XHTML-Print/CSS Print Profile Guidelines for PrintEnhanced:1

For UPnP™ Version 1.0
Status: Design Complete Version 1.0
Date: May 4, 2005

This document is being made available to UPnP™ Forum Members pursuant to the UPnP™ Forum Membership Agreement. UPnP™ Forum Members have rights and licenses defined by Section 3 of the UPnP™ Forum Membership Agreement to use and reproduce this document. All such use is subject to all of the provisions of the UPnP™ Forum Membership Agreement.

THE UPNP FORUM TAKES NO POSITION AS TO WHETHER ANY INTELLECTUAL PROPERTY RIGHTS EXIST IN THE PROPOSED DCPS, IMPLEMENTATIONS OR IN ANY ASSOCIATED TEST SUITES. THE PROPOSED DCPS, STANDARDIZED DCPS, IMPLEMENTATIONS AND ANY ASSOCIATED TEST SUITES ARE PROVIDED "AS IS" AND "WITH ALL FAULTS". THE UPNP FORUM MAKES NO WARRANTIES, EXPRESS, IMPLIED, STATUTORY, OR OTHERWISE WITH RESPECT TO THE PROPOSED DCPS, STANDARDIZED DCPS, IMPLEMENTATIONS AND ASSOCIATED TEST SUITES, INCLUDING BUT NOT LIMITED TO ALL IMPLIED WARRANTIES OF MERCHANTABILITY, NON-INFRINGEMENT AND FITNESS FOR A PARTICULAR PURPOSE, OF REASONABLE CARE OR WORKMANLIKE EFFORT, OR RESULTS OR OF LACK OF NEGLIGENCE.

© 2005 Contributing Members of the UPnP™ Forum. All Rights Reserved.
1.1. Contributors

The following is a list of active contributors of this document.

Shivaun Albright, Working Committee Chair
Atsushi Nakamura, Editor

Dean Anderson
Lee Farrell
Melinda Grant
Junzo Ikuta
Hideyuki Kuwano
Yohei Maekawa
Hideki Morozumi
Kazuyuki Murata
Fumio Nagasaka
Kenji Sakuda
Mike Stiltz
Akira Suzuki
Yiruo Yang
1.2. Contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1.</td>
<td>CONTRIBUTORS</td>
</tr>
<tr>
<td>1.2.</td>
<td>CONTENTS</td>
</tr>
</tbody>
</table>

2. OVERVIEW AND SCOPE

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1.</td>
<td>OBJECTIVES</td>
</tr>
<tr>
<td>2.2.</td>
<td>ORGANIZATION</td>
</tr>
<tr>
<td>2.3.</td>
<td>CONVENTIONS</td>
</tr>
<tr>
<td>2.3.1.</td>
<td>Assumptions for Source Examples</td>
</tr>
</tbody>
</table>

3. GUIDELINES

<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1.</td>
<td>REFERENCES</td>
</tr>
<tr>
<td>3.2.</td>
<td>INTERPRETATION GUIDELINES AND CLARIFICATIONS</td>
</tr>
<tr>
<td>3.2.1.</td>
<td>Establishing the Default Page Box</td>
</tr>
<tr>
<td>3.2.2.</td>
<td>Mapping the Page Box to the Page Sheet</td>
</tr>
<tr>
<td>3.2.2.1.</td>
<td>Page Size: auto</td>
</tr>
<tr>
<td>3.2.2.2.</td>
<td>Origin</td>
</tr>
<tr>
<td>3.2.2.3.</td>
<td>Mapping a Page Box to a Larger Page Sheet</td>
</tr>
<tr>
<td>3.2.2.4.</td>
<td>Mapping a Page Box to a Smaller Page Sheet</td>
</tr>
<tr>
<td>3.2.3.</td>
<td>Padding and Borders</td>
</tr>
<tr>
<td>3.2.4.</td>
<td>Units of Measure</td>
</tr>
<tr>
<td>3.2.5.</td>
<td>Absolute Positioning and the Origin</td>
</tr>
<tr>
<td>3.2.6.</td>
<td>Page Break</td>
</tr>
<tr>
<td>3.2.6.1.</td>
<td>Condition on Implicit Page Break in the Normal Flow</td>
</tr>
<tr>
<td>3.2.6.2.</td>
<td>Supressing the First and/or Last Blank Page that Occurred by Forced Page Break</td>
</tr>
<tr>
<td>3.2.7.</td>
<td>Contents Outside the Page Box</td>
</tr>
<tr>
<td>3.2.8.</td>
<td>Images</td>
</tr>
<tr>
<td>3.2.8.1.</td>
<td>Image Rendering Resolution</td>
</tr>
<tr>
<td>3.2.8.2.</td>
<td>Image Rotation</td>
</tr>
<tr>
<td>3.2.8.3.</td>
<td>Image Across Pages</td>
</tr>
<tr>
<td>3.2.8.4.</td>
<td>Images in Table Cells</td>
</tr>
<tr>
<td>3.2.9.</td>
<td>Layer Control</td>
</tr>
<tr>
<td>3.2.9.1.</td>
<td>Static Positioning</td>
</tr>
<tr>
<td>3.2.9.2.</td>
<td>Absolute Positioning</td>
</tr>
<tr>
<td>3.2.10.</td>
<td>Text Across Pages</td>
</tr>
<tr>
<td>3.2.11.</td>
<td>Character Sets and Fonts</td>
</tr>
<tr>
<td>3.2.11.1.</td>
<td>Character Repertoire</td>
</tr>
<tr>
<td>3.2.11.2.</td>
<td>Font Family</td>
</tr>
<tr>
<td>3.2.12.</td>
<td>Word Wrap and Line Breaks</td>
</tr>
<tr>
<td>3.2.12.1.</td>
<td>Word Wrap</td>
</tr>
<tr>
<td>3.2.12.2.</td>
<td>Line Break</td>
</tr>
<tr>
<td>3.2.12.3.</td>
<td>Other Suggested References</td>
</tr>
<tr>
<td>3.2.13.</td>
<td>Forms</td>
</tr>
<tr>
<td>3.2.13.1.</td>
<td>Text Area Element</td>
</tr>
<tr>
<td>3.2.13.2.</td>
<td>Select Element</td>
</tr>
<tr>
<td>3.2.13.3.</td>
<td>Input Element</td>
</tr>
<tr>
<td>3.3.</td>
<td>USAGE GUIDELINES</td>
</tr>
<tr>
<td>3.3.1.</td>
<td>Media Information</td>
</tr>
<tr>
<td>3.3.2.</td>
<td>Paging</td>
</tr>
<tr>
<td>3.3.2.1.</td>
<td>@page Size</td>
</tr>
<tr>
<td>3.3.2.2.</td>
<td>Page Break Control</td>
</tr>
<tr>
<td>3.3.2.3.</td>
<td>Page Size Change within a Document</td>
</tr>
<tr>
<td>3.3.3.</td>
<td>Pixels</td>
</tr>
<tr>
<td>3.3.3.1.</td>
<td>Unit of Measure</td>
</tr>
<tr>
<td>3.3.3.2.</td>
<td>Image Resolution</td>
</tr>
</tbody>
</table>

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
2. Overview and Scope

2.1. Objectives

This document provides guidelines for PrintEnhanced:1 Print Service developers and Control Points and other content authors with the following objectives:

- Enable an improved level of output consistency on print media among printing devices which support XHTML-Print and CSS Print Profile.
- Suggest methods for content authors to avoid ambiguities within the current W3C specifications referenced by the PrintEnhanced:1 service.
- Provide guidance to Print Service developers in areas of ambiguity.
- Provide leveragable samples for content authors to build from.

