

PSP™ (PlayStation® Portable)
インターネットブラウザ向け
コンテンツ作成ガイドライン

Version 3.10

© 2007 Sony Computer Entertainment Inc.
All Rights Reserved.

[商標]

"UMD"および"PlayStation"は株式会社ソニー・コンピュータエンタテインメントの登録商標です。

"PSP"は株式会社ソニー・コンピュータエンタテインメントの商標です。

"Memory Stick"はソニー株式会社の商標です。

"XMB"はソニー株式会社および株式会社ソニー・コンピュータエンタテインメントの商標です。

本ドキュメントは、一部に株式会社 ACCESS の著作物を含んでいます。

"PSP"のインターネット機能は株式会社 ACCESS の **NetFront** を搭載しています。

Copyright© 1996-2007 ACCESS CO.,LTD.

ACCESS、NetFront は株式会社 ACCESS の日本またはその他の国における商標または登録商標です。

Java、JavaScript は、米国およびその他の国における Sun Microsystems,Inc.の商標または登録商標です。

Flash、Macromedia および Macromedia Flash は、米国およびその他の国における Adobe Systems Incorporated の商標または登録商標です。

VeriSign は、米国およびその他の国における VeriSign,Inc.の商標または登録商標です。

RSA は、米国およびその他の国における RSA Security,Inc.の商標または登録商標です。

"Mozilla"は、Mozilla Foundation の登録商標です。

その他、本ドキュメントに記載されている商標や会社名は、一般的にそれぞれ所有する各社の商標または登録商標です。

[使用条件]

本ガイドラインに関する著作権を含む一切の権利は、当社が管理または保有しています。私的使用その他法律によって明示的に認められる範囲を超えて、これらの情報を使用（複製、改変、ネットワーク上へのアップロード、掲示、送信、頒布、ライセンス、販売、出版等を含む）することは、事前に当社の文書による許諾がない限り、禁止します。

本ガイドラインの内容は、全てが将来のシステムソフトウェアにおいて有効であることを保証するものではありません。当社は理由のいかんにかかわらず、かかる変更によって生じるいかなる損害についても責任を負うものではありません。あらかじめご了解ください。

目次

1 はじめに	5
関連ドキュメント	5
2 ブラウザ基本機能	6
HTTP	6
Cookie	6
キャッシュ	7
HTTP 認証	7
プロキシ認証	8
スキーム	8
HTML	8
SSL/TLS	9
CSS	10
DOM	10
JavaScript	10
画面表示	10
文字表示	10
画像表示	13
画面表示	14
プラグイン	15
3 ブラウザアプリケーション機能	16
アドレス	16
タイトル	16
文字入力	17
ブックマーク	18
タブ	18
ダウンロード	19
アップロード	21
自動 HTTP 認証	22
4 クライアント識別情報	23
リクエストヘッダ	23
JavaScript	26
5 Flash® Player プラグイン	27
バージョン	27

対応デバイス.....	27
非対応機能.....	27
制限事項.....	28
6 "PSP"プラグイン.....	30
対応機能.....	30
プラグイン定義.....	30
アプリケーションへの戻り値書き込み.....	30
インターネットブラウザの終了.....	31
サンプル.....	32
7 "PSP"拡張プラグイン.....	34
対応機能.....	34
プラグイン定義.....	34
インターネットブラウザ拡張ヘッダ情報取得.....	34
コンテンツヒープ最大使用量取得.....	35
サンプル.....	36
8 Appendix.....	38
詳細仕様 HTML.....	38
詳細仕様 CSS.....	40
詳細仕様 JavaScript.....	41
詳細仕様 DOM.....	41
初期値と制限値.....	47
9 変更履歴.....	48

1 はじめに

本ドキュメントは、"PSP"インターネットブラウザ向けの WEB コンテンツを作成される場合に必要クライアント仕様、その他情報、およびガイドラインを記述するものです。

"PSP"インターネットブラウザ向けの WEB コンテンツは本ドキュメントの内容を参考に作成されることを推奨しますが、主に不具合の修正やクオリティアップの目的にて妥当と思われる範囲においてインターネットブラウザのクライアント仕様を変更する場合も想定されます。

本ドキュメントの内容は全てが将来のシステムソフトウェアのバージョンにおいて有効であることを保証するものではありません。本ドキュメント内においては、システムソフトウェアのバージョンによって対応の異なる箇所については"2.00"、"3.10"のように対応バージョンを明記します。

関連ドキュメント

インターネットブラウザが準拠または対応する標準仕様の詳細についてはそれぞれのドキュメントを参照してください。

プロトコル

[RFC2616] Hypertext Transfer Protocol - HTTP/1.1

PERSISTENT CLIENT STATE HTTP COOKIES

http://wp.netscape.com/newsref/std/cookie_spec.html

マークアップ言語

HTML 4.01 Specification <http://www.w3.org/TR/html401/>

XHTML™ 1.1 - Module-Based XHTML <http://www.w3.org/TR/xhtml11/>

XHTML™ Basic <http://www.w3.org/TR/xhtml1-basic/>

CSS

Cascading Style Sheets, level1 <http://www.w3.org/TR/REC-CSS1>

Cascading Style Sheets, level2 <http://www.w3.org/TR/REC-CSS2/>

JavaScript

Standard ECMA-262

<http://www.ecma-international.org/publications/standards/Ecma-262.htm>

2 ブラウザ基本機能

"PSP"インターネットブラウザは、インターネット標準プロトコルに準拠し、HTML 4.01、XHTML 1.1 等の標準マークアップ言語、および CSS、DOM、JavaScript 等に対応しています。本章ではそれらの基本機能について説明します。

HTTP

(1) 対応プロトコル

インターネットブラウザは、以下の標準プロトコルに対応しています。

- HTTP/1.0
- HTTP/1.1

(2) 対応メソッド

インターネットブラウザは、以下の HTTP メソッドに対応しています(使用します)。

- GET
- POST
- HEAD

(3) 対応機能

インターネットブラウザは、その他、以下の機能に対応しています。

- Redirect
- Keep-Alive
- Cookie
- キャッシュ
- HTTP 認証

Cookie

Cookie の保存はユーザ設定またはインターネットブラウザを呼び出したアプリケーションの指定によります。常に保存されるとは限りません。また、ユーザによる確認でキャンセルされる場合もあります。

(1) 対応属性

インターネットブラウザは、以下の Cookie 属性に対応しています。

- Expires
- Path
- Domain

-
- Secure

(2) 制限事項

Cookie 機能には以下の制限があります。

Cookie 保持件数

1 ドメインにつき最大 20 件保持します。

Cookie サイズ

Cookie のサイズは 1 件につき最大 4KB です。

キャッシュ

(1) キャッシュサイズ

キャッシュのサイズはユーザの設定またはインターネットブラウザを呼び出したアプリケーションの指定によります。標準では 512KB です。

また、オフの設定または指定がなされる場合もあります。

(2) 制限事項

キャッシュ機能には以下の制限があります。

キャッシュの保存期間

キャッシュはインターネットブラウザの実行中のみ有効です。インターネットブラウザを終了するとキャッシュの内容は全て消去されます。

なお、インターネットブラウザ実行中であっても、ユーザによる「キャッシュの削除」操作によって消去されます。

キャッシュの無効化

ページ内の META タグによるキャッシュ無効の指定には対応していません。

HTTP レスポンスヘッダ

Cache-Control: no-cache

で指定する必要があります。

HTTP 認証

(1) 対応認証手段

インターネットブラウザは、以下の HTTP 認証手段に対応しています。

- ベーシック認証
- ダイジェスト認証

(2) 入力情報の保存

ベーシック認証、ダイジェスト認証にて入力したユーザ名、パスワードの入力情報は URL 毎に

保存され次回の認証時に自動入力されます。

入力情報の保存は"PSP"本体に対して行われ、ユーザによる「認証情報の削除」操作、または「設定の初期化」が行われるまで有効です。

(3) 自動 HTTP 認証

インターネットブラウザは、アプリケーションから指定された認証情報を用いて、ユーザにユーザ名、パスワードを入力させること無く自動的に HTTP 認証を行う機能に対応しています。詳細については、「ブラウザアプリケーション機能」の章を参照してください。

