KTYYゼミ資料: 論文の今後について

川合研 M1 伊戸川 暁

1998 年 10 月 21 日

1　経過報告

9月4日 論文投稿。(最終的に提出した論文を同時に配布しました。ただし今回はまた前提からやり直すつもりなので、前半だけ見てもらえれば結構です)

10月8日 シンポジウムへの採用ならびに論文誌への採録が決定される。しかし多くの修正が必要そう。

査読者のコメントは以下の通り。

> ・まだ完全に完結はしていないようですが、その方向性はシンポジウム参加者の興味を十分引くものがあるし、内容もまとまっていると思います。
>
> ・論文誌への掲載は、詳しさの点で少しもの足りないが、情報学シンポの発表としては面白い。但し、実際のプレゼンテーションでは、システム・イメージ等の具体的な例示が望まれる。この論文のままでは、単なる構想提案と捉えられてしまいそうだ。
>
> ・電子的な記録の長期保存は重要な問題であるが，ハードウェアには触れず「ソフトウェア的側面」のみについて論じるとなれば，ここで対象とされる「オンライン電子文書」の定義と特性と，長期保管に必要な要件をよく検討する必要があろう。ここにあげられている問題点，(1) 媒体の信頼性，(2) リンクの脆さ，(3) 価値判断，(4) メタデータのための枠組の不十分さ，はレベルの異なるものであり，それ自体疑問があるが，さらに本論文では(1) と (3) は全く扱われていない。そのために，提案されている長期保管のためだけの「POT」の妥当性に疑問が生じる。　少なくとも，タイトルは内容に合うように変更し，上記の点について再検討して頂きたい。

11月27日 最終稿締切日。これがcamera-readyな原稿の締切なので、査読者とのやりとりはこれより前に済ませていなければならない。

2 問題点群の見直し

「挙げた問題点のレベルが揃っていない」とまで指摘されてしまったので、少し問題点群のダメ出しをしてみた。

2.1 議論の前提に関する補足

前に出した論文ではあまりきちんと説明していなかったが、システム及びそれが対象とする文書のイメージは、

> 「そのままではその場で流通・消費のみされて残らなくなってしまうような情報を、体系的にフローからすくい上げて保存する。なお、情報の価値は、すくい上げるもとの情報フローの関数であると仮定する」

というものである。

2.2 問題点(1)(媒体の信頼性)関係

本質的な変更ではないが、「媒体変換」という術語を論文に導入しようと思う。

ここに、広義の媒体変換(data migration)とは、情報を他の媒体に移転することで、さらに細かくは、バイト列のレベルで情報を保持しつつ移転するもの(replication)と、論理的に同一ならばバイト列のレベルでは移転前後の同一性を問わないもの(狭義のmigration)とに分けられる。[1]

効率的な媒体変換のためには、媒体の特性を記述したメタデータをシステムが持っていなければならない。ではそれはどこから取得するかという問題があるが、

- テスト用ソフトを付け、各自でmediaの丈夫さを試してもらう[2]
- 調査機関の結果を用いる
- メーカーに情報を提供してもらう

---

[1] 後者の場合、変換前後の情報の同一性の保証が難しくなる(単純にハッシュ関数の結果で判断することができなくなる)が、XMLの規格の一部として策定が進められつつあるDOM(Document Object Model)を応用すれば、文書のバイト列ではなく、文書情報をobject化したもので同一性を比較できるかもしれない。

[2] 書込/消去を激しく繰り返せばテストができるような気がするが、どうなのだろうか?

などが考えられるが、いずれにせよ、ファイル形式と同様 (次節参照)、使用するデバイスに関する情報をどこかに登録し、そこで authorize されたデバイスのみを使うようにするということになろう。

「媒体に記録されている内容は、やはり OS のファイルシステムに依存している」という問題も存在するが、ファイルシステムもまたデバイス同様登録制をしくことにする。

2.3　問題点 (2)(リンクの脆さ) について

「文書の内容にのみ依存し、文書の所在地からは独立した名前を与える」という方針は変更しないが、題を 「命名体系の不安定さ」などに改め、広くファイル一般の命名システムのあり方などと絡めて議論するようにする。

最近、「記号列よりも自然言語に基づいた命名システムの方が経時的変化に強いのではないか?」という仮説を立ててみている (未検証)。

2.4　問題点 (3)(価値判断) について

結局、システムが扱う文書の価値判断は予めなされているものとして議論を進めようとしていることが明らかなので、問題点として挙げるのは止め、議論の前提に含めることとした。

2.5　問題点 (4)(メタデータの枠組の不十分さ) について

改良案募集中。

2.6　追加する問題点: ソフトウェアの陳腐化

これを電子情報の長期保存に関わる問題点として新しく提起する。

ソフトウェア技術も年々進歩しているので、ファイル自体が存在していても、それを読むためのソフトウェアがないのでファイルが読めないという事態は往々にして起こる。そこで、以下のような方法によって、この問題を防ぐことを考える。

- まず、英文 plain text を boot strap として仮定する。
- Plain text 以外の format に関しては、それを見た人が容易に viewer を実装できるような仕様書を用意しないと登録できないようにする。[3]仕様書も、その保存の方法は一般の文書と変わるところはない。

要するに登録制にするということですね。

---

[3] 本システム自体の仕様も同様の扱いとする。

3 再び仕様を考える — 実装方針

- 仕様は全て open とする。
  私企業の製品でも、最近この論文と同じような目的のものを少なからず見かけるが、仕様が完全に透明であるものは (無知のせいか) 未だ知らない。
  - システムが扱うファイルの種類やデバイスに関する情報は、その仕様が完全に公開されているものに限られ、またその仕様は、システムの保存にかかる文書として登録されていなければならない。(即ち、いつでも仕様書から viewer が実装できるようにしなければならない)
  - メタデータの各要素などの記述方法は、現行の Internet の慣用を尊重する。Server 間の転送は勿論のこと、server 内についても、極力外部用書式からの変換を行わないで済むようにすることが望ましい。
- 等々……。

まだ「システムイメージの具体的例示」には辿りついていません。やれやれ^{TM}。

4 近況

- 実装されるであろうものは分散データベースの一種と言えるので、少しCORBA のお勉強中 (でもさっぱり分からない……)。
- 命名システムに関するヒントを捜索中。