2.2. Organization

To achieve the above objectives, the document is organized into the following sections:

- Clarification and Interpretation Guidelines for XHTML-Print [XHTML-PRINT] /CSS Print Profile [CSSPP] Content parsing and Layout:
  - Guidelines on how PrintEnhanced:1 compliant printers should interpret XHTML-Print [XHTML-PRINT] and CSS Print Profile [CSSPP] expressions that are ambiguous and may lead to varying outputs
- XHTML-Print /CSS-Print Usage Guidelines
  - Guidelines on how XHTML-Print [XHTML-PRINT] and CSS Print Profile [CSSPP] contents intended for output from PrintEnhanced:1 compliant printers should be created in order to achieve output consistency among various printers.
- Sample Template XHTML-Print [XHTML-PRINT] /CSS Print Profile [CSSPP] source and corresponding expected output for several simple photo layout contents.
Output of XHTML and CSS to paged media, such as printed-paper is different from display on screen media in that pagination is considered. This, with the fact that some ambiguity exists in the interpretation of XHTML and CSS is thought to lead to various inconsistent outputs among output devices (i.e. printers), which is unexpected. The interpretation guidelines and usage guidelines in this document will assure a certain level of output consistency among printers, regardless of layout complexity of the content. In addition, use of sample templates for simple photo layouts will serve as reference to both printer implementations and content authors to achieve a high level of consistency.

2.3. Conventions

2.3.1. Assumptions for Source Examples

All examples assume default values as recommended for the Printer’s default style sheet in Section 8.5 of [CSSPP] unless otherwise stated.
3. Guidelines

3.1. References

This section lists the references that this document refers to and the tag inside square brackets that is used for each such reference:


3.2. Interpretation Guidelines and Clarifications

This section provides guidelines and clarifications for PrintEnhanced:1 compliant printer implementations, especially where the W3C reference specifications are ambiguous or allow varying interpretations that can lead to visibly different output.

It is strongly recommended that printers follow the informative guidelines provided in Section 8.5 of [CSSPP] in establishing default style sheet values.

In addition, PrintEnhanced:1 printer implementations should comply with the following guidelines in order to achieve greater output consistency.

3.2.1. Establishing the Default Page Box

When an @page ‘size’ attribute is specified, the content area of the infinitely sized canvas is constrained as indicated by the page size. When the @page ‘size’ attribute is not specified, it is recommended that the canvas size is constrained as though

```css
    @page { size: auto; }
```

were specified.

This establishes a default page box the same size as the target page sheet. The resulting page area is the initial containing block. The page area is the page box minus the page margins.
3.2.2. Mapping the Page Box to the Page Sheet
The current CSS specifications do not specify how the page box is mapped to the target page sheet. This document recommends the following guidelines:

3.2.2.1. Page Size: auto
When the size of the page box is established with the value ‘auto’, the page box should exactly overlay the target page sheet; i.e., the edges of the page box and the edges of the target sheet are aligned.

3.2.2.2. Origin
If there is no ancestor with a 'position' of 'absolute', 'relative' or 'fixed', the origin with respect to the positioning of elements is the upper-left corner of the page box.
While this is unambiguous for single-page documents, CSS is ambiguous when the document contains more than one page. See Section 3.2.5 for further details.

3.2.2.3. Mapping a Page Box to a Larger Page Sheet
When the size of the page box is smaller than the size of the physical page sheet, the printer may (in order of preference):

1. Prompt the user to provide media the size of the page box.
2. Center the page box on the page sheet without scaling.
3. Position the top and left edges of the page box at the top and left edges of the page sheet.
4. Uniformly scale the page box up as much as possible to fit the page sheet while preserving the aspect ratio of the page box.
5. Non-uniformly scale the page box to take up the entire page sheet.

3.2.2.4. Mapping a Page Box to a Smaller Page Sheet
When the size of the page box is larger than the size of the physical page sheet, the printer may (in order of preference):

1. Prompt the user to provide media the same size as the page box.
2. Scale the page box down to fit the page sheet.
3. Crop the portions of the page box which do not fit on the page sheet, centering the page box with respect to the page sheet.

Crop the portions of the page box which do not fit on the page sheet, aligning the top and left edges of the page box and the page sheet.

3.2.3. Padding and Borders
As defined in CSS, padding and border widths are NOT included in the content width /height.

Source

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.2.4. Units of Measure
Printers implement a wide range of resolutions and pixel depth. To improve consistency of printed output, and for compliance with the recommendation in Section 4.3.2 of [CSS2_1], it is recommended that Printers implement the pixel unit of measure as approximately 1/96 inch, or 0.26 mm, unless dimensions are otherwise specified.

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
### 3.2.5. Absolute Positioning and the Origin

From Cascading Style Sheets Level 2 Revision 1 [CSS2_1], Section 10.1: 'In paged media, an absolutely positioned element is positioned relative to its containing block ignoring any page breaks (as if the document were continuous). The element may subsequently be broken over several pages.'

Therefore, in the absence of an ancestor with a 'position' of 'absolute', 'relative' or 'fixed', the origin for any absolutely positioned content within a document would seem to be the origin of the first page of the document.

However, Section 13.2 of [CSS 2_1] says “the edges of the page area act as the initial containing block...” One might expect that pages subsequent to the first page of a document would establish new “initial containing blocks”.

Indeed, what other reasonable interpretation can be made? Since absolute positioning is done relative to the containing block, this would seem to imply the origin should reset to the top of the next page upon a page break.

The W3C CSS WG needs to resolve or clarify this ambiguity. Until such time, this Guideline makes no recommendation as to which interpretation a printer should make. Rather, Section 3.3.3 strongly cautions content authors to avoid these ambiguous scenarios.

### 3.2.6. Page Break

This section describes the recommended behaviors of PrintEnhanced:1 printers for page break control

#### 3.2.6.1. Condition on Implicit Page Break in the Normal Flow

The printer should not break the current page until content is seen which is placed on the subsequent page.

#### 3.2.6.2. Suppressing the First and/or Last Blank Page that Occurred by Forced Page Break

If a document uses a common forced page break style on all pages including the first and/or last pages (to make the document source simple), user agents may suppress the first and/or last blank page that occurs by the forced page break.

### 3.2.7. Contents Outside the Page Box

- Contents, or portions of contents that overflow horizontally out of the page box should not be printed.

- Contents, or portions of contents that overflow above (vertically negative out of ) the page box should not be printed.

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.2.8. Images

3.2.8.1. Image Rendering Resolution
When size information is not provided for an image (i.e., neither height nor width is specified, or they are both set to 'auto'), it is recommended that the image is printed at about 96 dpi, or about 5 dots per mm. (This is consistent with the resolution used for computer displays, and will promote consistency in output across displays and printers.)

3.2.8.2. Image Rotation
Use of the JFIF APP0 marker and the EXIF APP1/APP2 markers to determine image orientation are deprecated; any rotation information in such markers should be ignored. The CSS3 Paged Media Module [CSS3_PM] image-orientation property should be used instead.