プロキシ認証

(1) 対応認証手段

インターネットブラウザは、以下のプロキシ認証手段に対応しています。

- ベーシック認証
- ダイジェスト認証

(2) 自動プロキシ認証

インターネットブラウザは、ユーザが設定した認証情報を用いて、ユーザ名、パスワードの入力ダイアログを表示させること無く自動的にプロキシ認証を行う機能に対応しています。

(3) 対応システム

3.00 以降

スキーム

インターネットブラウザは、以下のスキームに対応しています。

- http
- https

なお、file スキームによる"Memory Stick"上のコンテンツの閲覧にも機能的には対応していますが、その動作および将来に渡っての対応を保証するものではありません。

HTML

インターネットブラウザは、以下のマークアップ言語標準仕様に対応しています。

- HTML 4.01
- XHTML 1.1
- XHTML Basic

対応の詳細については、「Appendix:詳細仕様 HTML」を参照してください。

SSL/TLS

インターネットブラウザは、以下の標準仕様に準拠しています。

- SSL v3.0

(1) ルート証明書

"PSP"は、以下のルート証明書を搭載しており、インターネットブラウザにおける SSL 接続にて使用されます。

証明書	搭載システム
Verisign Root CA	2.00 以降
Verisign Class 1 Public Primary CA	2.00 以降
Verisign Class 2 Public Primary CA	2.00 以降
Verisign Class 3 Public Primary CA	2.00 以降
Verisign Class 1 Public Primary CA G2	2.00 以降
Verisign Class 2 Public Primary CA G2	2.00 以降
Verisign Class 3 Public Primary CA G2	2.00 以降
Verisign Class 4 Public Primary CA G2	2.00 以降
Verisign Class 1 Public Primary CA G3	2.00 以降
Verisign Class 2 Public Primary CA G3	2.00 以降
Verisign Class 3 Public Primary CA G3	2.00 以降
Verisign Class 4 Public Primary CA G3	2.00 以降
Verisign RSA Secure Server CA	2.00 以降
Verisign Time Stamping Authority CA	2.00 以降
RSA Root CA	2.00 以降
RSA Security Root CA 1024 (Valicert Class 3 CA)	2.00 以降
RSA Security Root CA 2048 V3	2.00 以降
GeoTrust Root CA	2.00 以降
GeoTrust Global CA	2.00 以降
GeoTrust Equifax Secure CA	2.00 以降
GeoTrust Equifax Secure eBusiness CA-1	2.00 以降
EnTrust Root CA	2.00 以降
EnTrust.net Secure Server CA (CPS)	2.00 以降
Valicert Root CA	2.50 以降
Valicert Class 2 CA	2.50 以降
OmniRoot(CyberTrust CA)	2.50 以降
Omni Baltimore CyberTrust CA	2.50 以降
Omni GTE CyberTrust Global Root CA	2.50 以降
Omni GTE CyberTrust Root CA	2.50 以降
Omni Globalsign Root CA	2.50 以降
Thawte Root CA	2.70 以降
Thawte PremiumServer CA	2.70 以降
Thawte Server CA	2.70 以降

CSS

インターネットブラウザは、以下の標準仕様に対応しています。

- CSS1
- CSS2 の一部

対応の詳細については、「Appendix:詳細仕様 CSS」を参照してください。

DOM

インターネットブラウザは、以下の標準仕様に対応しています。

- DOM level1
- DOM level2 の一部

対応の詳細については、「Appendix:詳細仕様 DOM」を参照してください。

JavaScript

インターネットブラウザは、以下の標準仕様に対応しています。

- JavaScript 1.5 の一部

対応の詳細については、「Appendix:詳細仕様 JavaScript」を参照してください。

画面表示

インターネットブラウザは、"PSP"の表示画面全体を使ってページ表示を行います。サイズは480x272で、縮小表示や拡大表示は行いません。

表示画面サイズを超えるページはユーザ操作によってスクロールしますが、その場合でもスクロールバーの表示は行いません。

操作メニューはページの上に重ねて表示します。

その他全てにおいて、ユーザ操作によってページの表示サイズが動的に変化することはありません。

文字表示

(1) フォントサイズ

インターネットブラウザは、5種類のフォントサイズに対応しています。コンテンツではそれ以上の種類のサイズが指定されたり、直接ポイントを指定される場合があります。また、ユーザ設定により文字サイズを「大」、「標準」、「小」の3段階から選択されます。

インターネットブラウザは以下のルールに従い、コンテンツによるサイズ指定を5種類のフォントサイズにマッピングします。

ポイント指定

コンテンツでのポイント指定に、文字サイズが「大」の場合は1、「標準」の場合は3/4、「小」の場合は1/2、を乗じたポイントでマッピングします。

ポイント	フォントサイズ
16pt 以上	Size 4 (最大)
12pt～15pt	Size 3
10pt～11pt	Size 2
8pt～9pt	Size 1
7pt 以下	Size 0 (最小)

サイズ指定

文字サイズ「大」「標準」「小」のそれぞれの設定において、以下の通りにマッピングします。

サイズ指定	フォントサイズ(大)	フォントサイズ(標準)	フォントサイズ(小)
7	Size 4	Size 4	Size 3
xx-large	Size 4	Size 3	Size 2
6	Size 4	Size 3	Size 2
x-large	Size 3	Size 2	Size 0
5	Size 3	Size 2	Size 0
large	Size 3	Size 1	Size 0
4	Size 3	Size 1	Size 0
medium	Size 2	Size 1	Size 0
3	Size 2	Size 1	Size 0
small	Size 1	Size 0	Size 0
2	Size 1	Size 0	Size 0
x-small	Size 0	Size 0	Size 0
1	Size 0	Size 0	Size 0
xx-small	Size 0	Size 0	Size 0

(2) フォントファミリー

インターネットブラウザは、ラテンフォントについてのみフォントファミリー指定による表示フォントの変更に対応しています。

指定可能なフォントファミリーは以下の通りです。

フォントファミリー

- Roman
- Arial
- Serif
- Sans-Serif

※ただし、font タグでの指定は無効で、span タグ等によるスタイル指定のみ有効です。

例

```
<span style="font-family:Roman">Roman</span>
```

(3) フォントスタイル

インターネットブラウザは、ラテンフォントについてのみ *italic*、**bold** のフォントスタイルおよびフォントウェイト指定に対応しています。

指定可能なフォントスタイルは以下の通りです。

フォントスタイル

- *italic*
- **bold**

例

bold and *italic*
italic

(4) 対応文字セット

インターネットブラウザは、HTML コンテンツにおいて META タグによる指定に基づき文字セットを決定し表示を行います。

表示に対応している文字セットは以下の通りです。

US-ASCII	ASCII
ISO-8859-1	西欧
Windows-1252	西欧
ISO-8859-2 *	中欧
Windows-1250 *	中欧
ISO-8859-3 *	南欧
ISO-8859-4 *	バルト語
Windows-1257 *	バルト語
ISO-8859-5	キリル文字
Windows-1251	キリル文字
KOI8-R *	キリル文字
ISO-8859-7 *	ギリシャ語
Windows-1253 *	ギリシャ語
ISO-8859-9	トルコ語
Windows-1254 *	トルコ語
GB2312	簡体字中国語
GBK	簡体字中国語
GB18030	簡体字中国語
Big5	繁体字中国語
EUC-KR	韓国語
ISO-2022-JP	日本語
Shift_JIS	日本語
EUC-JP	日本語
UTF-8	Unicode

※*印が付いた文字セットについては、「PSP」に内蔵しているフォントの文字種類の関係から、一部正しく表示できない文字が存在します。

(5) 制限事項

文字表示には以下の制限があります。

文字セット判別

インターネットブラウザは、META タグにて

```
<meta http-equiv="Content-Type" content="text/html; charset=iso-8859-1" />
```

のように指定された文字セットで表示を行います。

指定が無かった場合の動作が一定である保証はありません。

- 本体の表示言語の設定に応じて特定の文字セットを選択する
- 必ず Unicode (UTF-8)が選択される

等、リリースによって変わる可能性があります。

フレームの扱い

frame や iframe を用いたページにおいて、フレームのファイルに文字セットの指定が無い場合、ベースのファイルに文字セットの指定があってもそれが継承される保証はありません。

画像表示

(1) 対応フォーマット

インターネットブラウザは、GIF、PNG、JPEG 形式の画像フォーマットの表示に対応しています。それぞれの対応の詳細は以下の通りです。

GIF	GIF87a、GIF89a インタレース、ノンインタレース、透過色、アニメーション
PNG	インタレース、ノンインタレース、透過色、半透明(α チャンネル) ※アニメーション(MNG)には非対応
JPEG	ベースライン DCT、プログレッシブ DCT

