株式会社 ITS MORE

2020年4月始動

日付 : 2020.04.23
23:26:58 +09'00'

2020年4月23日 投稿者: YSATO@DELEGATE.ORG

data schemeは健在だった

Macに触れたのがきっかけで、つい昔の事を思い出してしまう。URLと言う表現形式の成長期に提案された dataと言うスキームもそのひとつ。

URLは http や ftp と言ったプロトコル名（つまりアクセス手段）と、//ホスト名/パス名、という「リソースのロケーション」つまりアクセス先のアドレスを表す表記法・手段（scheme）だ。その中のひとつとして、リソースのデータ自体をURLに表記する方法として提案されたのが「data」だった。似ていて対極にあるものに、抽象的にメールアドレスを表記する mailto （転送手段も規定していない）があり、現在広く使われている。

CPUの命令アドレスフィールドに即値（immediate value）を持たせられるのは当たり前だし、変数と定数とかポインタなど、その他の類想からも、コンピュータ分野で育てば当然考えられるべき事なのだが、当時はなるほど（膝をポン）と思った記憶がある。
一般的にも「名は体を表す」と言うが、体自体を名前にしてしまうジョークや商品名は昔からある。data scheme はまさにそれに相当する。

以下に具体例を示す。上がデータ（gifイメージ）のナマ表現で、下がそれをブラウザが解釈して表現した結果だ（画像は、data scheme 提案者のLarry Mainster 氏のポートレート）。

<IMG SRC="

AAAC8IyPqcvt3wCcDkiLc7C0qwyGHhSWpjQu5yqmCYsapyuvUUlvONmOZtfzgFz

ByTB10QgxOR0TqBQejhRNzOfkVJ+5YiUqrXF5Y5lKh/DeuNcP5yLWGsEbtLiOSp

a/TPg7JpJHxyendzWTBfX0cxOnKPjgBzi4diinWGdkF8kjdfnycQZXZeYGejmJl

ZeGl9i2icVqaNVailT6F5iJ90m6mvuTS4OK05M0vDk0Q4XUtwvKOzrcd3iq9uis

F81M1OIcR7lEewwcLp7tuNNkM3uNna3F2JQFo97Vriy/Xl4/f1cf5VWzXyym7PH hhx4dbgYKAAA7" ALT="Larry">

ここにはきっちりと、当時確立していたMIMEのデータ型名と、base64エンコーディングが、やはりWWWの初期に確立していたHTMLのタグIMG SRCの中に表現されている。だから、それら極めて基本的な標準仕様に準拠し、gifを表示できるブラウザなら、必ず表示できる表記だった（はずだったが、多くのブラウザの実装ではそうではなかった…）。