3.2.8.3. Image Across Pages
When an image extends beyond the bottom of a page, and it is not positioned (that is, it is statically positioned in the normal flow), the image should NOT be divided, but should be placed on the next page. When the image is positioned (that is, its position property is absolute, fixed, or relative), three possible outputs are allowed due to ambiguity in [CSS 2_1].

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.2.8.4. Images in Table Cells

When an image is laid out inside a table cell, the following guidelines apply:

1. If an ‘image-orientation’ is specified, the image is first rotated accordingly. After rotation, the ‘width’ property corresponds to the size of a row of pixels, while the ‘height’ property corresponds to the size of a column of pixels.

2. If both ‘width’ and ‘height’ properties specify a length, the image is scaled to the indicated height and width and centered within the table cell. If portions of the image extend beyond the content area of the table cell, the ‘overflow’ property determines whether those portions are displayed or cropped.

3. If either the ‘width’ or the ‘height’ property specifies a length (but not both; the other property is absent or ‘auto’), the indicated aspect of the image is scaled to the specified dimension, and the other aspect of the image is scaled so that the intrinsic aspect ratio of the source image is preserved.

4. If neither the ‘width’ nor the ‘height’ property specifies a length, the intrinsic size and aspect ratio of the source image should be preserved. If the intrinsic size is not known, and in the absence of any other

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
‘clues’ as to the intended output size, it is recommended that the image is rendered at approximately 96
pixels per inch or 0.26 mm.

### 3.2.9. Layer Control

This section describes the recommended behaviors of PrintEnhanced:1 printers for layer control.

#### 3.2.9.1. Static Positioning

In the normal flow, the block boxes can be positioned to overlap by specifying the “margin-top, margin-left margin-
bottom, margin-right” property. When block boxes visually overlap, those appearing later in the source should be
painted nearer to the user and, those appearing earlier in the source should be painted further from the user.

#### 3.2.9.2. Absolute Positioning

The block boxes can be positioned to overlap by specifying the “top, left, bottom, right” property. When block-
boxes visually overlap, those appearing later in the source should be painted nearer to the user and, those appearing
earlier in the source should be painted further from the user.

### 3.2.10. Text Across Pages

A text box laid out according to the normal flow which extends beyond the bottom of the page box should be
divided between line boxes to fill the first page. Which line boxes of the paragraph are on the current page and
which are on the next page is determined by the widows and orphans properties. With absolute positioning, two
possible outputs are allowed due to ambiguity in [CSS 2_1].

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.2.11. Character Sets and Fonts

3.2.11.1. Character Repertoire
PrintEnhanced:1 compliant printers should support Basic Latin (U+0000-U+007F) and Latin-1 Supplement (U+0080-U+00FF) characters of the set ISO/IEC 10646. When PrintEnhanced:1 printers encounter characters that are recognized but not renderable, they should substitute another rendering that gives the same meaning, or provide a way to indicate that normal rendering has not been successful.

3.2.11.2. Font Family
If PrintEnhanced:1 printers support extended character repertoires, then "serif", "sans-serif" and "monospace" font families should also include the extended characters.
3.2.12. Word Wrap and Line Breaks

3.2.12.1. Word Wrap

- Complete words that won't fit on the current line should be wrapped onto the next line.

See Section 9.4.2 of CSS 2.1 [CSS2_1] for exceptions and further details.

- Shortened words, such as “what’s”, should be processed as one word.

3.2.12.2. Line Break

Spaces and hyphens should be treated as a line break opportunity, and line breaks should not occur within a word.

The following algorithm is provided as an example of one optional approach to determine line breaking.

- Any of the following characters shown in the following tables, if appearing at the end of a word should not wrap to the next line.
- Any of the following single characters shown in the following tables if immediately preceding an associated word should not be split from that word for word wrap purposes. The character should be wrapped along with the word.
- Handling of repetitive use of the characters shown in the following tables will be implementation dependent, and out of scope of this guideline.

Characters that Should NOT be Placed on Top of the Line

Category-1; Global

<table>
<thead>
<tr>
<th>Unicode code point</th>
<th>U+0021</th>
<th>U+0029</th>
<th>U+002C</th>
<th>U+002E</th>
<th>U+003A</th>
<th>U+003B</th>
<th>U+003F</th>
<th>U+005D</th>
<th>U+007D</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>!</td>
<td>)</td>
<td>,</td>
<td>:</td>
<td>;</td>
<td>?</td>
<td>]</td>
<td>}</td>
<td></td>
</tr>
</tbody>
</table>
### Characters that Should Not be Placed at the End of the Line

**Category-1; Global**

<table>
<thead>
<tr>
<th>Unicode code point</th>
<th>U+0028</th>
<th>U+005B</th>
<th>U+007B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Character</td>
<td>(</td>
<td>{</td>
<td>[</td>
</tr>
</tbody>
</table>

**Category-2; Regional**

**Japanese language**

<table>
<thead>
<tr>
<th>Unicode code point</th>
<th>U+FF62</th>
<th>U+FF08</th>
<th>U+3014</th>
<th>U+FF3B</th>
<th>U+FF5B</th>
<th>U+3008</th>
<th>U+300A</th>
<th>U+300C</th>
</tr>
</thead>
<tbody>
<tr>
<td>shift-jis</td>
<td>A2</td>
<td>8169</td>
<td>816B</td>
<td>816D</td>
<td>816F</td>
<td>8171</td>
<td>8173</td>
<td>8175</td>
</tr>
<tr>
<td>Character</td>
<td>]</td>
<td>)</td>
<td>)</td>
<td>}</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
3.2.12.3. Other Suggested References

The Unicode Standard Annex #14 Line Breaking Properties [LINE-BREAK] provides a comprehensive treatment of line breaking recommendations, for best practice and further references. However, it is to be expected that many PrintEnhanced:1 printers will not implement line break algorithms requiring linguistic analysis or large pair-based tables.

3.2.13. Forms

PrintEnhanced:1 compliant printers should comply with the following guidelines in interpreting and rendering the following form elements.

3.2.13.1. Text Area Element

- Scrollbars may or may not be printed for text area elements.
- A slider should be printed if the text area contents overflow the input field and overflow=’hidden’.
- The text area may be enlarged to encompass overflow contents when overflow=’visible’.
- A slider should NOT be printed if the contents of the text area do not overflow from the input field.
- A slider should NOT be printed if there are no contents in the text area.
- The design and layout of the scroll bar is implementation dependent.
A portion of it will be printed.

An example of a text area output where the text does not overflow from the textarea.

If the designated text area is too small to represent all the content, a portion of it will not be printed.

An example of a text area output where the text overflows from the textarea.

An example of a text area output where there is no text in the textarea.