(2) 省メモリ設定

"PSP"システムソフトウェア 3.10 以降では、「表示設定」の「省メモリ」を「オン」に設定することで、画像の表示によるヒープメモリ消費を抑えることができます。

その際インターネットブラウザは、以下の条件でコンテンツ内の画像の表示を行います。

共通	• 16 ビットカラーでデコードします。 • 縦×横が 130560 ピクセルを超える場合は、130560 ピクセル以下になるように縮小デコードします。
PNG	• 半透明(α チャンネル)は、無効となります。


(3) 制限事項

画像表示には以下の制限があります。

画像サイズ


画像の表示を行う際に一時的に画像のピクセル数×4byte のヒープメモリを消費します。

ヒープメモリの残り容量は、インターネットブラウザを呼び出したアプリケーションが与えるヒープメモリのサイズ、キャッシュサイズの設定、他のタブの使用状況、等によって変わりますので状況によっては表示ができない場合があります。

画像の表示ができなかった場合は、その画像のサイズまたは **img** タグで指定されたサイズの空白に  のマークが表示されます。


非表示設定

ユーザ設定により、画像やアニメーション画像が非表示となる場合があります。

画像が非表示の場合は、その画像のサイズまたは **img** タグで指定されたサイズの空白に  のマークが表示されます。

アニメーション画像のみが非表示の場合は、そのアニメーション画像の最初のコマが静止画として表示されます。

省メモリ

省メモリ設定がオンの状態において条件に合う形でデコードができなかった場合は、その画像のサイズまたは **img** タグで指定されたサイズの空白に  のマークが表示されます。

画面表示

インターネットブラウザは、画面表示モードの切り替えに対応しています。表示モードの設定によってはコンテンツ側の意図しないレイアウトとなってしまう場合があります。

表示モードには以下のものがあります。

- 標準
元のページサイズ、レイアウトのまま表示を行います。
- ジャストフィットモード
元のレイアウトを保ったまま、画面の横幅に合わせて表示を行います。
文字サイズは「標準」と比べて小さくなる場合があります。また、自動的に折り返しがされる場合があります。
- スマートフィットモード
元のレイアウトを構成し直し、画面の横幅に合わせて表示を行います。
文字サイズは「標準」と比べて小さくなる場合があります。また、自動的に折り返しがされる場合があります。

プラグイン

インターネットブラウザは、プラグイン機能に対応しています。

(1) 対応プラグイン

対応しているプラグインは以下の通りです。

プラグイン名称	MIME-Type	対応システム
Flash® Player プラグイン※	application/x-shockwave-flash	2.70 以降
"PSP"プラグイン	application/x-psp-plugin	2.70 以降
"PSP"拡張プラグイン	application/x-psp-extplugin	2.80 以降

※Adobe の Macromedia® Flash® Player テクノロジーを含んでいます。

各プラグインの詳細については、それぞれの章を参照してください。

(2) 制限事項

プラグイン機能には以下の制限があります。

起動確認

システムソフトウェアのバージョンによっては、プラグイン起動時にユーザ確認のダイアログを表示する場合があります。

3 ブラウザアプリケーション機能

"PSP"インターネットブラウザは、タブ機能、ブックマーク機能、ファイルダウンロード機能など、基本的なブラウジング以外のアプリケーション機能を多数持っています。
本章では、それらのうちコンテンツ作成に関わる部分について説明します。

アドレス

インターネットブラウザは、コンテンツ上のリンクや画像参照に用いるアドレスの長さには特に制限を持ちません。ただし、アドレス編集、ブックマーク登録等、本体にてアドレス文字列を扱う場合に一律で最大 512 文字の制限を掛けています。

512 文字を超えるアドレスは 512 文字以下に自動的にカットされますので、特にブックマーク登録をされ得るようなページにおいては注意が必要です。

以下の機能がこの制限に該当します。

- 現在表示中のページのアドレス編集
- ブックマーク登録

タイトル

インターネットブラウザは、コンテンツのタイトルの長さには特に制限を持ちません。ただし、ブックマーク登録等、本体にてタイトル文字列を扱う場合に一律で最大 512 文字の制限を掛けています。

512 文字を超えるタイトルは 512 文字以下に自動的にカットされますので、特にブックマーク登録をされ得るようなページにおいては注意が必要です。

以下の機能がこの制限に該当します。

- ブックマーク登録

また、表示において、長いタイトルは省略または自動スクロールがなされます。省略やスクロールが発生しない文字数は、日本語・韓国語・中国語の場合で 20 文字程度、それ以外の場合で 30 文字程度を目安としてください。

以下の機能がこの制限に該当します。

- タイトルバーの表示 (自動スクロール)
- マイブックマークのリスト表示 (省略)
- 履歴のリスト表示 (省略)

文字入力

(1) 対応機能

インターネットブラウザは、アドレス入力等、ユーザによる文字入力に対応しています。入力はオンスクリーンキーボードを用いて行いますが、その文字数は最大 512 文字または 256 文字に制限されます。

以下の機能がこの制限に該当します。

- アドレス入力
- ホームページ設定のアドレス入力
- プロキシサーバー設定のアドレス入力
- プロキシ認証設定のユーザ名入力 ※256 文字
- プロキシ認証設定のパスワード入力 ※256 文字
- ブックマーク編集のアドレス入力
- ブックマーク編集のタイトル入力
- フォーム入力
 - input (text, password, file)
 - textarea
 - Flash®コンテンツ内のフォーム
- ダウンロード、リンク先保存、画像保存のファイル名入力
- HTTP 認証ダイアログのユーザ名、パスワード入力 ※256 文字
- プロキシ認証ダイアログのユーザ名、パスワード入力 ※256 文字
- JavaScript の prompt()ダイアログの文字入力 ※256 文字

また、入力可能言語についてもオンスクリーンキーボードの対応に依存します。表示言語設定に応じて、以下の言語の入力が可能です。

入力対応言語	対応システム
ドイツ語	2.00 以降
英語	2.00 以降
スペイン語	2.00 以降
フランス語	2.00 以降
イタリア語	2.00 以降
オランダ語	2.00 以降
ポルトガル語	2.00 以降
ロシア語	2.00 以降
日本語	2.00 以降
韓国語	2.50 以降

(2) 制限事項

文字入力機能には以下の制限があります。

input タグの maxlength 指定

input タグの maxlength アトリビュートによる入力文字数制限は、maxlength 指定値 \geq (多バイトコード文字数 \times 3 + 1 バイトコード文字数) の条件にてバイト数で判定しているため、多バイトコード入力を想定する場合はコンテンツ側で調整が必要になります。

- 多バイトコードの文字入力を想定する場合は、期待文字数 \times 3 を maxlength に指定してください。
- ただしその場合 1 バイトコードは 3 倍の文字数入力できてしまいますので、コンテンツ側で後処理を行ってください。

ブックマーク

インターネットブラウザは、"Memory Stick"へのブックマーク登録に対応しています。以下はブックマーク機能の概要です。

最大登録数	1000 件
登録項目	アドレス、タイトル、最終アクセス日 ※タイトルの無いページの場合はタイトルの代りに アドレスが登録されます。
アドレス文字数	最大 512 文字
タイトル文字数	最大 512 文字
重複登録	可能

タブ

インターネットブラウザは、タブ機能に対応しています。以下はタブ機能の概要です。

最大タブ数	1～3 ※インターネットブラウザを呼び出したアプリケーションの指定により 1～3 で変化します。 "XMB"の「インターネットブラウザ」アイコンから起動した場合は常に 3 となります。 ※最大タブ数を超えて別のタブでページを開こうとした場合は、ユーザによる選択でキャンセルされる場合があります。
ターゲット指定	可能 ※同じターゲットを指定してページを開く事で同じタブが使用されます。 ※ターゲットに"_blank"を指定した場合は常に新しいタブでページを開きます。

JavaScript 確認	あり ※JavaScript によって別のタブでページを開こうとした場合は、ユーザによる選択でキャンセルされる場合があります。
---------------	--

ダウンロード

インターネットブラウザは、ページにリンクされたデータのダウンロードに対応しています。インターネットブラウザは、ダウンロードに際し対象データの MIME-Type(Content-Type)および拡張子を評価し、"Memory Stick"上の適切なフォルダを自動的に決定します。