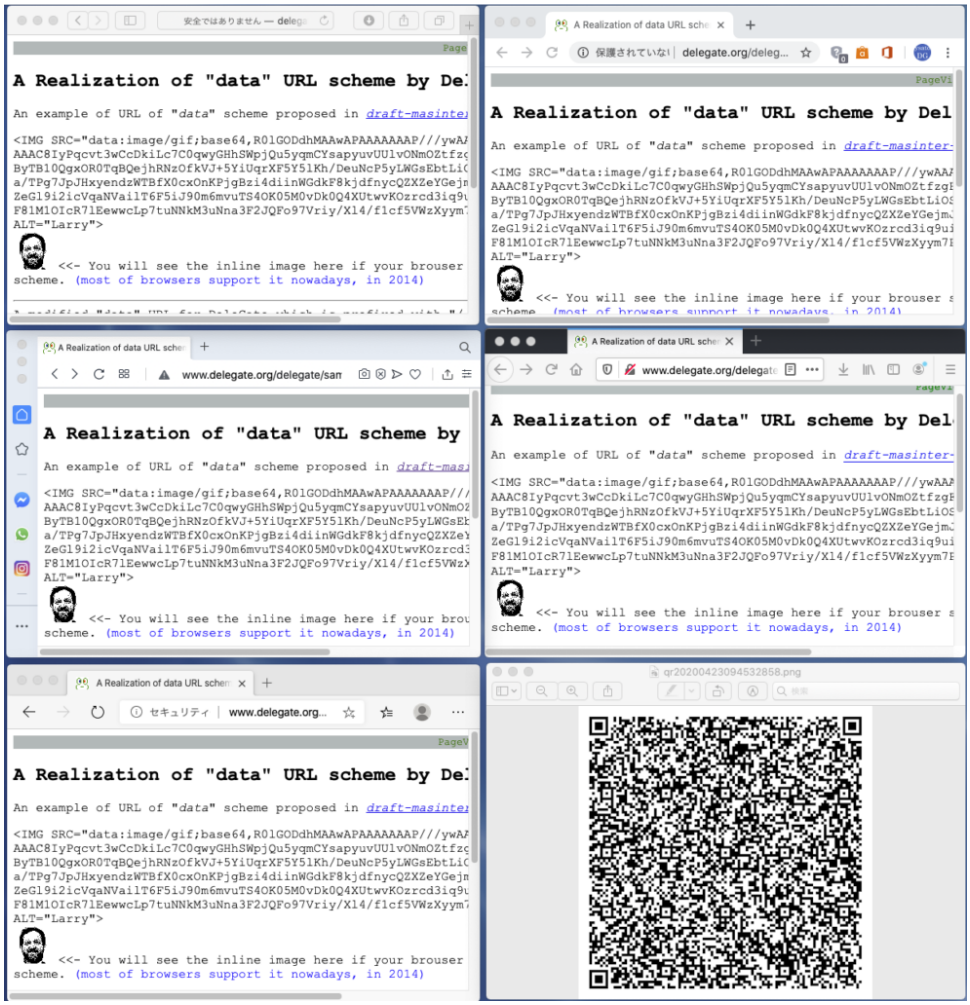
data scheme には実際、特に当時の未成熟なブラウザの状況もあり、色々な応用が期待された。data:表現が提案されたのは1997ねんで、翌1998年に正式にRFC2397となった。私はこれを使って、ブラウザによってサポートされていない文字や画像を埋め込む事を考えた。当時書いたデモ用のページが今も残っている（というか、動態保存している笑）
URL: http://www.delegate.org/delegate/sample/data-url.shtml ↓

data scheme を代理で変換する試み（1997年）

これは、ブラウザローカルに解釈できる事がメリットのはずの data:表現を実装していないブラウザのために、リモートの http:サーバでそれを補ってやろうという、分かってはいたが本末転倒の試みであった（実際、ウケなかった）。そして、data がきちんと実装されるより前に、ブラウザの多言語対応や各種イメージ型の表示能力が成熟し、当時私が考えた応用の目的は失われた。

その後の2014年の追記にあるように、やがてほとんどのブラウザがこれを実装した。ただ、私の偏見から来る記憶違いかもしれないが、MSのIEの対応は遅かったように思う。

以下は2020年4月現在の、５大メジャーブラウザでの data scheme の表示。いずれもちゃんと表示できている。



各ブラウザでのdata:表現の実装状況（2014年）

右下は昨今流行りのQRコードでこの data を表現したもの。残念ながら、iPhoneのカメラアプリは、これを認識してくれなかった（笑）

ここまで長く、これを書いたのは、data scheme という歴史に埋もれ忘れ去られた表記を、今も全ての主要ブラウザが解釈できるという事実に基づく、新しい応用があると考えているからだ。応用するのは、良い人かも知れないし、悪い人かも知れないが…

たとえば、文字にしてもUnicodeでは足りないケースも多いはずだ。そういうニッチは今後も完全に無くなる事は無い。ニッチをグローバルなインフラからサポートするのはコストに合わない。