3.2.13.2. Select Element

- A Down-Arrow may or may not be printed if the size attribute is unset or set as 1.
- A Scroll bar may or may not be printed if the size attribute set as greater than 1.
- A Slider should be printed if the number of option elements is greater than the value of set attribute.
- The design and layout of the scroll bar is implementation dependent.
- The first selected element should be printed.
### Source

<table>
<thead>
<tr>
<th>Size attribute not set</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;select&gt;</code> <code>&lt;option&gt;Object1&lt;/option&gt;</code> <code>&lt;/select&gt;</code></td>
<td>Object1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Size attribute set to 1</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;select size=&quot;1&quot;&gt;</code> <code>&lt;option&gt;Object1&lt;/option&gt;</code> <code>&lt;/select&gt;</code></td>
<td>Object1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Size attribute set to greater than 1</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;select size=&quot;2&quot;&gt;</code> <code>&lt;option&gt;Object1&lt;/option&gt;</code> <code>&lt;option&gt;Object2&lt;/option&gt;</code> <code>&lt;/select&gt;</code></td>
<td>Object1, Object2</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Number of option elements are greater than size attribute value</th>
<th>Expected Output</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;select size=&quot;2&quot;&gt;</code> <code>&lt;option&gt;Object1&lt;/option&gt;</code> <code>&lt;option&gt;Object2&lt;/option&gt;</code> <code>&lt;option&gt;Object3&lt;/option&gt;</code> <code>&lt;/select&gt;</code></td>
<td>Object1, Object2</td>
</tr>
</tbody>
</table>
3.3. Usage Guidelines

This section provides guidelines on how XHTML-Print [XHTML-PRINT] and CSS Print Profile [CSSP] contents intended for output from PrintEnhanced:1 compliant printers should be created in order to achieve output consistency among various printers. Though the guidelines do not have to be applied to contents with non-print media as it’s primary intended output, its strongly recommended to follow these guidelines in authoring contents that prioritize output consistency on paged media.

3.3.1. Media Information

Contents intended for printing should specify media information in it’s style sheet by the use of @media print{}.

3.3.2. Paging

Paged media differ from screen media in that it introduces the concept of pagination. Contents intended for PrintEnhanced:1 compliant printers should be conscious of page media by following the guidelines below. Though contents that do not include page information, or are intended for screen media only can be printed, they may exhibit less output consistency among printers.

3.3.2.1. @page Size

In order for the printer to recognize the intended page size of the content, @page size information should always be included in the contents. The page size property is essential for printers to maintain the intended output design, since the contents structures typically have strong relationships with media size.

3.3.2.2. Page Break Control

There are two models of use which should be carefully considered when constructing XHTML and CSS content intended for printing: one where the Printer is primarily in control of the placement of page breaks, and the other where the Content Author is primarily in control. Each has its merits and disadvantages.

3.3.2.2.1. Implicit (Printer) Control

When it is important that all the content be printed, but it is not important that the content is laid out in a certain fashion, it is best to by and large use static positioning, and let the Printer decide how to lay out the contents and where to put page breaks. This model is also preferable when the contents contain mixed fonts and various point sizes of text that flows across pages. In this case it’s better to let the printer do the page breaks, attending to widows and orphans and putting the breaks between line boxes as required by CSS2.1. In this scheme, page-break-* would be used sparsely for stylistic reasons, such as to ensure that major headings start at the top of the page, or page breaks are avoided inside list item elements. Although the exact layout of the document will vary from printer to printer, this approach will yield pleasing results from most all printers.

3.3.2.2.2. Explicit Control

When it is important that contents are laid out in a very specific and deterministic manner, page break information should be explicitly provided for contents intended for inclusion in one page. The use of page-break-before, page-break-after, page-break-inside, and named pages [CSS3_PM] is recommended to inform the printer of paging information. Note that these properties only apply to block elements.
3.3.2.3. Page Size Change within a Document.

Changing the page size or orientation within a single html element is allowed. The example below intends to show that a document can have more than one page-configuration and, alteration of the media-size within the document results in page-break. This is achieved by defining different @page guidelines for each page size or orientation to be used within the document source.

- If a block box with inline content has a 'page' property that is different from the preceding block box with inline content, then one or two page breaks are inserted between them, and the boxes after the break are rendered on a page box of the named type.

- @page rules apply to pages, not to elements, and so inheritance does not apply as pages do not have children.

- @page rules *do* cascade however, as several different rules may match the same page, and their declarations will need to be ordered by specificity / priority.

- While a preceding block element has the @page value, “initial value” is applied to the subsequent element and if they are different; then a page break should be generated.

Example 1: Page Size Change

Description

In this example, the generation of the fourth page deserves some additional explanation:

- The <h2> element of class 'newpage-1' generates a new page (the second page), because of the explicit 'page-break-before' in the newpage-1 selector:
  
  “h2.newpage-1 {page-break-before: always;}”

- The <h2> element of class 'newpage-2' generates another new page (the third page), because the page property overrides the initial value of A4 portrait with a value of A4 landscape; the 'newpage-2' contents are therefore placed on a landscape page:
  
  “h2.newpage-2 {page: a4-landscape;}”

- The page attribute for the final <h2> element cascades back to the page size and orientation that were in effect before the <h2> of class 'newpage-2' (since the scope of that element has closed), i.e., A4 portrait, causing another page break and generating the fourth page.
Source

```html
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Page Size Change within a Document</title>
<style type="text/css">
@media print {
@page {size:A4 portrait;}
@page a4-landscape {size:A4 landscape;}
h2.newpage-1 { page-break-before:always; }
h2.newpage-2 { page:a4-landscape; }
}
</style>
</head>
<body>
<h2>Section-1: Portrait Page</h2>
<p>page one contents</p>
<h2 class="newpage-1">Section-2: Portrait Page</h2>
<p>page two contents</p>
<h2 class="newpage-2">Section-3: Landscape Page</h2>
<h2>Section-4: Portrait Page</h2>
<p>page four contents</p>
<p>(Page three has no content other than the Section heading.)</p>
</body>
</html>
```

Expected Output

Section-1: Portrait Page
page one contents

Section-2: Portrait Page
page two contents

Section-3: Landscape Page

Section-4: Portrait Page
page four contents
(Page three has no content other than the Section heading.)

Example 2: Nested element

Source

```html
<html xmlns="http://www.w3.org/1999/xhtml">
```
Section-1: Portrait Page
page one contents

Section-2: Portrait Page
page two contents

Section-3: Landscape Page
page three contents
Named page’s property of the parent is inherited to the nested element: "page:a4-landscape"

Section-4: Portrait Page
page four contents

Example 3: Not-Nested element
Source

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
<head>
<title>3.3.2.3 Page Size Change within a Document</title>
<style type="text/css">
@media print {
@page { size:A4 portrait; }
@page a4-landscape { size:A4 landscape; }
h2.newpage-1 { page-break-before:always; }
*.newpage-2 { page:a4-landscape; }
}
</style>
</head>
<body>
<h2>Section-1: Portrait Page</h2>
<p>page one contents</p>
<h2 class="newpage-1">Section-2: Portrait Page</h2>
<p>page two contents</p>
<h2 class="newpage-2">Section-3: Landscape Page</h2>
<p class="newpage-2">page three contents</p>
<p class="newpage-2">all non-nested elements should have a common @page property</p>
<h2>Section-4: Portrait Page</h2>
<p>page four contents</p>
</body>
</html>

Expected Output

Section-1: Portrait Page
page one contents

Section-2: Portrait Page
page two contents

Section-3: Landscape Page
page three contents
all non-nested elements should have a common @page property

Section-4: Portrait Page
page four contents

h2.newpage-1 { page-break-before:always; }

<h2>: *.newpage-2 { page:a4-landscape; }

Same as 'named page' : *.newpage-2

<h2>: @page { size: A4 portrait }

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.3.3. Pixels

3.3.3.1. Unit of Measure
Printers are expected but not required to implement the pixel unit of measure as approximately 1/96 inch or .26 mm. Unless there are good reasons to use the pixel length, consideration should be given to use a measure guaranteed to be consistent across user agents, such as the millimeter.