(1) 評価手順

リンク上で決定を行った場合と「リンク先を保存」を実行した場合とで評価手順が異なります。それぞれの評価手順は以下の通りです。

リンク上で決定

- (1) コンテンツのヘッダを読み込み、MIME-Type を得る。
- (2) Content-Disposition ヘッダ※で指定されたファイル名、あるいは URL から拡張子を得る。
- (3) MIME-Type と拡張子から表示対象コンテンツかどうかを判断する。
- (4) 表示対象コンテンツであった場合は表示を行う。→終了
- (5) MIME-Type と拡張子からダウンロード先のフォルダを決定する。

※ Content-Disposition ヘッダには 2.60 以降のシステムで対応しています。

リンク先を保存

- (1) URL から拡張子を得る。
- (2) 拡張子からダウンロード先のフォルダを決定する。

(2) 表示対象コンテンツ

表示対象コンテンツは以下の通りです。

MIME-Type

- text/html
- text/plain
- image/gif
- image/png
- image/jpeg
- 各プラグイン(有効時のみ)

拡張子

- html, htm
- txt
- gif

- png
- jpg, jpeg

(3) ダウンロード先フォルダの決定

MIME-Type、拡張子とダウンロード先フォルダの関係は以下の通りです。

MIME-Type	拡張子	フォルダ	ファイル名	変更	対応システム
-	jpg	/PSP/PHOTO	※1	可	2.00～2.71
-	jpg	/PICTURE ※3	※1	可	2.80 以降
-	jpeg	/PSP/PHOTO	※1	可	2.00～2.71
-	jpeg	/PICTURE ※3	※1	可	2.80 以降
-	gif	/PSP/PHOTO	※1	可	2.00～2.71
-	gif	/PICTURE ※3	※1	可	2.80 以降
-	png	/PSP/PHOTO	※1	可	2.00～2.71
-	png	/PICTURE ※3	※1	可	2.80 以降
-	bmp	/PSP/PHOTO	※1	可	2.00～2.71
-	bmp	/PICTURE ※3	※1	可	2.80 以降
-	tif	/PSP/PHOTO	※1	可	2.00～2.71
-	tif	/PICTURE ※3	※1	可	2.80 以降
-	tiff	/PSP/PHOTO	※1	可	2.00～2.71
-	tiff	/PICTURE ※3	※1	可	2.80 以降
audio/*	-	/MUSIC ※4	※1	可	2.80 以降
x-audio/*	-	/MUSIC ※4	※1	可	2.80 以降
-	mp3	/PSP/MUSIC	※1	可	2.00～2.71
-	mp3	/MUSIC ※4	※1	可	2.80 以降
-	mp4	/PSP/MUSIC	※1	可	2.00～2.71
-	wav	/PSP/MUSIC	※1	可	2.00～2.71
-	wav	/MUSIC ※4	※1	可	2.80 以降
-	wma	/PSP/MUSIC	※1	可	2.60～2.71
-	wma	/MUSIC ※4	※1	可	2.80 以降
-	aa3	/PSP/MUSIC	※1	可	2.60～2.71
-	aa3	/MUSIC ※4	※1	可	2.80 以降
-	m4a	/PSP/MUSIC	※1	可	2.70～2.71
-	m4a	/MUSIC ※4	※1	可	2.80 以降
-	3gp	/MUSIC ※4	※1	可	2.80 以降
video/vnd.sony.mp4-m4v	-	/MP_ROOT/???M NV01	※2	不可	2.00 以降
video/vnd.sony.mp4-maq	-	/MP_ROOT/???A NV01	※2	不可	2.00 以降
video/vnd.sony.mp4-mav	-	/MP_ROOT/???A NV01	※2	不可	3.00 以降
video/vnd.sony.mp4-mas	-	/MP_ROOT/???A NV01	※2	不可	3.00 以降
video/vnd.sony.mp4-mah	-	/MP_ROOT/???A NV01	※2	不可	3.00 以降
video/*	-	/VIDEO	※1	可	2.80 以降
x-video/*	-	/VIDEO	※1	可	2.80 以降
-	mp4	/VIDEO	※1	可	2.80 以降
-	m4v	/VIDEO	※1	可	2.80 以降

MIME-Type	拡張子	フォルダ	ファイル名	変更	対応システム
-	avi	/VIDEO	※1	可	3.00 以降
上記以外の組み合わせ		/PSP/COMMON	※1	可	-

※1 サーバー指定に準ずるファイル名になります。

※2 ファイル名がインターネットブラウザ側で決定されます。

※3 "UMD"上、"Memory Stick"上のアプリケーションから起動された場合は/PSP/PHOTO となる場合があります。

※4 "UMD"上、"Memory Stick"上のアプリケーションから起動された場合は/PSP/MUSIC となる場合があります。

一部を除き、ダウンロード先のフォルダ、ファイル名はユーザによる変更が可能です。その場合、ファイル名はオンスクリーンキーボードにより任意に編集出来ますが、フォルダはダウンロード対象フォルダからの選択となります。また、インターネットブラウザを呼び出したアプリケーションの指定により変更が制限される場合もあります。

(4) ダウンロード対象フォルダ

ダウンロード対象フォルダは以下の通りです。

フォルダ	対応システム
/PICTURE	2.80 以降
/MUSIC	2.80 以降
/VIDEO	2.80 以降
/PSP/COMMON	2.00 以降
/PSP/PHOTO	2.00 以降
/PSP/MUSIC	2.00 以降
/MP_ROOT/???MNV01	2.00 以降
/MP_ROOT/???ANV01	2.00 以降

アップロード

インターネットブラウザは、フォームからのファイルアップロードに対応しています。input タグの入力ボックスに直接入力する場合は特に制限はありませんが、選択ボタンを押して UI でアップロードするファイルを選択する場合はアップロード対象フォルダ以下のファイルのみが選択対象となります。

(1) アップロード対象フォルダ

アップロード対象フォルダは以下の通りです。

フォルダ	対応システム
/DCIM	2.70 以降
/PICTURE	2.80 以降
/PSP/PHOTO	2.00 以降
/PSP/COMMON	2.00 以降

自動 HTTP 認証

インターネットブラウザは、ゲームプログラム等のアプリケーションから指定された認証情報により、ユーザに意識させること無く自動的に HTTP 認証を行う機能に対応しています。

(1) 適用条件

自動 HTTP 認証が適用される条件は以下の通りです。

- コンテンツ側の Realm(AuthName)が、アプリケーションが指定したもので始まる
- コンテンツの URL が、アプリケーションが指定したもので始まる

※最長一致するものを含む認証情報が採用されます。

- コンテンツ側の認証手段が、アプリケーションが指定したものと一致する

これらが全て満たされた場合に限り自動的に HTTP 認証が行われ、目的のページが表示されます。

(2) サーバ設定

コンテンツ側は、サーバの設定によって HTTP 認証を有効にする必要があります。

その上で、自動 HTTP 認証を行うコンテンツ毎に以下を行います。

- 認証手段を、アプリケーションと合わせたものに設定する
※BASIC 認証あるいは DIGEST 認証のみ利用可能です。
- Realm(AuthName)を、アプリケーションが指定する文字列で始まるものに設定する
※特に指定の無い場合は、“Auto-HTTP-Auth/1.0” が標準で指定されます。
- ユーザ名、パスワードを、アプリケーションと合わせたものに設定する
- URL をアプリケーションから指定する

サーバの設定方法についてはご利用のサーバソフトウェアのマニュアルを参照していただくか、サーバ管理担当者にご相談ください。

4 クライアント識別情報

"PSP"インターネットブラウザは、前章で説明した標準仕様に基きクライアントの識別に有効な情報を取得する手段を提供します。サーバープログラム、CGI、JavaScriptなどでそれらの情報を利用し、コンテンツの動的な振り分け、処理の切り替え、閲覧の制御などを行う事が可能です。本章ではそれらの識別情報について説明します。

リクエストヘッダ

HTTP リクエストヘッダにてインターネットブラウザから送信され、サーバープログラムまたはサーバー変数として CGI から参照可能な識別情報です。

インターネットブラウザが送信するクライアント識別に関わるリクエストヘッダには以下のものがあります。

(1) User-Agent

ヘッダ内容

User-Agent: Mozilla/4.0 (PSP (PlayStation Portable); 2.00)