インターネットは100倍早くなったかも知れないが、一方でユーザは100倍短気になった。それに、いくらネットワークの片道スループットが上がったとしても、往復レスポンスは光の速さを超えられない。途中の停車駅だってそうは減らせないだろう。実際、ギガビット・インターネットのRTTは当時の10倍程度しか高速化していない、今もミリ秒の単位のようだ。

ウェブには１ページを表示するのに数十回のHTTPリクエストが発生するのはざらにある。細々としたイメージデータを多数取得するためである場合が多い。それらは、転送時

にdata:にして埋め込んでやれば、一発転送で終わる。（いや、それをDeleGateにやらしてもらう必要は無いけど 笑）。同じデータを繰り返し送るとい無駄は発生するが、データの素性を知っていれば自動的・機械的なトレードオフはできる。それは長期的に保存するにしても、埋め込み型が有用なケースも多いだろう。

data scheme を IMG SRC 以外でも使えたとしたら、何が（何か）起こるか。応用は、全てをひとつのHTMLファイルに押し込む手段としてだけでは無い。MIME multipart 型と同様、古びた骨董の中に新しい可能性が埋もれている可能性はある。普通に普及しているアーカイブ形式をファイルシステムとして解釈する手法との優劣かなと思う。形式的には格好良くは無い。XMLが、ASN.1 があると言う人もいるかも知れない。しかし実現上の強みは、すでに data scheme は（長い時間をかけて）実装普及が完了していて準備OKであり、シームレスな移行かトランスペアレントな高速化が可能な事だ。

…と、つい昔のエキサイティングな時代を思い出してエキサイトしてしまった…

インターネットに破れて死んだOSIが電子証明書の中に残して行った形見のような X.509 を思いつつ。

2020-0423 sato@izmoh

data-schemeは健在だった—–株式会社-ITS-more　　**ダウンロード**

2020-0423 追記
Wikipedia に比較的最近（2018年）に更新された data scheme のページがあった。さすが、素晴らしい（個人的にWikipediaには献金しているが、会社で寄付したら控除してくれないかな 笑）。それによると、インラインイメージの埋め込みによって転送効率を上げられる、一方キャッシュの有効性は落ちる、と言う認識は元々共通のもののようだ。また元のRFCから指摘されているように、各種セキュリティ上の問題ははらんでいる。MSでの対応に関しては「IEとEdgeでは未実装の部分がある」ともあるので、私の偏見ではなかったようだ。

data:表現の用途としてはインラインイメージ（IMG SRC）以外にも、各種スクリプトのデータのソースとして利用する例が挙げられている。メールに画像を表示する利用例が挙げられているが、現状、サムネールなどを送るのに利用されていたりするのだろうか？

このWikiページを斜めに読んだが、次の言及に目が止まった「他にサポートされないであろうと思われる要素に、電子メールクライアントのマルチパート形式やmessage/rfc822などがある」ピクピク！

え？今日、電子メール（MIME)で添付ファイルを送るのには実際、マルチパート形式が使われているし、メール読み書きツールはそれを理解するから成り立っている。なので、これはウェブ用の閲覧ツール（ブラウザ）における状況を指摘しているのだろう。data scheme に関する記事の中での言及なのだから、data:multipart/mixed,–xxx… と言った表現の事を指しているのだろう…

ひとつ前の文から読み直すと「ブラウザなど、多くの処理環境ではメタデータ、データ圧縮、コンテントネゴシエーションのような複雑な処理はサポートしないであろう。他にサポートされないであろうと思われる要素に、電子メールクライアントのマルチパート形式やmessage/rfc822などがある。」となっていた。

HTTPサーバとクライアントのやりとりが無い環境での使用も想定されるので、ネゴシエーションは無いだろう。と言うかそれは、ブラウザがローカルなHTMLファイルを開く場合でも同じ事だ。編者が言いたいのは、HTTPネゴシエーションでクライアントが受け取りを拒否したデータ型のdata:を、サーバが削除して返すことはないだろう、と言う意味だろうか。