3.3.3.2. Image Resolution
It is strongly recommended that either height or width or both are specified for image size. (The XHTML-Print specification allows printers to omit images when no size information is provided.) When no size information is provided, or both height and width are set to 'auto', printers are expected but not required to render images at about 96 dpi or 5 dots per mm.

3.3.4. Positioning (absolute positioning / static positioning)
The following should be taken into consideration when creating XHTML-Print / CSS Print contents intended for output from PrintEnhanced:1 compliant printers.

- Absolute positioning should ONLY be used for contents that are sure to fit within the current page box. Avoid the use of absolute positioning unless positioning is carefully considered to fit within the specified page size.
- Contents that may not fit in one page should be laid out by one of the following methods:
  - Paginate by explicitly using page-breaks and named pages, in case of using absolute positioning.
  - Use static positioning for layout.
- Contents creation that does not consider pagination should use static positioning.
- Objects should not be positioned either entirely or partly outside of the page box, unless cropping of the overflowed content is intended. In this case, the overflow property should be set to 'hidden'.
- If content authors apply forced page-break style to every page, the style may generate either an empty start page or an empty end page. To avoid this inconsistency, content authors should:
  - Avoid making documents that begin with a FORCED page break
  - Avoid making documents that end with a FORCED page break

- The following CSS styling is recommended to establish a div which maps to the page area. Elements can then be positioned relative to the div, which is equivalent to positioning relative to the page area:

Source

```html
html, body { height: 100%; }
div.page-div {width: 100%; height: 100%; position: relative; page-break-after: always; overflow: hidden;}
div.page-div-end {width: 100%; height: 100%; position: relative; overflow: hidden;}
```

The following example makes use of the CSS recommended above. There is no ambiguity in the expected output:

Source

```html
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>page-div</title>
<style type="text/css">
```

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
@media print {
  @page { size: auto; } /* or whatever size is desired */
}

html, body { height: 100%; }
div.page-div { width: 100%; height: 100%; position: relative; page-break-after: always; overflow: hidden; }
div.page-div-end { width: 100%; height: 100%; position: relative; overflow: hidden; }
p.textbox1 { position: absolute; left: 10mm; top: 10mm; }
p.textbox2 { position: absolute; left: 10mm; top: 20mm; }

</style>
</head>
<body>
<div class="page-div">
  <p class="textbox1">contents page1</p>
</div>
<div class="page-div-end">
  <p class="textbox2">contents page2</p>
  <p>static position</p>
</div>
</body>
</html>

**Expected Output:**

(0,0)  (0,0)

![Diagram showing contents page 1 and static position](image)
3.3.5. Images

3.3.5.1. Image Size and Page Size

Consideration is recommended for the relationship between the size of the image and the page size. The size of the image should be within the size of the page size specified by @page unless it is intended that part of the image should be cropped (i.e., not printed). Output consistency is not assured for images that exceed the page size.

When part of the image should be cropped, it is important that the overflow property be set to ‘hidden’, so that the printer understands it need not take steps to display the overflowed content.

For example, when the intrinsic aspect ratio of an image differs from the aspect ratio of the destination media, and the Content Author wishes to generate full-bleed output of the image, the width or the height of the image object may intentionally exceed the width or height of the page sheet. (Explanation is given in section 3.4, and examples are given in section 3.4.3.1.3 and 3.4.3.2.1) Information on full bleed can also be referenced in sections 3.2.2 f) and 3.8.6 of PrintEnhanced:1 [PE1].

3.3.5.2. Image Rotation

The following should be taken into consideration when creating XHTML-Print / CSS Print Profile contents that include image rotation.

- Image rotation should be specified using CSS3 rotation mechanisms. Do not rely on the EXIF App markers to rotate images.

Ex. CSS - img { image-orientation:90deg; width: W; height: H; }

Source image (input) expected result (output)

Rotate the image clockwise by 90-degree increments (90,180,270) relative to the orientation of the page

- If both CSS rotation property and EXIF App marker exist for a given image, CSS image rotation property will take precedence over EXIF App markers

- Whether the printer references the EXIF App markers for image rotation will be device/application dependent.
3.3.5.3. URI of Images

When an XHTML-Print file is sent using the HTTP POST method, either the base element should be provided if relative URIs are used, or full URIs should be specified, in order to reference external objects.

The following example makes use of the base element recommended above. There is no ambiguity in the expected output:

Source

```html
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>URI of Images</title>
;base href="http://www.upnp.com/images/" />
</head>
<body>
<div><img src="hdtv320x180.jpg" alt="photo" /></div>
</body>
</html>
```

In the example above, given
```html
;base href="http://www.upnp.com/images/" />
the IMG element
```
3.3.6. Layer Control

Layered presentations using static positioning and absolute positioning are described below.

Static positioning: In the normal flow, the block boxes can be positioned to overlap by specifying the "margin-top, margin-left, margin-bottom, margin-right" property. When block boxes visually overlap, those appearing later in the source should be painted nearer to the user and, those appearing earlier in the source should be painted further from the user.

Absolute positioning: The block boxes can be positioned to overlap by specifying the “top, left, bottom, right” property. When block-boxes visually overlap, those appearing later in the source should be painted nearer to the user and, those appearing earlier in the source should be painted further from the user.

The next example shows a recommended method to overlap block boxes and related components. In this example, the device will likely interpret the source code shown below as follows: First, the background-color of `<body>` is painted on the canvas, followed by `<div>` block box, which contains image with background color. Then, `<p>` block box, which includes inline (explicitly specified with `<span>`) text with border and background-color, is painted. Expected output is illustrated below.

Source

```
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Layer Control</title>
<style type="text/css">
@media print {
  @page {size:A4 portrait;}
}
body{background-color:#87ceeb;}
div{width:80mm; height:50mm; background-color:green;}
img{width:78mm; height:48mm;}
p{margin-top:-1.5em; text-align:center;}
span{background-color:yellow; border:solid 1mm;}
</style>
<body>
<div><img src="hdtv320x180.jpg" alt="photo" /></p>
<p><span>date&amp;time</span></p>
</div>
</body>
</html>
```

Expected Output
3.3.7. Elements that may Affect Output Consistency

The following should be taken into consideration when creating XHTML-Print / CSS Print Profile contents intended for output from PrintEnhanced:1 compliant printers.

3.3.7.1. Float
Usage of the float element may affect output consistency. Its usage is not recommended in situations where consistency across printers is critical.

3.3.7.2. List-Style-Type / List-Style-Image
Usage of the List-style-type and/or List-style-image element may affect output consistency. Its usage is not recommended in situations where consistency across printers is critical.

3.3.7.3. Risky Margin Settings
Positioning content very near the edge of the page sheet may affect output consistency among printers, due to differences in the printer’s Non-Printable Areas. When output consistency is important, it is recommended that content be placed at a reasonable distance from the page sheet edge.

The use of page margins provides an easy way to ensure statically positioned content follows this guideline.

```css
@page {
    margin: 10mm;
}
```

Printers that follow the informative guidelines in Section 8.5 of [CSSPP] will by default establish a 10% page margin.

However, when the page margins are used for headers and footers, the Content Author must take care to ensure the header and footer contents are not placed within the Printer’s Non-Printable Area.

3.3.7.4. Relative Font Size
Usage of relative font size may affect output consistency. Its usage is not recommended in situations where consistency across printers is critical. Usage of absolute font size is recommended.
3.3.7.5. Character Encoding (UTF-16)

Contents creators should take into account that printers may not support UTF-16. Usage of UTF-8 encoding is recommended. For control points, PrintEnhanced:1 [PE1] defines a state variable `DocumentUTF16Supported` that can be used to determine UTF-16 support for the document format.