説明

インターネットブラウザの種類やバージョンを表す標準ヘッダです。

CGI では変数 HTTP_USER_AGENT にて参照します。

この内容はシステムソフトウェアのバージョンアップに伴って変更する予定はありませんが、将来のバージョンにて大幅な機能変更や機能拡張が行われ、サイト側での対応が必須となるような事態が発生した場合には変更をする場合があります。

対応システム

2.00 以降

(2) Accept-Language

ヘッダ内容

Accept-Language: xx-xx

説明

インターネットブラウザ(ユーザ)が許容する言語を表す標準ヘッダです。

CGI では変数 HTTP_ACCEPT_LANGUAGE にて参照します。

インターネットブラウザは、本体の表示言語設定に従い許容する言語を決定します。コンテンツは必要に応じてこのヘッダを参照し表示する言語を切り替えます。

xx-xx 部分は以下の通りに変化します。

表示言語設定	xx-xx	対応システム
ドイツ語	“de”	2.00 以降
英語	“en”	2.00 以降
スペイン語	“es”	2.00 以降
フランス語	“fr”	2.00 以降
イタリア語	“it”	2.00 以降
オランダ語	“nl”	2.00 以降
ポルトガル語	“pt”	2.00 以降
ロシア語	“ru”	2.00 以降
日本語	“ja”	2.00 以降
韓国語	“ko”	2.00 以降
簡体字中国語	“zh-cn”	2.70 以降
繁体字中国語	“zh-tw”	2.70 以降

例

Accept-Language: de

(3) x-psp-productcode

ヘッダ内容

x-psp-productcode: XXX

説明

"PSP"本体の仕向け情報を表す"PSP"独自拡張ヘッダです。

CGI では変数 HTTP_X_PSP_PRODUCTCODE にて参照します。

このヘッダの内容は同一の"PSP"においてはユーザのいかなる使用方法においても変化することはありません。コンテンツは必要に応じてこのヘッダを参照しユーザを適切なページに誘導します。

XXX 部分は以下の通りに変化します。

仕向け	XXX
開発ツール	“TOOL”
日本	“J1”
北米	“UC2”
東欧/欧州	“CEL”
韓国	“KR2”
UK	“CEK”
メキシコ	“MX2”
AU/NZ	“AU3”
南アジア	“E12”
台湾	“TW1”
ロシア	“RU3”
中国	“CN9”

例

x-psp-productcode: J1

対応システム

2.50 以降

(4) x-ppsp-browser

ヘッダ内容

x-ppsp-browser: n.nn (xxx; yyy; zzz; ...)

説明

インターネットブラウザのシステムソフトウェアのバージョンや起動モード等を表す"PSP"独自拡張ヘッダです。

CGI では変数 HTTP_X_PSP_BROWSER にて参照します。

n.nn はインターネットブラウザのシステムソフトウェアのバージョンを表します。これは基本的に"PSP"本体のシステムソフトウェアのバージョンと同期しますが、インターネットブラウザの仕様に変化の無いマイナーバージョンアップの場合等は変更を保留する場合があります。

xxx、yyy、...には、起動モードを表すキーワードやその他のプロパティが入ります。キーワードおよびプロパティの出現する順番は定義しません。

現在定義されているキーワードおよびプロパティは、以下の通りです。

キーワード	意味	対応システム
LX	"XMB"から起動された。	2.50 以降
SX	"XMB"上のアプリケーションから起動された。	2.50 以降
LU	"UMD"上、"Memory Stick"上のアプリケーションから起動された。	2.50 以降

プロパティ	値	対応システム
system	システムソフトウェアバージョン	2.80 以降 (LX,SX)
		3.00 以降 (LU)

例

x-ppsp-browser: 3.10 (LU; system=3.10)

対応システム

2.50 以降

(5) x-ppsp-application

ヘッダ内容

x-ppsp-application: xxx...

説明

インターネットブラウザを呼び出したアプリケーションの名前やバージョン等を表す"PSP"

独自拡張ヘッダです。

CGI では変数 `HTTP_X_PSP_APPLICATION` にて参照します。

`xxx...` はアプリケーションが指定した文字列です。インターネットブラウザがこれを加工することはありません。

例

`x-bsp-application: Everybodys Golf Portable/1.0.0 (UCJS-10001)`

対応システム

2.60 以降

JavaScript

インターネットブラウザの JavaScript エンジンに組み込まれ、コンテンツ内の JavaScript コードから参照可能な識別情報です。

この内容はシステムソフトウェアのバージョンのバージョンアップに伴って変更する予定はありませんが、将来のバージョンにて大幅な機能変更や機能拡張が行われ、サイト側での対応が必須となるような事態が発生した場合には変更をする場合があります。

主な識別情報には以下のものがあります。

(1) navigator オブジェクト

プロパティ	値
<code>userAgent</code>	"Mozilla/4.0 (PSP (PlayStation Portable); 2.00)"
<code>appName</code>	"Mozilla"
<code>appName</code>	"PSP (PlayStation Portable) Internet Browser"
<code>appVersion</code>	"2.00"
<code>platform</code>	"PSP"

(2) screen オブジェクト

プロパティ	値
<code>availWidth</code>	480
<code>availHeight</code>	272
<code>width</code>	480
<code>height</code>	272
<code>colorDepth</code>	32
<code>pixelDepth</code>	32

上記に加え、"PSP"拡張プラグインの機能により JavaScript コードから取得可能な識別情報もあります。詳細については、「"PSP"拡張プラグイン」の章を参照してください。

5 Flash® Player プラグイン

"PSP"インターネットブラウザは、Flash® Player プラグインによる Flash®コンテンツの再生に対応しています。

本章では、"PSP"システムソフトウェア 3.10 が搭載している Flash® Player の機能の概要を説明します。

バージョン

"PSP"システムソフトウェア 3.10 搭載の Flash® Player のバージョンは 6(6,0,72,27)です。基本機能は同バージョンの Flash® Player の仕様に準じます。

対応デバイス

"PSP"システムソフトウェア 3.10 搭載の Flash® Player は以下の入出力デバイスに対応しています。

(1) マウス

アナログパッドおよび決定ボタンでワンボタンマウスに対応しています。ポインタの移動、クリック、ドラッグ&ドロップが可能です。

(2) キーボード

方向キーでキーボードのカーソルキー上/下/左/右に対応しています。

(3) 文字入力

オンスクリーンキーボードでテキストボックスへの文字入力に対応しています。

(4) フォント

デバイスフォントに対応しています。フォントのサイズ、スタイル等の対応範囲はインターネットブラウザと同等です。

(5) サウンド

本体のスピーカー/ヘッドフォンで PCM、ADPCM、MP3 のオーディオ/サウンド再生に対応しています。

非対応機能

"PSP"システムソフトウェア 3.10 搭載の Flash® Player では以下の各機能は非対応となっています。

(1) クリップボード

文字のコピー/カット&ペーストには対応していません。

(2) ビデオ

H.263、Sorenson Video、Motion JPEG 等のビデオデータ再生には対応していません。

(3) コンテキストメニュー

コンテキストメニューの表示/操作および通常そこに含まれる機能には対応していません。

(4) プリンティング

プリントには対応していません。

(5) Live Connect

JavaScript⇔Flash® Player プラグイン間の通信機能には対応していません。

(6) FCS (Flash Communication Server)

FCS への接続、およびそれを必要とする機能には対応していません。

関連機能

- HTTP Tunneling
- Screen Sharing

(7) XMLSocket

XMLSocket 機能を用いた、サーバーとの継続的な接続および通信には対応していません。

(8) 背景透明表示

wmode を"transparent"指定することによる背景の透明表示には対応していません。

(9) ストリーミング

サウンド等のストリーミング再生には対応していません。

制限事項

"PSP"システムソフトウェア 3.10 搭載の Flash® Player では以下の制限があります。

(1) コンテンツのサイズ

サイズの大きな Flash®コンテンツは再生できません。

コンテンツの内容によって異なりますが、平均的にコンテンツファイルサイズの約 3 倍のメモリを必要とします。

その点から、HTML 内に Flash®コンテンツを配置した場合で、ページ内の Flash®コンテンツのサイズの目安は合計約 1.5MB となります。

(2) 外部ファイルのロード

loadVariable()等で外部ファイルからデータをロードする場合、ロードされるタイミングが PC 等とは異なる場合があります。

ロード完了の確認は必ず行う必要があります。

(3) デバイスフォントのサイズ

デバイスフォントのサイズはインターネットブラウザ同様 5 段階です。

コンテンツが期待する領域内に文字列が入りきらない等、PC 等と同等のレイアウトができない場合があります。

(4) mouseout イベントの発生

コンテンツ領域の端にボタン等のオブジェクトを配置した場合、ポインタがコンテンツから出た際に mouseout イベントが発生しない場合があります。

コンテンツ内の表示が想定外の状態になる等の不具合が想定されますのでご注意ください。

(5) ActionScript のネスティング

ActionScript で深いネスティングが発生した場合の制限が PC 等とは異なります。PC 等に比べて浅いネスティング数で終了するようになっています。

6 "PSP"プラグイン

"PSP"インターネットブラウザは、コンテンツに"PSP"専用のプラグインの定義およびそれを使用する JavaScript の記述を行い、ゲームプログラム等のアプリケーションにて対応をすることで、コンテンツとアプリケーションを連携させる機能に対応しています。

