KDDI's Position Paper for Second W3C Workshop on Electronic Books and the Open Web Platform
4 June 2013, Tokyo, Japan

Yukinori Yamada KDDI CORPORATION
Koichi Takagi KDDI CORPORATION
Satoru Takagi KDDI CORPORATION
booklista Co.,Ltd

1. Introduction

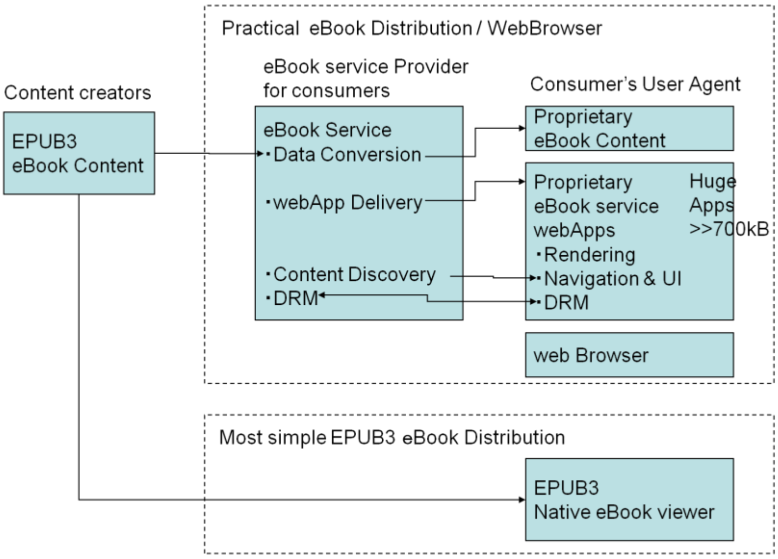
KDDI is a Japanese network operator mainly on mobile, fixed line and cable television. And KDDI provides an eBook service for consumers with BookLista in Japan. Booklista is an eBook distributors aiming at content selling service including computerizing and maintaining book content, billing, and promotion.

KDDI's perspective for this Workshop is to find standardization issues to promote more efficient construction of eBook services using a Web browser.

2. Characteristics of the KDDI's eBook service

- Main part of content: Comics
- Web browser-based service on iOS
- Hybrid app-based service on Android

3. Relations with HTML and EPUB in the current KDDI's eBook service



As this figure shows, KDDI's eBook service adopts EPUB format to receive the delivery of contents. And contents offered to consumers follow proprietary data format. It is displayed on the web browser by huge webApps. Not only contents rendering, user interfaces (flipping pages and content discovery) and the payment management functions account for a considerable part of this webApp.

Therefore, the webApp architecture for this eBook service is strongly influenced by these functions except for rendering eBook content.

On the other hand, there is out of scope of EPUB format other than expression for rendering. Therefore, most part of the architecture for this eBook service is different from EPUB format. As a result, webApp for the KDDI's eBook service depends on the proprietary data.

4. Issues and Findings

There are some issues and findings provided by the setup of our eBook service.

4.1. EPUB

- Though we build webApp including the function to render original EPUB data that have considerably common specifications to HTML, the code size is considerable huge. One of the reasons is adoption of the streaming architecture for copyright protection. For instance, the cord size of webApp is more than 700kB that supports fixed-layout only.
- For images in EPUB content, it is desirable that the specifications can declare the value of maximum resolutions in meta information.
- Fixed-layout cannot fill a gap of the expression by the difference of the publisher.
    - Appearance of a standard and open authoring tool is expected.
    - Establishment of the framework which can easily extract layout information is desired.

4.2. Web Browser

- In the case of streaming, in particular, an API to get the network connection state is especially desired for user experience. > DAP/SysApps. WG
    - connection type (Wi-Fi? cellular? ...)
- More minute specifications about the securing of capacity of the Web storage are expected. This is because we had trouble with filling a gap between browsers when we used a slightly large storage.
- In the case of streaming eBook distribution, the processing time for DRM decoding by javascript is not negligible.
    - It can be barely solved by utilizing a background process, but simpler elucidation is expected.
- Construction of huge webApp almost equal to native App was required for eBook service.
    - For stable operation, careful resource management was considerably required.
- Establishment of the standards about the control of the tab of the Web browsers is desirable.
- Introduction of the richer concept of "page" of the book to HTML is desirable.
    - Simple and declarative expression of the rich transition/animation between pages