3.3.7.6. Forms

It is important when designing forms that will be printed to consider the differences between printed output and output displayed on a screen. The most likely usage scenario for printing forms is to create a record of a transaction. In this case, it is important to explicitly control layout so that the printed output shows which items were selected and any text the user may have entered.

Text areas should be sized large enough to show all critical contents. Setting overflow to 'visible' will help ensure all text that the user has entered is displayed on the printed page, but may lead to layout inconsistencies.

Select elements should be sized to allow all selected elements to be printed. Setting overflow to 'visible' will help ensure all selected items will be displayed on the printed page, but may lead to layout inconsistencies.
3.4. Sample Templates

This section provides XHTML-Print [XHTML-PRINT] and CSS Print Profile [CSSPP] contents source and corresponding expected output description for several simple photo-related layouts. The source is intended for use as examples and reference and as a basis for modification. Be aware that the templates will not necessarily produce the expected outputs when displayed on screen.

3.4.1. Basic Layout Concept

The templates are categorized by 1) the number of images per page and 2) with borders (white space) or without borders (full bleed). Bordered output is basically achieved by the image object(s) being positioned and sized within a positive (non-zero) page margin, whereas output without borders (full bleed) are achieved by positioning the object(s) on or outside the edge of a zero-margin page box, while sizing it equal to or greater than the page size.

Output with Borders Example

```xml
<?xml version="1.0" encoding="UTF-8"?><!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd"><html xmlns="http://www.w3.org/1999/xhtml"><head><style>@media print{
........
@page{...... margin:margin_Amm;}
........
img.basic1{width:image_size_B; height:image_size_C;}
}
</style></head><body>...
<img class="basic1" src="image1.jpg"/>
...
</body></html>
```

Object size should be within the page size

set page margin to value greater than zero to achieve white space

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
Output without Borders (Full Bleed) Example

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
  "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<style>
@media print{
  @page{margin:0;}
  div {position:absolute; top:E; left:F; }
  img.basic2{width:image_size_G; height:image_size_H;}
</style>
</head>
<body>
...<div>
<img class="basic2" src="image2.jpg"/>
</div>
...
</body>
</html>
```

Set page margin value to zero to achieve non borders

Object should be positioned on or outside the edge of the page.

Object size should be larger or equal to the page size. Only one of width or height should be specified to preserve the aspect ratio of the image.
The basic idea shown above can be applied to various layouts as shown below, which some of which will be shown in detail in the following sections.

3.4.2. Definition of Terms

ARp : Aspect ratio of paged media (height/width)

ARi : Aspect ratio of image data (pixels of a line / number of lines)

Note: Printer should assume square pixels in the image data, as typical image data consists of square pixels (i.e. pixels with an aspect ratio of 1:1).
3.4.3. One Image in One Page

3.4.3.1. With Borders (with White Space)

3.4.3.1.1. Borders on All Sides

Note

- The aspect ratio of the image is preserved.
- The image is not cropped.
- In order to achieve bordered output, @page margins of positive values should be specified.

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>One Image in One Page</title>
<style type="text/css">
@media print{
@page{size:A4 landscape; margin:5mm;} /*set margin to value greater than zero*/
body{padding:0mm; }
div{font: bold 36pt; text-align:center;}
img.frame11 {width:270mm; height:180mm;}
}
</style>
</head>
<body>
<div>Title<br /></div>
<div class="frame11" src="swimming.jpg" alt="swimming" /></div>
</body>
</html>
```

Output

![Image of a swimming pool]

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.4.3.1.2. **Aligning the Horizontal Edge of the Image to the Horizontal Edge of the Page Media Without Clipping**

**Note**

- The aspect ratio of the image is preserved.
- Paper size is A4 landscape.
- ARi < ARp
- ARi of the image is 4/3
- In order to achieve vertical zero margins, @page margins of zeros must be specified.

**Source**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<style type="text/css">
@media print {
    @page {size:A4 landscape; margin: 0mm;} /*set the horizontal margin to zero*/
}
body {padding:0mm;}
img {height:210mm;} /* Be careful to preserve image aspect ratio */
div {margin-left:8.5mm;} /* margin-left can be an arbitrary value */
</style>
</head>
<body>
<div><img src="wavingocn.jpg" alt="photo" /></div>
</body></html>
```

**Output**

8.5mm 280mm

210mm

297mm
### 3.4.3.1.3. **Case : Aligning the Vertical Edge of the Image to the Vertical Edge of the Page Media Without Clipping**

**Note**

- The aspect ratio of the image is preserved.
- Paper size is A4 landscape.
- ARi of the image is 16/9.
- ARi > ARp.
- In order to achieve vertical zero margins, @page margins of zeros must be specified.

**Source**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Aligning the Vertical Edge of the Image to the Vertical Edge of the Page Media Without Clipping</title>
<style type="text/css">
@media print {
    @page {size:A4 landscape; margin: 0mm;}/*set the vertical margin to zero*/
}
body {padding:0mm;}
img {width:297mm;} /* Be careful to preserve image aspect ratio */
div { padding-top:21.47mm;} /* padding-top can be an arbitrary value */
</style>
</head>
<body>
<div><img src="blueocn.jpg" alt="photo" /></div>
</body>
</html>
```

**Output**

```
297mm

21.47mm

167.06mm

210mm
```

© 2002-2005 Contributing Members of the UPnP Forum. All rights Reserved.
3.4.3.2. Full Bleed

3.4.3.2.1. Fitting/Full Bleed - Fitting the Vertical Edge of the Image on the Page Media (to Achieve Full Bleed)

Note

- The aspect ratio of the image is preserved
- ARi < ARp
- In order to achieve full bleed, @page margins of all zeros must be specified.
- In order to center the image, the image margin must be specified with a vertically negative value.
- In order to avoid the <div> container from flowing over to the next page instead of successfully cropping the portions vertically outside the page, absolute positioning should be applied to the <div> container.

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
 "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Fitting/Full Bleed – Fitting the Vertical Edge of the Image on the Page Media</title>
<style type="text/css">
@media print {
@page {size:A4 landscape; margin:0mm;}
body{padding:0mm; width:100%; height:100%;}
div {width:100%; height:100%; overflow: hidden;}
p {position:absolute; margin:0mm; width:100%; height:1em; bottom:0.5em;
font-size:36pt; color:white; text-align:center;}
img {width:100%; margin-top:-6.375mm;}
</style></head>
<body>
<div>
<img src="wavingocn.jpg" alt="photo" />
<p>date&amp;time</p>
</div>
</body>
</html>
```

Output

```
6.375mm

210mm

222.75mm

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
```
3.4.3.2.2. **Fitting/Full Bleed – Fitting the Horizontal Edge of the Image on the Page Media (to Achieve Full Bleed)**

**Note**

- The aspect ratio of the image is preserved
- ARi > ARp
- Text (Date) is overlaid above the image
- In order to achieve full bleed, @page margins of all zeros must be specified.
- In order to center the image, the image margin must be specified with a horizontally negative value.
- In order to avoid the <div> container from flowing over to the next page, the height of the image (<img>) should not exceed the page height.

**Source**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Fitting/Full Bleed, Fitting the Horizontal Edge of the Image</title>
<style type="text/css">
@media print {
@page {size:A4 landscape; margin:0mm;}
body {padding:0mm; height:100%; width:100%;}
img {width:373.3mm; height:210mm;}
div {margin-left:-38.16mm;}
p {margin-top:-1.5em; margin-left:238.16mm; font-size:3em; color:white;}
</style>
</head>
<body>
<div>
<img src="blueocn.jpg" alt="photo" />
<p>Jan. 1 2004</p>
</div>
</body>
</html>
```

**Output**

```
38.16mm
297mm
```

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.4.3.2.3. **Fitting/Full Bleed – Cropping the Horizontal Edges of the Image on the Page Media**
*(to Achieve Full Bleed)*

**Note**

- The aspect ratio of the image is preserved
- ARi > ARp
- Text (Date) is overlaid above the image
- In order to achieve full bleed, @page margins of all zeros must be specified.
- In order to center the image, the image margin must be specified with a horizontally negative value.
- In order to crop both sides of the image, “overflow: hidden” must be set to the div container.

