

Code4Lib JAPAN

URL の固定化(パーマリンク)の必要性の提言

2011 年 6 月 23 日

2011 年 7 月 4 日 一部字句修正

Code4Lib JAPAN 提言事業

URL: <http://www.code4lib.jp/contribution/>

インターネット上での存在がデータの存在の有無そのものの扱いにもなっている現在、図書館もそれぞれの館が持つリソースをメタデータだけでもネット上に公開することの重要性が認識されてきている。しかし、それらコンテンツの URL については、認識が低いようである。

URL とは(注 1)、プロトコルで、<http://>で始まるものはインターネット上でのリソースの存在を URL スキームで表示したものである。また、URL は、リソースの名称であり、ブラウザのアドレス欄に表示されるものである。

各々の URL は、インターネット上で唯一の URL でなくてはならないし、極力永続性を考えて作られることが望まれるものである。

なぜ固定化が求められているのか。それは固定化していないと、この資料はこの URL であると特定しておけないため、再利用、引用が難しくなる。具体的には、リンク切れが生じたり、リンク先が別資料になる可能性が生じ、データへの信頼が問題となってしまうため、固定化は情報提供機関として重要な課題となっている。

図書館の対応

1. 個々の書誌に対してその館で永続して持つ書誌 ID (以下、ユニーク ID) を持つように仕様に入れる。

書誌 ID については、ベンダにより、各図書館の希望に沿ったルールで付与できるものもあるが、システムで決められてしまっているものもある。そのような書誌 ID は、システムのリプレイスや、ハードの入れ替えなどでデータ移行をしたときに、別の書誌 ID が振られることになる。

それを避けるため、いわゆるローカル書誌タグ(図書館で管理できる MARC 上にはないタグ番号)に館のユニーク ID を書誌生成時に自動で重複なく割り振られるようにすることが必要である。このユニーク ID は、検索対象にし、システム更新時には移行対象にする。

2. ユーザに見せる URL と内部的 URL を分けられるようにする。

個々の書誌データは、ユニーク ID を使用した URL で指摘できるようにする。

3.プログラミング言語や実装依存の URL や拡張子は利用しないで、シンプルな URL にする。

プログラミング言語や実装依存の URL の例)

`http://www.example.jp/cgi-bin/***.cgi`

`http://www.example.jp/***.asp`

`http://www.example.jp/servlet/***.jsp`

URL 固定化のサンプル例

a) 拡張子で表現を指定する方法。

`http://www.example.jp/id/html/jp/`

`http://www.example.jp/id/json/jp/`

`http://www.example.jp/id/xml/jp/`

`http://www.example.jp/id/atom/jp/`

(注 2) この例の場合の html、json、xml、atom は、フォーマットを指示しており、言語や実装依存ではない。

b) 日付を使う方法。

`http://www.example.jp/news/2011/05/25/html/`

c) 検索式もシステム依存から抜け出すことが望まれる。

これは API 仕様時に提言予定。

(注 1) URL について

URL (Uniform Resource Locator) を包括する概念が、URI (Uniform Resource Identifier) であり、インターネット上のリソースの所在を特定する表記方法です。