また、URLの長さの制約からして、その他に言及されている機能がdata:表現においてサポートされないのも仕方が無い。ただ、複数のdata:と言うかリソースを連結してひとつのデータとしてみなす、分割パートのような一般的な表現方法が規定されていれば、そうとは限らない。

あるいは連結とは逆に、一つのデータの中の一部を切り出す、たとえばあるデータの中での開始オフセットとサイズとか、HTMLやXMLの中の名前のついたエレメントを参照するとか。これがあれば、何もdata:に詰め込む必要はない。DeleGateのマニュアルはそれを試行したものだ。一つのHTMLファイルの部分部分を、単独のHTMLのようにも表示する事ができる。

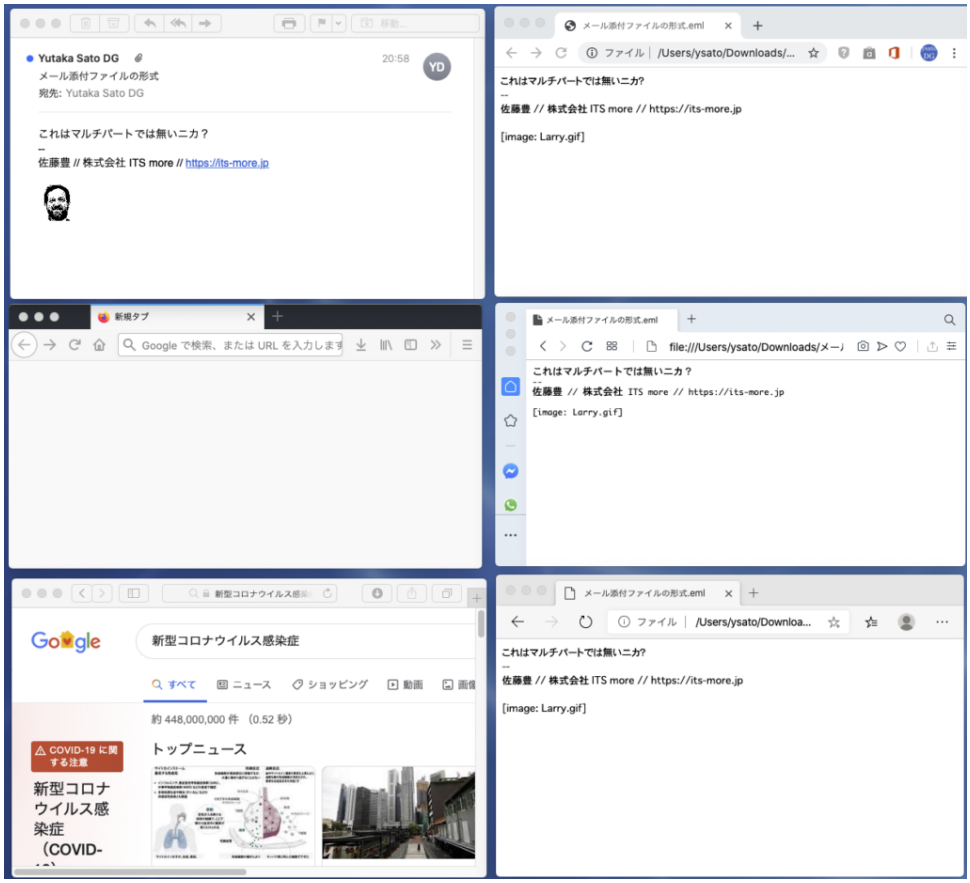
つまり、ひとつのファイルに複数の（URLで識別可能な）リソースを詰め込む事による利点は実現出来ている。また、複数に分割してHTMLを書くよりも、一枚で書いて、色なビューで切り分けて眺められる方が便利だった。だが、この方法は、ウェブサーバがDeleGateで無いと、サポートされない X-)