**Source**

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Fitting/Full Bleed, Cropping the Horizontal Edge of the Image</title>
<style type="text/css">
@media print {
@page {size:A4 landscape; margin:0mm;}
body {padding:0mm; height:100%; width:100%;}
ing { height:100%; margin-left:-38.16mm;}
div {height:100%; width:100%; overflow: hidden;}
p {margin-top:-1.5em; margin-left:200mm; font-size:3em; color:white;}
</style>
</head>
<body>
<div>
<img src="blueocn.jpg" alt="photo" />
<p>Jan. 1 2004</p>
</div>
</body>
</html>
```
Output

38.16mm  297mm

210mm

373.3mm
3.4.4. Two Images in One Page

3.4.4.1. With Borders (with White Space)

Note

- The aspect ratios of the images are preserved.
- There is white area around each image.
- Paper size is A4 portrait (210mm x 297mm).
- Page margin is 5mm (Page box size is 200mm x 287mm).
- ARi of the upper image is 4/3.
- ARi of the lower image is 16/9.
- The images are not cropped.
- The source code example uses absolute positioning.
- It is possible to use static positioning with modification of the source structure instead of absolute positioning.
- In order to achieve bordered output, @page margins of positive values should be specified.

Source

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Two Images in One Page With Borders</title>
<style type="text/css">
@media print {
@page { size: A4 portrait; margin: 5mm; } /* don’t use default margin */
}
* { margin: 0mm; } /* Disabled default style sheet */
img.frame21 { position: absolute; }
div.frame21 { position: absolute; left: 10mm; width: 180mm; height: 120mm; }
/* div box position from the left-edge of page box */
#div1 { top: 15mm; } /* Upper image box position from top of page box */
#div2 { top: 150mm; } /* Bottom image box position from top of page box */
#img1 { left: 10mm; top: 0mm; height: 120mm; }
/* upper image top & left margin from div box */
#img2 { left: 0mm; top: 9.375mm; width: 180mm; }
/* bottom image top & left margin from div box */
</style>
</head>
<body>
<div class="frame21" id="div1">
<img class="frame21" id="img1" src="swimming.jpg" alt="swimming" />
</div>
</body>
</html>
```
Output

```
</div>
<div class="frame21" id="div2">
<img class="frame21" id="img2" src="bird.jpg" alt="bird" />
</div>
</body>
</html>
```

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.4.4.2. Full Bleed

Note

- The aspect ratios of the images are preserved.
- Paper size is A4 portrait.
- Page margin is 0 (Page box size is 210mm x 297mm).
- ARi of the upper image is 16/9. (Right and left portions of the image are cropped)
- ARi of the lower image is 4/3. (Top and bottom portions of the image are cropped)
- Each image is rendered in half size of a paged media.
- Each image is cropped if ARi of the image was not equal to div.
- In order to avoid the <div> container from flowing over to the next page instead of successfully cropping the portions vertically outside the page, absolute positioning should be applied to the <div> container.
- In order to achieve full bleed output, @page margins of all zeros must be specified.
- In order to crop the images, “overflow: hidden” should be use to crop outside of “div” box.

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Two Images in One Page Full Bleed</title>
<style type="text/css">
@media print {
@page { size: A4 portrait; margin: 0mm;}
}
* { margin: 0mm; } /* Disabled default style sheet */
div.borderless21 { position: absolute; left: 0mm; height: 148.5mm; width: 210mm; overflow: hidden;}
/* image box is half of page box, and out of image box is hidden */
img.borderless21 { position: absolute; }
#div1 { top: 0mm; } /* upper image box top offset from page box */
#div2 { top: 148.5mm; } /* lower image box top offset from page box */
#img1 { left: -27mm; top: 0mm; height: 148.5mm; } /* left negative margin */
#img2 { left: 0mm; top: -4.5mm; width: 210mm; } /* top negative margin */
</style>
</head>
<body>
<div class="borderless21" id="div1">
<img class="borderless21" id="img1" src="bird.jpg" alt="bird" />
</div>
</body>
</html>
```
Output

Output diagram:

- div box of img1.
- img box of img1.
- Non-viewing area of img1
- Non-viewing area of img2.

Diagram labels:
- img1 left
- div2 top
- img1 height
- / div height
- img2 top
- div height
- img2 width
- / div width
- paper size
- div box of img2.
- img box of img2.
3.4.5. Four Images in One Page

3.4.5.1. With Borders (with White Space)

Note

- The aspect ratios of the images are preserved.
- There is white area around each image.
- Paper size is A4 landscape.
- Page margin is 5mm (Page box size is 287mm x 200mm).
- The images are not cropped to bleed fully in 4 predetermined areas on the paged media.
- ARi of the upper-left and lower-right images are 16/9.
- ARi of the upper-right and lower-left images are 4/3.

In order to avoid the <div> container from flowing over to the next page instead of successfully cropping the portions vertically outside the page, absolute positioning should be applied to the <div> container.

In order to achieve bordered output, @page margins of positive values should be specified.

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Four Images in One Page with Borders</title>
<style type="text/css">
@media print {
  @page { size: A4 landscape; margin: 5mm; } /* don't use default margin */
  }
  *
  /* Disabled default style sheet */
  */
  div.frame41 { position: absolute; width: 135mm; height: 90mm; }
  overflow:hidden;}
  /* all div box size & outside img from div box is hidden */
  #div1 { top: 5mm; left: 4mm; } /* top-left image position (img1) */
  #div2 { top: 5mm; left: 148mm; } /* top-right image position (img2) */
  #div3 { top: 105mm; left: 4mm; } /* bottom-left image position (img3) */
  #div4 { top: 105mm; left: 148mm; } /* bottom-right image position (img4) */
  img.frame41 { position: absolute; }
  #img1 { left: -12.5mm; top: 0mm; height: 90mm; }
  /* img1 left negative margin from div1 box & height (width is calculated ) */
  */
  #img2 { left: 0mm; top: -5.625mm; width: 135mm; }
  /* img2 top negative margin from div2 box & width (height is calculated ) */
</style>
</head>
<title>Four Images in One Page with Borders</title>
<style type="text/css">
@media print {
  @page { size: A4 landscape; margin: 5mm; } /* don't use default margin */
  }
  *
  /* Disabled default style sheet */
  */
  div.frame41 { position: absolute; width: 135mm; height: 90mm; }
  overflow:hidden;}
  /* all div box size & outside img from div box is hidden */
  #div1 { top: 5mm; left: 4mm; } /* top-left image position (img1) */
  #div2 { top: 5mm; left: 148mm; } /* top-right image position (img2) */
  #div3 { top: 105mm; left: 4mm; } /* bottom-left image position (img3) */
  #div4 { top: 105mm; left: 148mm; } /* bottom-right image position (img4) */
  img.frame41 { position: absolute; }
  #img1 { left: -12.5mm; top: 0mm; height: 90mm; }
  /* img1 left negative margin from div1 box & height (width is calculated ) */
  */
  #img2 { left: 0mm; top: -5.625mm; width: 135mm; }
  /* img2 top negative margin from div2 box & width (height is calculated ) */
</style>
</head>
</html>
```
Output
3.4.5.2. Full Bleed

