

データベース出版におけるレイアウト最適化手法

亀岡 慎平 † 遠山 元道 ‡

† 慶應義塾大学大学院 理工学研究科 開放環境科学専攻

‡ 慶應義塾大学 理工学部 情報工学科

E-mail: † shinpei@db.ics.keio.ac.jp, ‡ toyama@ics.keio.ac.jp

データベースを利用する際、検索情報を端末の画面上だけでなく紙媒体に高品位で効率的に出力したいという需要が存在する。ここで問題となるのが、紙媒体は出力領域が用紙の規格によって制限されているということである。これまでデータベース内の情報を紙媒体に高品位に出力するためには、多くの場合情報をDTPなどを用いてレイアウトする必要があった。SuperSQLを用いたデータベース出版システムは、レイアウト指定演算子の組合せによって出力結果を構造化し、多様なレイアウト表現を簡単に実現可能であるというSuperSQLの特長を利用したシステムであるが、クエリによってレイアウトを指定するため、出力結果が紙面内に収まりきっているかどうかは事前にはわからない。そこで本研究では、指定されたレイアウトを極力崩さずに紙面内に収まるようにレイアウトを自動変換する手法を提案する。

キーワード：データベース出版、SuperSQL、最適化、PDF

Layout Optimization on Database Publishing

Shinpei KAMEOKA † Motomichi TOYAMA ‡

† School of Science for OPEN and Environmental Systems,

Faculty of Science and Technology, Keio University.

‡ Department of Information and Computer Science, Faculty of Science and Technology,

Keio University.

E-mail : † shinpei@db.ics.keio.ac.jp ‡ toyama@ics.keio.ac.jp

There are needs for an efficient and quality output of database query results on printed paper instead of on terminal displays. However, there arises a limitation that the output area is restricted by the size of paper. In order to overcome such inconvenience and achieve the task efficiently, we are required to utilize the applications like DTPs. Our database publishing system using SuperSQL structures the output by combining the layout specifying operators which as a result allows various layout representations. However, it is difficult to predict whether or not the output generated fits in the printed paper. In this paper, we propose a technique which efficiently rearranges the layout so that the information fits nicely in printed paper with no drastic changes in their layout.

keyword : Database Publishing , SuperSQL , Optimization , Abnormal Indication

1 背景

近年、インターネットや携帯電話、デジタルカメラに代表されるように世界規模のIT化・デジタル化が進んでいる。しかしその一方で、デジタルカメラで撮った写真を従来のようにプリントアウトする需要が絶えないように、デジタルなものを以前のもの、いわば非デジタルなものにレンダリングする需要は今後もなくならないと思われる。つまり、デジタル情報が増えれば増える程、それらを効率的に紙媒体などにレンダリングする技術が重要となるといえる。

さらに、デジタル化が進んだことで新聞や雑誌などで提供されている情報を Web 上で公開したり、オンラインで自動配信したりするといったサービスが始まっている。将来的には情報を個人の好みに合わせてカテゴリ化出来るなどの利点から、そのような情報を個人のデータベースに配信するサービス等が始まる事が予想される。つまり、個人がデジタル情報を扱う機会が増えれば増える程、それらを効率的に管理できるデータベースは個人にとって今までに比べてより身近なものになると考えられる。

個人がデータベースを利用する際、検索情報を端末の画面上だけでなく紙媒体に高品位で効率的に出力したいという需要が存在する。ここで問題となるのが、紙媒体は出力領域が用紙の規格によって制限されているということである。これまでデータベース内の情報を紙媒体に高品位に出力するためには、多くの場合情報を DTP などを用いてレイアウトする必要があった。表示されている情報を印刷したいというような場合、これでは非常に効率が悪い。そこで本研究では、ユーザの指定したレイアウトが紙面に入りきらない場合、ユーザの意図するレイアウトに最も近い形で紙面に入りきるようにレイアウトを自動変換し効率的に出力する手法を提案する。

以下、2章で関連研究を紹介し、3章で SuperSQL[1, 2]、4章で SuperSQL を用いたデータベース出版システムについて簡単に説明し、5章で本研究のレイアウト最適化手法について述べる。その後6章で今後の課題を述べる。