本章では、"PSP"プラグインが提供する機能を使用する際のコンテンツ側での対応方法について説明します。

対応機能

"PSP"プラグインが現在対応しているアプリケーション連携機能は、以下の通りです。

連携機能	対応システム
アプリケーションへの戻り値書き込み	2.70 以降
インターネットブラウザの終了	2.70 以降

なお、"PSP"プラグインの連携機能を利用するにはアプリケーション側の指定が必要であり、"XMB"から起動されたインターネットブラウザでは無効となります。

プラグイン定義

連携機能を利用するためには、以下のように HTML コンテンツに"PSP"プラグインのオブジェクト定義を記述します。

記述例

```
<object name="psp" type="application/x-ppsp-plugin"></object>
```

上記記述例で name アトリビュートは "psp" としていますが、任意の名前を指定可能です。JavaScript スクリプト内ではここで指定した名前でオブジェクトを扱うことになります。

アプリケーションへの戻り値書き込み

アプリケーションへの戻り値の書き込みは、アプリケーションが提供するバッファにコンテンツの JavaScript スクリプトからデータを書き込むことで実現します。

コンテンツから戻り値を書き込むには、以下のように記述します。

記述例

```
<script language="JavaScript">
<!--
a = 10;
b = new Array(3);
b[0] = 1; b[1] = 2; b[2] = 3;
if (psp) {
    psp.save(a);
    psp.save(b);
}
//-->
</script>
```

データの書き込みには `save()` メソッドを使用します。書き込まれるデータは JavaScript の内部表現に基く形式の文字列となり、例えば上記記述例で `a` の値が戻り値となった場合は “10” がバッファに書き込まれ、`b` の値が戻り値になった場合は “1,2,3” がバッファに書き込まれます。また、インターネットブラウザの起動中に複数回のデータ書き込みがあった場合は、バッファは常に最後に行われた書き込みで上書きされます。上記記述例では、`b` の値が有効となります。

インターネットブラウザの終了

コンテンツからの指定によりインターネットブラウザを終了させることが出来ます。本機能は、“PSP” プラグインが有効であれば、コンテンツ側の記述のみで実現可能です。インターネットブラウザを終了させるには、以下のように記述します。

記述例

```
<script language="JavaScript">
<!--
a = 10;
if (psp) {
    psp.save(a);
    psp.exit();
}
//-->
</script>
```

インターネットブラウザを終了させるには `exit()` メソッドを使用します。上記記述例では、`save()` メソッドで戻り値として “10” を書き込み、次の `exit()` メソッドで終了させます。

`exit()` メソッドが実行された際のインターネットブラウザの挙動は、ユーザによって終了操作が行われた場合と同じです。

サンプル

以下に、プラグインの有効チェックも含めたサンプルコードを示します。

サンプルコード

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<title>Sample for the PSP plugin</title>
</head>
<body>
<script type="text/javascript">
function exit_browser()
{
    if (psp) {
        psp.exit();
        return true;
    }
    return false;
}

function write_result1()
{
    a = 10;
    if (psp) {
        psp.save(a);
        return true;
    }
    return false;
}

function write_result2()
{
    a = 10;
    b = new Array(3);
    b[0] = 1; b[1] = 2; b[2] = 3;
    if (psp) {
        psp.save(a);
        psp.save(b);
        return true;
    }
    return false;
}
</script>
<p>
<button onclick="write_result1();">click to write result (1)</button>
required:"10"<br/>
<button onclick="write_result2();">click to write result (2)</button>
required:"1,2,3"<br/>

```

```
<button onclick="exit_browser();">click to exit</button><br/>
</p>
</body>
<script type="text/javascript">
if (navigator.mimeTypes && navigator.mimeTypes["application/x-ppsp-plugin"]) {
    var plugin = navigator.mimeTypes["application/x-ppsp-plugin"].enabledPlugin;
    if (plugin) {
        document.write("<object name='ppsp'
type='application/x-ppsp-plugin'></object>");
    }
};
//
// below is also OK.
//
//if (navigator.plugins) {
//    var plugin = navigator.plugins["ppsp plugin"];
//    if (plugin) {
//        document.write("<object name='ppsp'
type='application/x-ppsp-plugin'></object>");
//    }
//}
</script>
</html>
```

MIME-Type “application/x-ppsp-plugin” にてプラグインの有効チェックを行います。

<body>～</body>の下に object タグを記述するようにしていますが、これは現在のシステムソフトウェアバージョンでは object タグを記述した位置に空白が描画されてしまう為です。将来のシステムソフトウェアバージョンにて修正がなされた場合に影響が無いよう、サンプルコード同様の位置に object タグを記述することを推奨します。

7 "PSP"拡張プラグイン

"PSP"インターネットブラウザは、"PSP"プラグインによるアプリケーション連携機能に加え、"XMB"から起動されたインターネットブラウザでも有効な"PSP"拡張プラグインによる連携機能に対応しています。

本章では、"PSP"拡張プラグインが提供する機能を使用する際のコンテンツ側での対応方法について説明します。

対応機能

"PSP"拡張プラグインが現在対応しているアプリケーション連携機能は以下の通りです。

連携機能	対応システム
インターネットブラウザ拡張ヘッダ情報取得	2.80 以降
コンテンツヒープ最大使用量取得	2.80 以降

また、上記以外にもゲームプログラム等のアプリケーションが登録した機能が利用可能な場合があります。

プラグイン定義

連携機能を利用するためには、以下のように HTML コンテンツに"PSP"拡張プラグインのオブジェクト定義を記述します。

記述例

```
<object name="pspext" type="application/x-ppsp-extplugin"></object>
```

上記記述例で name アトリビュートは "pspext" としていますが、任意の名前を指定可能です。JavaScript スクリプト内ではここで指定した名前でオブジェクトを扱うことになります。

インターネットブラウザ拡張ヘッダ情報取得

インターネットブラウザが、サーバへのリクエスト時に付加する、拡張ヘッダの内容を取得することが出来ます。これにより JavaScript コードにおいて、より高精度なクライアント識別が可能となります。

拡張ヘッダ情報を取得するには、以下のように記述します。

記述例

```
<script language="JavaScript">
<!--
if (pspext) {
    psp_browser = pspext.sysGetEnv(' x-psp-browser' );
}
//-->
</script>
```

拡張ヘッダ情報の取得には `sysGetEnv()` メソッドを使用します。このメソッドをヘッダ名を指定して呼び出すことで該当のヘッダの内容が取得出来ます。

取得可能なヘッダ情報は以下の通りです。

ヘッダ	対応システム
x-psp-productcode	2.80 以降
x-psp-browser	2.80 以降
x-psp-application	2.80 以降
User-Agent ※	2.80 以降

※User-Agent は標準ヘッダですが取得に対応しています。

コンテンツヒープ最大使用量取得

コンテンツにより使用されたヒープメモリ量のピーク値を取得することが出来ます。これにより、コンテンツ表示に必要な大凡のヒープメモリ量の把握が可能となり、コンテンツ制作を補助します。なお、このピーク値は全てのタブで使用されたヒープメモリ量の合計であり、また、システムが標準で使用するヒープメモリ量も含まれます。

使用ヒープメモリ量のピーク値を取得するには、以下のように記述します。

記述例