要するに、URLで識別されるデータの論理的な単位と、実際のデータの物理表現が１対１になっている事が根元にあるように思う。それは、ひとつのURLのデータをGETする毎に、ひとつのTCPコネクションを貼っていた、Keep-Aliveが出来る前のHTTPを思い出させる。

そもそも、MIMEフォーマットがHTTPのメッセージ形式として採用された時には、マルチパート型がそのような役割を果たすのでは無いかと期待したが、そうはならなかった。

その後この分野をフォローしていなかったので、現在そのような役割を果たす標準形式、何か別のマークアップ言語なりコンテントタイプなりで、そう言う機能が存在するのかも知れない。

さて。それでは、昨今のウェブブラウザは、URLとしてのdata:multipart ではなく、HTTPのメッセージボディとしては、Content-Type: multipart を処理してくれるのだろうか？と一瞬期待して、試してみた（↓）。メール（左上）のソースをGmailのメッセージダウンロードで .eml に吐いて、ドラッグ＆ドロップして各ブラウザに食わせた。結果は残念ながら、きちんと表示してくれるブラウザはなかった。無視または外部のブラウザ（メールツール）にまる投げするのが Firefox と Safari、一応マルチパートを解釈するがプレインテキストだけ表示するのがChrome, Opera, Edge だった。↓



ブラウザにメールを食わせて見ると…

それにしても、メールツールとウェブブラウザが一体化しなかったのは不可解だ。ウェブブラウザたる Mozilla から派生し、おそらくエンジンを共有しているであろう Thunderbird が、なぜウェブブラウザも兼ねないのだろうか？

もし両者が統合した形で発展していたら、当然マルチパート形式をウェブブラウザは理解するだろうし、それによって開けるデータ転送性能向上や新しい応用があったと思う。

ウェブサーバで提供される「ウェブメール」にしても、ブラウザでローカルに見た目が同じものを作ることはできる。逆に、ウェブ上の各種のフォーラムやらこう言ったブログの

ようなものを、メールツールからメールと見做して、つまりメールと同じ操作で読み書きすることも出来るはずだ。新着メール検出にRSSを使うことも出来るだろう。

そのためにはもちろん、データの識別子とデータ形式の標準化が必要だ。それがないから、有象無象のウェブメールや、ブログ形式が作られていて、わけがわからない。 HTMLにスクリプトやらが加わった表現能力が高過ぎるのだろう。まあ、いろんな創造性が発揮されていると言う見方は出来る。

とは言え、今ではHTTPサーバがバックエンドにMySQLなどのデータベースを備えるのが普通らしい。であれば、データベースの検索の結果出てきたデータを、ウェブとして読むか、メールとして読み書きするか、選べると言うことで良いのかも知れない。

電子メールにはURLでは無いが、Message一IDと言うユニークIDがある（ユニークだと言う保証は無いが。電子ニュースではそうだった）。メールの引用関係はハイパーテキストを構成している。メールの特長の一つは、確実な日付順に情報を並べて見れることだ。もしそれができなかったら役に立たない。Message-IDの形式を標準化して、日付を内包する形式を定めれば、データベースを検索した結果をメール的なビューに食わせる事が出来るだろう。もちろん、メールヘッダ（MIMEヘッダ）全体をキーにするのでも良い。

プライバシーの問題があるから、ひとりにひとつのデータベースにするか、暗号化してデータベースに格納する必要はあるだろう。ただ、ヘッダの検索だけでなく、本文まで全文検索をするには暗号化はよろしくないとは思う。と言うか、全文検索エンジンはいわゆるデータベースでは違うか。

色々書き散らしたので、的外れな事が多いとは思うが、一言で言えば、現状のウェブとメールの関係、ウェブサーバとメールサーバ、ウェブブラウザとメールツールの関係は、どうも納得できない。あちらでは出来る事が、こちらでは出来ない。不便に感じる、と私は言いたいのだろう。

2020-0423 sato@izmoh