2 関連研究

制限された表示領域への出力対応の研究として、小さな携帯端末等からのデスクトップ PC 等を想定してデザインされた Web ページの閲覧手法がある。Bickmore らは Web コンテンツを省略・小型化して提示する Digestor システムを提案した [3]。これは主にハイパーリンクを用いたアウトライニングの技術を用いており、小型化した情報をインタラクティブにユーザに提示する。Buyukkokten らは文章の要約を行いテキストのみでユーザに提示するシステムを提案 [4]、携帯端末の通信速度なども踏まえ、要約を行うことでデータ量を減らし、より快適な閲覧手法を提供する power browser システムを開発した。前田らは SuperSQL を用いてテーブル構造を変換して提示するシステム actiview を提案している [5]。このシステムはテーブルの横幅が端末の画面に収まるようにレイアウトを自動変換してユーザに提供する。しかし、これらの研究はどれも一方方向へのスクロールは許容しており、紙媒体のように縦・横両方の領域を制限された環境ではない。

また、大規模階層型データの視覚化手法として、伊藤らは限られた領域に出来るだけ多くの情報の概観を提示するシステムを提案している [6]。

3 SuperSQL とは

SuperSQL は SQL を拡張したワンソースマルチユースを実現する言語である。その質問文は SQL の SELECT 句を

```
GENERATE< media >< TFE >
```

の構文を持つ GENERATE 句で置き換えたものである。ここで < media > は出力媒体を示し、HTML、XML、Excel、 \LaTeX などの指定ができる。また < TFE > はターゲットリストの拡張である Target Form Expression[7] を表し、連結子、反復子などのレイアウト指定演算子を持つ一種の式である。

3.1 連結子

連結子はデータベースから得られたデータをどの方向(次元)に連結するかを指定する演算子であり、

以下の3種類がある。括弧内がクエリ中の演算子を示す。

- 水平連結子 (,)

データを横に連結して出力。

例：Name, Tel

name	tel
------	-----

- 垂直連結子 (!)

データを縦に連結して出力。

例：Name! Tel

name
tel

- 深度連結子 (%)

データを3次元方向へ連結。出力がHTMLならばリンクとなる。

例：Name % Tel

name	→	tel
------	---	-----

また時間軸方向連結として、時間連結子 (#) も存在する [8]。

3.2 反復子

反復子は指定する方向に、データベースの値があるだけ繰り返かえて表示する演算子であり、以下の3種類がある。括弧内はクエリ中の演算子を示す。

- 水平反復子 ([] ,)

データインスタンスがある限り、その属性のデータを横に繰り返す。

例：[Name],

name1	name2	...	nameN
-------	-------	-----	-------

- 垂直反復子 ([] !)

データインスタンスがある限り、その属性のデータを縦に繰り返す。

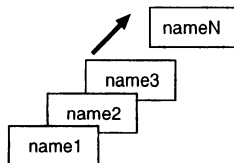
例：[Name]!

name1
name2
...
nameN

- 深度反復子 ([] %)

データインスタンスがある限り、その属性のデータを奥行き方向に繰り返す。

例：[Name]%



また時間軸方向連結として、時間反復子 ([] #) も存在する [8]。

反復子の中に反復子を用いることで、SQLにおけるグルーピングの効果を得ることができる。これにより、冗長な値の表示を制御し、より見やすい出力を得ることができる。つまり、反復子は単に構造を指定するだけでなく、そのネスト関係によって属性間の関連を指定することができる。例えば

[科目名 ! [学籍番号 , 評点] !] !

とした場合には、その各科目において学生の評点一覧が表示される。

3.3 装飾子

SuperSQLでは関係データベースより抽出された情報に、文字サイズ、セルの幅などの情報を付加できる。これらは以下のように装飾演算子 (@) によって指定することが可能である。

[s.city ! [d.name@{ "装飾指定" }] ,] !

装飾指定は“装飾子の名称 = その内容”として指定する。複数指定するときは各々を“,”で区切る。また、装飾子は属性だけでなく、反復子につけることもできる。

4 SuperSQLを用いたデータベース出版

SuperSQLでは、レイアウト演算子の組合せによって出力結果を構造化し、多様なレイアウト表現

を簡単に実現可能であるという SuperSQL の特長を利用して、新たな出力メディアとして PDF を実装することで、データベース出版を実現している。図 1 は以下のクエリでの出力例である。なお、装飾子は省略してある。

```

GENERATE PDF
[ model ! {stylist , photo} !
  '所属サロン' ! salon ! '料金設定' ! [ menu
, price ]! ]!
FROM DB 名
WHERE 結合条件 ;