```
<script language="JavaScript">
<!--
if (pspext) {
    peak = pspext.sysGetHeapUsePeak();
}
//-->
</script>
```

使用ヒープメモリ量のピーク値取得には、`sysGetHeapUsePeak()` メソッドを使用します。ピーク値は、バイト単位の整数値で取得されます。

サンプル

以下に、プラグインの有効チェックも含めたサンプルコードを示します。

サンプルコード

```
<?xml version="1.0" encoding="iso-8859-1"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<title>Sample for the PSP ext-plugin</title>
</head>
<body>
<script type="text/javascript">
function get_system_version()
{
    system_version = null;
    if (pspext) {
        psp_browser = pspext.sysGetEnv(' x-ppsp-browser' );
    }
    if (psp_browser) {
        start = psp_browser.indexOf(' system=' , 0) + ' system=' .length;
        end = psp_browser.indexOf(' ;' , start);
        if (end == -1) {
            end = psp_browser.indexOf(' )' , start);
        }
        system_version = psp_browser.substr(start, end-start);
    }
    // return system_version;
    alert(' system_version=' +system_version);
}

function get_heap_peak()
{
    heap_peak = -1;
    if (pspext) {
        heap_peak = pspext.sysGetHeapUsePeak();
    }
    // return heap_peak;
    alert(' heap_peak=' +heap_peak);
}
</script>
<p>
<button onclick="get_system_version();">click to get system version</button>
required:"3.00"<br/>
<button onclick="get_heap_peak();">click to get heap use peak</button> required:some
numeric value<br/>
</p>
</body>
```

