
	hostname	DOMString	No	
	href	DOMString	No	
	name	DOMString	No	
	pathname	DOMString	No	
	protocol	DOMString	No	
	search	DOMString	No	
	target	DOMString	No	
	(Method)			
blur()	void			
focus()	void			
HTMLImageElement	(Attribute)			
	align	DOMString	No	
	alt	DOMString	No	
	complete	boolean	Yes	
	height	long	No	
	hspace	long	No	
	src	DOMString	No	
	vspace	long	No	
width	long	No		
HTMLObjectElement	(Attribute)			
	data	DOMString	No	
	declare	boolean	No	
	height	DOMString	No	
	type	DOMString	No	
width	DOMString	No		
HTMLScriptElement	(Attribute)			
	src	DOMString	No	

HTMLTableElement	(Attribute)			
	bgColor	DOMString	No	
	border	DOMString	No	
	rows	HTMLCollection	Yes	
HTMLTableCaptionElement	width	DOMString	No	
	(Attribute)			
HTMLTableRowElement	align	DOMString	No	
	(Attribute)			
HTMLTableCellElement	align	DOMString	No	
	cells	HTMLCollection	Yes	
	vAlign	DOMString	No	
	(Attribute)			
HTMLTableCellElement	align	DOMString	No	
	bgColor	DOMString	No	
	colSpan	long	No	
	rowSpan	long	No	
	vAlign	DOMString	No	
	(Attribute)			
HTMLFrameSetElement	cols	DOMString	No	
	rows	DOMString	No	
HTMLFrameElement	(Attribute)			
	src	DOMString	No	
HTMLIFrameElement	(Attribute)			
	height	DOMString	No	
	name	DOMString	No	
	src	DOMString	No	
HTMLIFrameElement	width	DOMString	No	
	(Attribute)			

Appendix.G. DOM Style Level 2 Document Object Model C

Interface	Attribute/Method/Event Handler/Constructor	Type	Read Only	Remark
CSS2Properties	(Attribute)			
	background	DOMString	No	
	backgroundAttachment	DOMString	No	
	backgroundColor	DOMString	No	
	backgroundImage	DOMString	No	
	backgroundPosition	DOMString	No	
	backgroundRepeat	DOMString	No	
	border	DOMString	No	
	borderCollapse	DOMString	No	
	borderColor	DOMString	No	
	borderStyle	DOMString	No	
	borderTop	DOMString	No	
	borderRight	DOMString	No	
	borderBottom	DOMString	No	
	borderLeft	DOMString	No	
	borderTopColor	DOMString	No	
	borderRightColor	DOMString	No	
	borderBottomColor	DOMString	No	
	borderLeftColor	DOMString	No	
	borderTopStyle	DOMString	No	
	borderRightStyle	DOMString	No	
	borderBottomStyle	DOMString	No	
	borderLeftStyle	DOMString	No	
	borderTopWidth	DOMString	No	
	borderRightWidth	DOMString	No	
	borderBottomWidth	DOMString	No	
	borderLeftWidth	DOMString	No	
	borderWidth	DOMString	No	
	clear	DOMString	No	
	color	DOMString	No	
	display	DOMString	No	
	cssFloat	DOMString	No	
	font	DOMString	No	
	fontFamily	DOMString	No	
	fontSize	DOMString	No	
	fontStyle	DOMString	No	
	fontWeight	DOMString	No	
	height	DOMString	No	
	left	DOMString	No	
	lineHeight	DOMString	No	
	listStyle	DOMString	No	
	listStyleImage	DOMString	No	
	listStylePosition	DOMString	No	
	listStyleType	DOMString	No	
margin	DOMString	No		
marginTop	DOMString	No		
marginRight	DOMString	No		
marginBottom	DOMString	No		
marginLeft	DOMString	No		
overflow	DOMString	No		
padding	DOMString	No		
paddingTop	DOMString	No		
paddingRight	DOMString	No		

paddingBottom	DOMString	No	
paddingLeft	DOMString	No	
position	DOMString	No	
textAlign	DOMString	No	
textDecoration	DOMString	No	
textIndent	DOMString	No	
top	DOMString	No	
verticalAlign	DOMString	No	
visibility	DOMString	No	
whiteSpace	DOMString	No	
width	DOMString	No	
zIndex	DOMString	No	

Appendix.H. Parsing and serializing XML

Object	Attribute/Method/Event Handler/Constructor	Exception	Type	Read Only	Remark
XMLHttpRequest	(Attribute)				
	onreadystatechange		EventListener	No	
	readyState		unsigned short	Yes	0: 初期値(Uninitialized) 1: HTTP-Request 送信中(Open) 2: HTTP-Request 送信完了(Sent) 3: HTTP-Response 受信中(Receiving) 4: HTTP-Response 受信完了(Loaded) 1~3は実装依存
	responseText		DOMString	No	
	responseXML		Document	No	
	status	DOMException	unsigned short	No	
	statusText	DOMException	DOMString	No	
	(Method)				
	open(in DOMString method, in DOMString uri)		void		
	open(in DOMString method, in DOMString uri, in boolean async)		void		
	open(in DOMString method, in DOMString uri, in boolean async, in DOMString user)		void		
	open(in DOMString method, in DOMString uri, in boolean async, in DOMString user, in DOMString password)		void		
	setRequestHeader(in DOMString header, in DOMString value)	raises(DOMException)	void		書き換え可能なヘッダは"x-xml*"のみ
	send(in DOMString data)	raises(DOMException)	void		
	abort()		void		
	getAllResponseHeaders()		DOMString		
	getResponseHeader(in DOMString header)		DOMString		

Appendix.I. ブラウザ機能拡張端末向けコンテンツ におけるセキュリティ要件

ブラウザ機能拡張端末では、Ajax を利用した非同期通信を行うことが可能となる為、端末ユーザの意図によらず、通常の端末操作とは異なるタイミングで HTTP リクエストを行う場合がある。

このとき、端末では same origin policy によりスクリプト記述元のドメインのみリクエストを許容する。但し、悪意ある者が DNS Rebinding 攻撃を行った場合、通常はアクセスできないコンテンツに対し、リクエストが行われてしまう。

DNS Rebinding に対しては、Host リクエストヘッダを確認することで、正当なリクエストであるか、DNS Rebinding 攻撃であるかを見分けることが可能である。

ブラウザ機能拡張端末向けにコンテンツを提供する場合、そのコンテンツが JavaScript を利用しないコンテンツであっても、ユーザの意思によるリクエストであることを確認する必要がある際は、セキュリティ要件として Host リクエストヘッダの内容がコンテンツ提供を行っているドメインであるか確認すること。