```

もちろん複数のページに渡る出力も可能である。さらにこのシステムでは、出力領域が制限されているという制約の中でより多くの情報を適切にユーザーに提示するために、反復子によって繰り返される情報を折り畳む装飾指定を新たに SuperSQL に実装している。例えば、

```
[ name ]!@{ limit=3 }
```

とすると、属性 name が縦に 3 回繰り返して出力された後、横の一つずらしてまた上から 3 回繰り返して出力していくことになる。この装飾子 limit はこのような回数指定の他に長さでも指定可能である。また、装飾子の付く反復子の要素は単なる属性だけでなく、どのような入れ子であっても構わない。

この SuperSQL の PDF 出力はクエリ中でレイアウトを指定するため、出力結果が紙面内に収まっているかどうかは事前にはわからない。しかし、SuperSQL を用いることでクエリ中のレイアウト演算子を変更するだけで大きくレイアウトが変わるため、レイアウトの自動変換をする上で大きなメリットがある。そのため本研究ではこの PDF 出力モジュールをベースとして拡張する。

5 レイアウト最適化

この章では、ユーザーの指定したレイアウトが紙面に入りきらない場合、どのようにしてユーザーの意図するレイアウトに最も近い形で紙面に入りきるようにレイアウトを自動変換し、効率的に出力するかを説明する。まず、SuperSQL の PDF 出力のレイアウト指定において最上位の装飾子に最適化を行うと

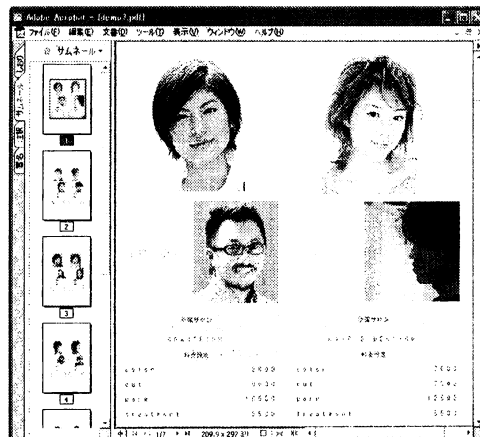


図 1: 出力例

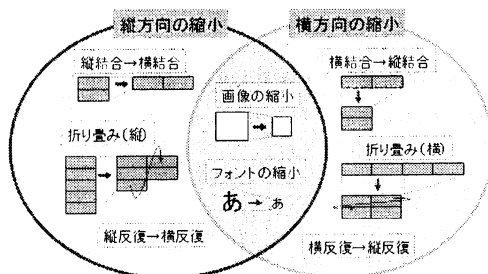


図 2: レイアウト変換要素

いう指定を記述する。この時、最適化の指定はレイアウトの一部だけに指定することは出来ず、必ず全体に対しての指定となる。必要に応じて同時に用紙のサイズ、基本点、制約条件を装飾子内に指定して実行するとシステムは自動的に PDF 出力のレイアウトを自動変換する。

以下、5.1 節で自動変換アルゴリズムで用いる各レイアウト変換要素について、5.2 節でどの変換が指定したレイアウトに一番近いかを定める評価関数について、5.3 節でユーザーのより細かな意図を反映する手法について述べる。