```
<script type="text/javascript">
if (navigator.mimeTypes && navigator.mimeTypes["application/x-ppsp-extplugin"]) {
    var plugin = navigator.mimeTypes["application/x-ppsp-extplugin"].enabledPlugin;
    if (plugin) {
        document.write("<object name=%"pspext%"
type=%"application/x-ppsp-extplugin%"></object>%n");
    }
};
//
// below is also OK.
//
//if (navigator.plugins) {
//    var plugin = navigator.plugins["ppsp extplugin"];
//    if (plugin) {
//        document.write("<object name=%"pspext%"
type=%"application/x-ppsp-extplugin%"></object>%n");
//    }
//}
</script>
</html>
```

MIME-Type"application/x-ppsp-extplugin"にてプラグインの有効チェックを行います。

<body>~</body>の下に object タグを記述するようにしていますが、これは現在のシステムソフトウェアバージョンでは object タグを記述した位置に空白が描画されてしまう為です。将来のシステムソフトウェアバージョンにて修正がなされた場合に影響が無いよう、サンプルコード同様の位置に object タグを記述することを推奨します。

8 Appendix

詳細仕様 HTML

インターネットブラウザは、HTML 4.1・XHTML 1.1・XHTML Basic に対応しています。ブラウザが対応している Elements 一覧を以下に記します。ただし、利用可能 Elements に入っている、未対応 Attribute に記載されている Attribute は利用できませんのでご注意ください。

また、全 Element に共通で、下記の Attribute には対応していません。

- ondblclick
- xml:lang
- xmlns

利用可能 Elements	未対応 Attribute	利用可能 Elements	未対応 Attribute	利用可能 Elements	未対応 Attribute
<HTML>	fragments	<BDO>	lang	<BLOCKQUOTE>	
<HEAD>	profile			<Q>	
<TITLE>				<SUB>	
<META>	scheme	<DFN>		<SUP>	
<BODY>	nowrap	<CODE>		<P>	
<DIV>	nowrap	<SAMP>		 	
		<KBD>		<PRE>	width, cols, wrap
<H1>		<VAR>		<INS>	datetime
<H2>		<CITE>			datetime
<H3>		<ABBR>		<PLAINTEXT>	
<H4>		<ACRONYM>		<XMP>	
<H5>				<BLINK>	
<H6>				<MARQUEE>	truespeed
<ADDRESS>					
	compact	<COLGROUP>	align, char, charoff	<A>	hreflang, type, rel, rev, charset
	compact, segenum	<COL>	align, char, charoff	<LINK>	Style, onclick, onmousedown, onmouseup, onmouseover, onmousemove, onmouseout, onkeypress, onkeydown, onkeyup, hreflang, rev, target,
	compact	<TR>	char, charoff, width, bordercolordark, bordercolorlight,		
<DL>	compact	<TH>	headers, scope, abbr, axis, char, charoff, bordercolordark		
<DT>					
<DD>					
<DIR>	compact				
<MENU>	compact				
<TABLE>	summary, bordercolordark, bordercolorlight,				

利用可能 Elements	未対応 Attribute	利用可能 Elements	未対応 Attribute	利用可能 Elements	未対応 Attribute
<TABLE>	ght, cols, hspace, vspace		k, bordercolorligh t		charset
<CAPTION>	valign	<TD>	headers, scope, abbr, axis, char, charoff, bordercolordar k, bordercolorligh t	<BASE>	
<THEAD>	align, char, charoff, valign				
<TFOOT>	align, char, charoff, valign				
<TBODY>	char, charoff				
	longdesc	<EMBAD>	align, alt, border, code, codebase, frameborder, name, palette, pluginpage, pluginurl, units, optional_parm		face
<IMAGE>				<BASEFONT>	face
<OBJECT>	classid, codebase, codetype, archive, standby, tabindex, align, border, accesskey, code	<NOEMBED>		<HR>	
		<MAP>		<FORM>	
<PARAM>	valuetype	<AREA>	alt	<INPUT>	
<APPLET>	codebase, code, name, archive, object, width, height, id, class, title, style, alt, align, hspave, vspacve, mayscript, src	<STYLE>	xml:space	<BUTTON>	
		<CENTER>		<SELECT>	
		<TT>		<OPTGROUP>	style
		<I>		<OPTION>	
				<TEXTAREA>	onselect
		<BIG>		<LABEL>	onfocus, onblur
		<SMALL>		<FIELDSET>	align
		<STRIKE>		<SCRIPT>	
		<S>		<FRAMESET>	framespacing
		<U>		<FRAME>	longdesc
				<NOFRAMES>	
				<IFRAME>	

詳細仕様 CSS

インターネットブラウザは、CSS1 と CSS2 の一部に対応しています。対応していない Properties 一覧を以下に記します。

(1) 対応していない Properties

Properties	Properties	Properties	Properties	Properties
E:lang(cc) E:first-letter E:first-line E:before E:after	content quotes counter-reset counter-increment marker-offset	font-stretch font-size-adjust text-shadow letter-spacing word-spacing text-transform	volume speak pause-before pause-after pause cue-before cue-after	pitch pitch-range stress richness speak-punctuation speak-numeral
unicode-bidi	size	table-layout speak-header	cue play-during azimuth elevation speech-rate voice-family	-wape-accesskey
min-width max-width height min-height max-height	marks page-break-before page-break-after page-break-inside page orphans windows	cursor outline outline-width outline-style outline-color		

(2) 対応していない Values がある Properties

Properties	Values	Properties	Values
display	list-item marker run-in compact table inline-table table-row-group table-column table-column-group table-header-group table-footer-group table-row table-cell table-caption	list-style-type	hebrew armenian georgian cjk-ideographic hiragana katakana hiragana-iroha katakana-iroha
position	Relative	font	icon menu message-box small-caption status-bar
overflow	Scroll auto	text-align	justify <string> inherit
clip	<shape> inherit	caption-side	left right
visibility	Collapse	border-spacing	inherit
		empty-cells	show inherit

詳細仕様 JavaScript

インターネットブラウザは、Java Script 1.5 の一部と ECMAScript 3rd Edition に対応しています。
対応していない Object 一覧を以下に記します。

また、全 Object に共通で、下記の Methods には対応していません。

- toSource()

Properties	Method	Statements
Objects	eval(string) unwatch(prop) watch(prop, handler)	export Statements import Statements
Function	arity	
Arguments	caller	
RegExp	\$1, ...\$9 input(also \$_) lastMatch(also \$&) lastParen(also \$+) leftContext(also \$`) rightContext(also \$') multiline(also \$*) toString() compile(pattern[, flags])	
Error	MemoryError	

詳細仕様 DOM

(1) DOM Level0

Object	Attribute	Method	Event Handler
Navigator	cookieEnabled oscpu vendor vendorSub	javaEnabled() preferens(in DOMString prefName[, in DOMString setValue]) savePreferences() taintEnabled()	
MimeType Array	.name		
Screen	availLeft availTop		
Window	crypto defaultStatus locationbar menubar offscreenBuffer ing outerHeight outerWidth pageXOffset	atob(encodedData) blur() btona(stringToEncode) captureEvents(eventType1[, eventTypeN ...]) disableExpternalCapture() enableExternalCapture() find(in DOMString string, in boolean caseSensitive, in boolean backward)	onabort onclose ondragdrop onerror onkeyup onmove onpaint onreset onresize

Object	Attribute	Method	Event Handler
	pageYOffset personalbar screen screenX screenY scrollbars status statusbar toolbar	focus() handleEvent(event) moveBy(in long horizontal, in long vertical) moveTo(in long x-coordinate, in long y-coordinate) print() releaseEvents(eventType1[, eventTypeN...]) resizeBy(in long horizontal, in long vertical) resizeTo(in long outerWidth, in long outerHeight) routeEvent(event) setHotKeys(in boolean setting) setResizable(in Boolean setting) setZOptions(windowPosition)	onscroll onselect onsubmit new window()
History	current next previous		
Bar	全て		
Crypto	全て		
Layer (layers)	全て		

(2) DOM Core Level 1,2

Object	Attribute	Method
DOMException	code	
DOMImplementation		
DocumentFragment		
Document	doctype	createAttribute(in DOMString name) createAttributeNS(in DOMString namespaceURI, in DOMString qualifiedName) createCDATASection(in DOMString data) createComment(in DOMString data) createDocumentFragment() createElement(in DOMString tagName) createElementNS(in DOMString namespaceURI, in DOMString qualifiedName) createEntityReference(in DOMString name) createProcessingInstruction (in DOMString target, in DOMString data) createTextode(in DOMString data) getElementsByTagNameNS (in DOMString namespaceURI, in DOMString localName) importNode(in Node ImporteNode, in boolean deep)

Object	Attribute	Method
Node	localName namespaceURI prefix	appendChild(in Node newChild) cloneNode(in Boolean deep) hasAttributes() insertBefore(in Node newChild, in Node refChild) isSupported(in DOMString feature, in DOMString version) normalize() replaceChild(in Node newChild, in Node oldChild) removeChild(in Node oldChild)
NamedNodeMap		getNamedItemNS (in DOMString namespaceURI, in DOMString localName) removeNamedItem(in DOMString name) removeNamedItemNS (in DOMString namespaceURI, in DOMString localName) setNamedItem(in Node arg) setNamedItemNS(in Node arg)
CharacterData		appendData(in DOMString arg) deleteData (in unsigned long offset, in unsigned long count) insertData (in unsigned long offset, in DOMString arg) replaceData(in unsigned long offset, in unsigned long count, in DOMString arg)
Attr	OwnerElement	
Element		getAttributeNodeNS(namespaceURI,localName) getAttributeNS(namespaceURI, localName); getElementsByTagNameNS(namespaceURI,localName) hasAttribute(name) hasAttributeNS(namespaceURI,localName) normalize() removeAttribute(in DOMString name) removeAttributeNode(in Attr oldAttr) removeAttributeNS(namespaceURI,localName) setAttributeNode(in Attr newAttr) setAttributeNodeNS(newAttr) setAttributeNS(namespaceURI,qualifiedName)
Text(TextNode)	全て	
Comment	全て	
CDATASection	全て	
DocumentType	全て	
Notation	全て	
Entity	全て	
EntityReference	全て	
ProcessingInstruction	全て	

(3) DOM Event Level 2

Object	Attribute	Method	Event Handler
EventTarget		addEventListener() removeEventListener()	
Event	data height layerX layerY modifiers pageX pageY		
EventException			
UIEvent	detail	initUIEvent()	
MouseEvent:UI Event	altKey button ctrlKey metaKey relatedTarget shiftKey	initMouseEvent()	

(4) DOM HTML Level 1,2

全 Object に共通で、下記の Event Handler には対応していません。

- Ondbclick

Object	Attribute	Method	Event Handler
HTMLDOMImp lementation	全て		
HTMLDocumen t	formName height ids layers tags	captureEvent(event Type) getSelection() handleEvent(event) releaseEvents(eventType) routeEvent(event)	
HTMLLinkEle ment			onclick onkeydown onkeyup onmousedown onmouseout onmouseover onmouseup
HTMLFormEle ment	encoding	handleEvent(event)	onclick onkeydown onkeypress onkeyup onmousedown onmousemove onmouseout onmouseover onmouseup onreset onsubmit
HTMLSelectEle ment		handleEvent(event)	

Object	Attribute	Method	Event Handler
HTMLOptionElement		newOption([text[, value[, defaultSelected [, selected]]]])	
HTMLInputElement	length	handleEvent(event) select()	onselect
HTMLLabelElement			onblur onfocus
HTMLAnchorElement	text x y		
HTMLImageElement		handleEvent(event)	onabort onerror
HTMLObjectElement	contentDocument		onclick onkeydown onkeypress onkeyup onmousedown onmousemove onmouseover onmouseup
HTMLAppletElement	全て		
HTMLScriptElement	event htmlFor		
HTMLTableElement		createCaption() createTFoot() createTHead() deleteCaption() deleteTFoot() deleteTHead() deleteRow(in long index) insertRow(in long index)	
HTMLTableSectionElement		deleteRow(in long index) insertRow(in long index)	
HTMLTableRowElement		deleteCell(in long index) insertCell(in long index)	
HTMLFrameElement	contentDocument		
HTMLIFrameElement	contentDocument		

(5) DOM Style Level2、Document Object Model CSS

Object	Attribute	Method	Event Handler
CSS2Properties	azimuth borderCollapse		

Object	Attribute	Method	Event Handler
	borderSpacing bottom captionSide clip content counterIncrement counterReset cue cueAfter cueBefore cursor elevation emptyCells fontSizeAdjust fontStretch fontVariant letterSpacing markerOffset marks maxHeight maxWidth minHeight minWidth orphans outline outlineColor outlineStyle outlineWidth overflow page pageBreakAfter pageBreakBefore pageBreakInside pause pauseAfter pauseBefore pitch pitchRange playDuring quotes richness right size speak speakHeader speakNumeral speakPunctuation speechRate stress tableLayout textShadow textTransform voiceFamily volume		

Object	Attribute	Method	Event Handler
	windows wordSpacing -wap-marquee -wap-marquee-style -wap-marquee-loop -wap-marquee-dir -wap-marquee-speed -wap-accesskey -wap-input-format -wap-input-required		

初期値と制限値

ユーザ操作、ユーザ設定、アプリケーションからの指定等によって変化し得る項目の初期値および制限値を以下にまとめます。

分類	項目	初期値	制限値/範囲	対応システム
HTTP	接続タイムアウト時間	60 秒	(固定)	2.00 以降
	送信タイムアウト時間	60 秒	(固定)	2.00 以降
	受信タイムアウト時間	120 秒	(固定)	2.00 以降
キャッシュ	キャッシュサイズ	512KB	オフ/ 512KB/ 1024KB/ 2048KB	2.00 以降
Cookie	Cookie 受信	受信する	受信する/ 受信しない/ 確認する	2.00 以降
	サイズ	-	~4KB	2.00 以降
	件数	-	~20 件/1 ドメイン	2.00 以降
表示	画像表示	オン	オン/オフ	2.00 以降
	アニメーション表示	オン	オン/オフ	2.00 以降
	JavaScript	オン	オン/オフ	2.00 以降
	文字サイズ	標準	小/標準/大	2.00 以降
	省メモリ	オフ	オン/オフ	3.10 以降
	表示モード	標準	標準/ ジャストフィット/ スマートフィット	2.00 以降
タブ	タブ数	3	1~3	2.00 以降
文字入力	アドレス入力	-	~512 文字	2.00 以降
	ホームページ設定	-	~512 文字	2.00 以降
	フォーム入力	-	~512 文字	2.00 以降
ブックマーク	保存件数	-	~1000 件	2.00 以降
	アドレス文字数	-	~512 文字	2.00 以降
	タイトル文字数	-	~512 文字	2.00 以降
訪問履歴	保存件数	-	~100 件	2.00 以降
入力履歴	アドレス入力履歴	-	~100 件	2.00 以降
	フォーム入力履歴	-	~100 件	2.50 以降

9 変更履歴

版	章	頁	変更内容
1st	-	-	・システムソフトウェア バージョン 3.10 における追加変更に対応