Note

- The aspect ratios of the images are preserved.
- Paper size is A4 landscape.
- Page margin is 0 (Page box size is 297mm x 210mm).
- Each image is rendered in quarter size of a paged media.
- Each image is cropped if ARi of the image was not equal to ARp.
- ARi of the upper-left and lower-right images is 16/9. (Right and left portions of the image are cropped)
- ARi of the upper-right and lower-left images is 4/3. (Top and bottom portions of the image are cropped)
- Date/Time is placed on the image.
- In order to avoid the <div> container from flowing over to the next page instead of successfully cropping the portions vertically outside the page, absolute positioning should be applied to the <div> container.
- In order to achieve full bleed output, @page margins of all zeros must be specified.
- In order to crop the images, “overflow: hidden” should be use to crop outside of “div” box.

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
 "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Four Images in One Page Full Bleed</title>
<style type="text/css">
@media print {
@page { size: A4 landscape; margin: 0mm;} /* fullbleed */
}

div.borderless41 { position: absolute; width: 148.5mm; height: 105mm; 
overflow: hidden; } /* all div box is 1/4 paper-size(A4) & outside of div box is hidden */
#div1 { top: 0mm; left: 0mm; } /* top-left image position (img1) */
#div2 { top: 0mm; left: 148.5mm; } /* top-right image position (img2) */
#div3 { top: 105mm; left: 0mm; } /* bottom-left image position (img3) */
#div4 { top: 105mm; left: 148.5mm; } /* bottom-right image position (img4) */
p.borderless41 { position: absolute; bottom: 0mm; left: 0mm; width: 148.5mm; height: 12mm; font-size: 24pt; text-align: center; } /* date/time text box */

img.borderless41 { position: absolute; }

#img1 { left: -19.05mm; top: 0mm; height: 105mm; }
```

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
/* img1 left negative margin from div1 box & height (width is calculated ) */
\#img2 { left: 0mm; top: -3.1875mm; width: 148.5mm; }
/* img2 top negative margin from div2 box & width (height is calculated ) */
\#img3 { left: 0mm; top: -3.1875mm; width: 148.5mm; }
/* img3 top negative margin from div3 box & width (height is calculated ) */
\#img4 { left: -19.05mm; top: 0mm; height: 105mm; }
/* img1 left negative margin from div4 box & height (width is calculated ) */
</style>
</head>
<body>
<div class="borderless41" id="div1">
  <img class="borderless41" id="img1" src="img1.jpg" alt="img1" />
  <p class="borderless41">2004/09/14</p>
</div>
<div class="borderless41" id="div2">
  <img class="borderless41" id="img2" src="img2.jpg" alt="img2" />
  <p class="borderless41">2004/09/15</p>
</div>
<div class="borderless41" id="div3">
  <img class="borderless41" id="img3" src="img3.jpg" alt="img3" />
  <p class="borderless41">2004/09/16</p>
</div>
<div class="borderless41" id="div4">
  <img class="borderless41" id="img4" src="img4.jpg" alt="img4" />
  <p class="borderless41">2004/09/14</p>
</div>
</body>
</html>
Output

- img1 left
- div width
- div height
- img2 top
- img3 top
- img4 top
- div left
- date/time text box
- paper size
- img box
- div box
- Non-viewing area of img

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
3.4.6. Index Printing

3.4.6.1. Simple Index Page Using Tables

Note

- none

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Simple Index Page Using Tables</title>
<style type="text/css">
@media print {
  @page {size:A4 portrait; margin:5mm;}
  body {padding:0mm;}
  table {font-size:12pt; text-align:center;}
  caption {font-size:24pt;}
  img {width:40mm; height:30mm;}
}
</style>
</head>
<body>
<table>
<caption>index print</caption>
<tr>
<td><img src="beach02.jpg" alt="image 1-1" /><br />November 17</td>
<td><img src="restaurant.jpg" alt="image 1-2" /><br />October 11</td>
<td><img src="beach01.jpg" alt="image 1-3" /><br />November 13</td>
<td><img src="swimming.jpg" alt="image 1-4" /><br />November 13</td>
</tr>
<tr>
<td><img src="restaurant.jpg" alt="image 2-1" /><br />October 11</td>
<td><img src="beach01.jpg" alt="image 2-2" /><br />December 11</td>
<td><img src="swimming.jpg" alt="image 2-3" /><br />November 13</td>
<td><img src="beach02.jpg" alt="image 2-4" /><br />November 13</td>
</tr>
</table>
</body>
</html>
```
Output

index print
3.4.6.1.2. Index page Without Using Tables

Note

- none

Source

```xml
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML-Print 1.0//EN" "http://www.w3.org/MarkUp/DTD/xhtml-print10.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Simple Index Page Without Using Tables</title>
<style type="text/css">
@media print {
  @page {size:A4 portrait; margin-top:5mm; margin-left:16.5mm; margin-right:5mm; margin-bottom:5mm; }

  body {padding:0mm;}

  h1{width:180mm; margin-top:0; font:bold 16pt serif; text-align:center; color:#ff9933;}

  p {height:47.5mm; color:blue; margin:0; padding:1mm; width:40mm; border:solid 0.25mm; text-align:center; font:11.5pt serif; background:#ffffcc;}

  p.col1 {margin-top:2.5mm; margin-left:0mm;}

  p.col2 {margin-top:-50mm; margin-left:45mm;}

  p.col3 {margin-top:-50mm; margin-left:90mm;}

  p.col4 {margin-top:-50mm; margin-left:135mm;}

  img {width:40mm;}

</style>
</head>
<body>
<h1>index print</h1>

<p class="col1"><img src="beach02.jpg" alt="image 1-1" title="May 7" /><br />November 17</p>
<p class="col2"><img src="restaurant.jpg" alt="image 1-2" title="October 11" /><br />October 11</p>
<p class="col3"><img src="beach01.jpg" alt="image 1-3" title="November 13" /><br />November 13</p>
<p class="col4"><img src="swimming.jpg" alt="image 1-4" title="November 13" /><br />November 13</p>

<p class="col1"><img src="restaurant.jpg" alt="image 2-1" title="May 14" /><br />October 11</p>
<p class="col2"><img src="beach01.jpg" alt="image 2-2" title="December 11" /><br />December 11</p>
<p class="col3"><img src="swimming.jpg" alt="image 2-3" title="November 13" /><br />November 13</p>
<p class="col4"><img src="beach02.jpg" alt="image 2-4" title="November 13" /><br />November 13</p>
</body>
</html>
```

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.
Output

index print

November 17

October 11

November 18

November 13

November 17

October 11

December 11

November 18

November 13

November 13

© 2002-2005 Contributing Members of the UPnP™ Forum. All rights Reserved.