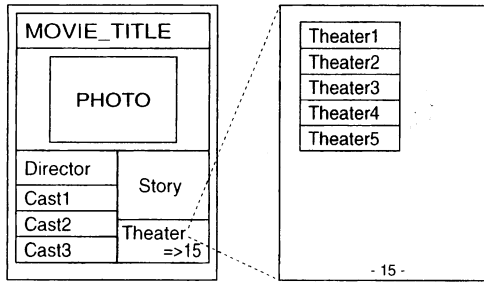


図 3: ハイパーリンクの出力

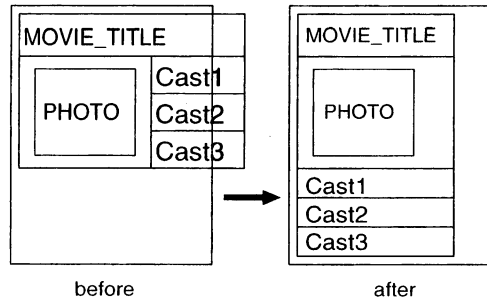


図 4: 変換例

5.1 レイアウト変換要素

SuperSQLにおいて、質問文に対して図2のような変換を適用することにより、出力領域を狭める事が出来る。変換は縦方向の縮小と横方向の縮小に大きく分けられ、これらを用いること以外に出力領域を縮小することは出来ない。もし指定したレイアウトが、出力領域を大幅にオーバーしていて、これらの変換をどう組み合わせても、指定された用紙内に情報を収めることが出来ない場合は、ページの分割を行う。ページの分割を行う場合は、その分割によってレイアウト指定演算子で指定した属性間の関連を失わないようにしなくてはならない。そのためシステムは一番外側のレベルの反復子の要素間以外で分割は行わない。ここで、一番外側の1要素自体が1ページに収まりきらない場合、今度はより内側レベルの反復子の内容をハイパーリンクを用いて別のページに送る。この時、PDFがオンライン文書ならばハイパーリンクが有効だが当然印刷してしまえばリンクの情報は失われてしまうので、印刷した際に情報が失われないようにリンク部分にはその内容を出力したページ数を一緒に出力する(図3)。

図2の変換とページの分割、そしてリンクを適用することで、用紙のサイズをオーバーしないレイアウトに変換することは可能であるとしても、それを実現する変換の組合せは一通りではなく、一般的には多数存在する。収まりきる変換の組合せのうち、検索者の意図に一番近いものを選択することが望ましい。

5.2 評価関数

本研究では各々の変換とユーザの意図との距離を評価するために加点方式を用いる。次のように各変換に基本点とその変換度合に応じた計算式を定義する。フォントと画像の縮小に関しては、

$$point \times |1 - C|$$

反復子の折り畳み、結合・反復方向の変更に関しては、

$$point \times \frac{1}{D}$$

ここでCは圧縮率、Dは入れ子構造の深さ、pointは各変換の基本点を表す。つまりフォントと画像についてはより大きなサイズ変更を行うと点数が高くなり、他の3つの変換については入れ子構造のより深いレベルで行ったほうが点数が低くなる。図4に変換例を示す。ここで変換前のレイアウト指定は

title ! { photo , [cast] ! }

変換後のレイアウト指定は

title ! { photo ! [cast] ! }

である。図4の例では、以下のような変換が行われている。

- フォントのサイズを80%に縮小
- D=2レベルでの横結合を縦結合に変換

ここで各変換の基本点を簡単に10点とすると、この例では上記の評価関数によって、

$$10 \times |1 - 0.8| + 10 \times \frac{1}{2} = 2 + 5 = 7$$

表 1: 基本点の例

変換	(a)	(b)
フォントの縮小	20	20
画像の縮小	10	30
折り畳み	×	10
横結合→縦結合	30	10
横反復→縦反復	40	40

と評価される。

指定のレイアウトが用紙のサイズを縦・横それぞれ (x,y) だけオーバーしていたとすると、本自動変換アルゴリズムは以上のような評価関数のもと、縦・横それぞれ (x,y) 以上縮小できる変換の組合せの中で最も合計点が低いものを解として選び、変換してユーザに提示する。

5.3 意図の反映

本研究の変換アルゴリズムでは、ユーザの指定したレイアウトから大きく変わらないような変換を解とするが、出力結果が必ずしも期待通りのものになるとは限らない。例えば、変換後のレイアウトが指定したレイアウトに近くても、出力ページ数が何百ページにもなっている場合にユーザの満足度が低い可能性がある。そのため本システムでは、ユーザはあらかじめ以下のような制約条件を指定することが出来る。

- 最大出力ページ数
- 最小フォントサイズ
- 折り畳みを許容するかしないか

さらに評価関数の各変換の基本点を変更することによって、より細かい意図を変換に反映することが出来る。図 5 に例を示す。

図 5(a) では

- フォントのサイズを 60% に縮小
- 画像を 50% に縮小

の変換が行われている。なぜならばこの変換は表 1 の基本点に基づき、

$$20 \times |1 - 0.6| + 10 \times |1 - 0.5| = 8 + 5 = 13$$

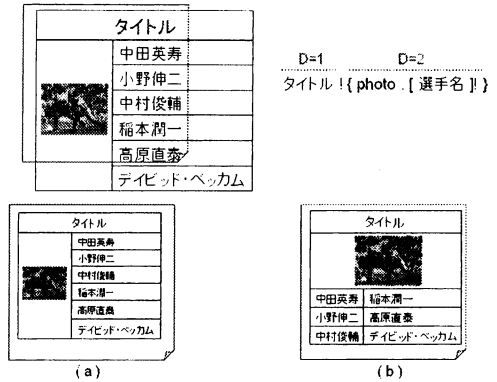


図 5: 表 1 の基本点に基づいた変換例

と評価され、これが最小の点数となる変換の組合せだったからである。例えば、フォントを 10% にまで圧縮することでも紙面内に収めることが出来るが、その場合、

$$20 \times |1 - 0.1| = 18$$

となり先程の 13 点よりも大きいのでこの変換は採用されない。図 5(b) では基本点が表 1 のようになっているため、図 5(a) の変換とは異なり、

- フォントのサイズを 70% に縮小
- D=2 レベルでの縦反復の折り畳み
- D=2 レベルの縦結合を横結合に変更

の変換が行われている。この変換は、

$$20 \times |1 - 0.7| + 10 \times \frac{1}{2} + 10 \times \frac{1}{2} = 3 + 5 + 5 = 13$$

と評価される。この場合の基本点で先程と同じ変換を評価すると、

$$20 \times |1 - 0.6| + 30 \times |1 - 0.5| = 8 + 15 = 23$$

となるためこの変換は採用されない。

このように基本点を変えることで最小点となる変換の組合せが変わるレイアウト変換の基本点を下げることなどによって、ユーザの細かい意図を自動変換に反映することが可能である。

6 今後の課題

まず第一に各変換の基準点の初期値の定義とその妥当性の検証が必要である。さらにその基準点を用いた2つ計算式の妥当性の検証、この2点が主な課題である。特に評価に用いる計算式は現在極めて単純なものであり、今後テストを繰り返しつつ改善していきたい。基準点に関しては初期値以外に用紙の規格などに対応するセットをいくつか定義しその妥当性についても検証を行う。さらに、点数が最小となる変換の組合せを決定するアルゴリズムに関しても、素早く解を導くことの出来る効率的な手法について研究していきたい。また出力するコンテンツに文字が多い場合は通常よりもフォントの縮小による効果が大きくなること等から、コンテンツによる基準点変化の可能性についても研究を進めていきたい。

7 まとめ

データベース出版における出力領域の制約に対して、従来のようなDTPなどを用いた細かいレイアウト指定ではなく、ユーザの指定したレイアウトを極力崩さないレイアウトの自動変換手法を提案した。その際システムには、クエリ中のレイアウト演算子を変更するだけで大きくレイアウトが変わるというレイアウトの自動変換をする上で大きなメリットがあるSuperSQLを用いたデータベース出版システムを採用した。また、自動変換に用いる評価関数や実際の変換例を示した。そして、基本点を変更することなどによりより細かいユーザの意図を自動変換に反映できることを示した。

参考文献

- [1] Motomichi Toyama, "SuperSQL: An Extended SQL for Database Publishing and Presentation," *Proceedings of ACM SIGMOD '98 International Conference on Management of Data*, pp. 584-586, 1998
- [2] SuperSQL: <http://www.db.ics.keio.ac.jp/ssql>
- [3] Timothy W. Bickmore and Bill N. Schilit, "Digester: Device-independent Access to the

World Wide Web." *Proceedings of the 6th International WWW Conference*, pp.655-663, 1997

- [4] Orkut Buyukkokten, Oliver Kaljuvee, et al, "Efficient Web Browsing on Handheld Devices Using Page and Form Summarization." *ACM Transactions on Information Systems*, Vol. 20, No. 1, pp. 82-115, 2002
- [5] Yoko Maeda, Motomich Toyama, "ACTIVIEW: Adaptive data presentation using SuperSQL." *27th VLDB*, 2001
- [6] Takayuki Itoh, Yumi Yamaguchi, et al, "Hierarchical Data Visualization Using a Fast Rectangle-Packing Algorithm" *Proceedings of the IEEE Transactions on Visualization and Computer Graphics*, Vol. 10, No. 3, pp. 302-313, 2004
- [7] 遠山 元道, "ターゲットリストの拡張によるデータベース出版と概視の実現": 信学技報, 電子情報通信学会, Vol.93, No.152, pp. 79-88, 1993
- [8] 笹田 麻衣子, 遠山 元道, "SuperSQLのHTML出力における時間連結子の実装": 日本データベース学会 *Letters*, Vol.2, No.1, pp. 135-138, 2003