

# SuperSQL クエリ自動修正機構の提案

星野 友亮<sup>†</sup> 五嶋 研人<sup>†</sup> 遠山元道<sup>††</sup>

<sup>†</sup> 慶應義塾大学理工学研究科開放環境科学専攻 〒223-8522 神奈川県横浜市港北区日吉 3-14-1

E-mail: <sup>†</sup>{yusuke,goto}@db.ics.keio.ac.jp, <sup>††</sup>toyama@ics.keio.ac.jp

あらまし SuperSQL は、関係データベースの出力結果を構造化し、多彩なレイアウト表現を実現する SQL の拡張言語である。現在、SuperSQL による Web ページの開発には主に SSedit という専用のエディタが用いられる。このエディタにより開発環境は飛躍的に改善されたが、開発者に表示されるエラーの品質に起因する二次エラーの問題が残る。そこで本論文では、SuperSQL による Web ページ開発時の実際のエラーログを取得し解析することで開発者のエラーパターンを分類。その分類に基づき、SuperSQL クエリのシンタックスエラーを自動で修正し、開発者にその解決策であるクエリの候補を提示する自動修正機構を提案する。

キーワード SQL, SuperSQL, エラー, オートコレクト

## 1. はじめに

SuperSQL は関係データベースの出力結果を構造化し、多様なレイアウト表現を可能とする SQL の拡張言語である。通常の SQL では、シンプルでフラットな表しか再現できず、自由なレイアウトの表を作るためには HTML や CSS といった他の言語の導入が不可欠である。しかし SuperSQL を用いることで他の言語を使用することなく様々なレイアウトな表を作成することができ、SQL、構造、メディアの三次元的組み合わせによって SuperSQL による開発は無限の広がりを見せると言える。また、SuperSQL を用いると、HTML と PHP を用いた一般的な方法に比べて、はるかに少ない行数で Web ページを生成することができる。

現在、SuperSQL を用いた Web ページの開発にはコマンドラインでの実行の他に、SSedit という専用のエディタを使用する方法がある。このエディタの導入により、データベースのテーブル・スキーマの表示や新規作成用のクエリテンプレートの挿入、ファイルの操作などを GUI で行うことができるようになり、SuperSQL での Web ページ開発環境は飛躍的に改善された。しかし、エラー表示の分かりにくさや読みにくさ、不正確さなどといった、SSedit が開発者に提示するエラーの品質に起因する二次エラーの問題が依然として残っている。そこで本論文では、SuperSQL による Web ページ開発時に発生した実際のエラーログを取得し解析することで、開発者に共通する頻出シンタックスエラーパターンを分類する。その分類に基づきパターン化することで、パターンとマッチした SuperSQL クエリのシンタックスエラーを自動で修正し、開発者にその解決策であるクエリの候補を提示する自動修正機構を提案する。

以下に本稿の構成を示す。2 章で SuperSQL の概要、3 章で研究背景について述べる。4 章ではエラーログの分析について述べ、5 章で自動修正機構について述べる。そして、6 章では評価について述べ、7 章で結論と今後の課題を述べる。

## 2. SuperSQL

この章では SuperSQL について簡単に述べる。SuperSQL は関係データベースの出力結果を構造化し、多様なレイアウト表現を可能とする SQL の拡張言語であり、慶應義塾大学遠山研究室で開発されている [1], [2]。そのクエリは 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、PDF 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 を表し、結合子、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。

### 2.1 結合子

結合子はデータベースから得られたデータをどの方向 (次元) に結合するかを指定する演算子であり、以下の 3 種類がある。括弧内はクエリ中の演算子を示している。

- 水平結合子 (,)

データを横に結合して出力。

例: Name, Tel 

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

- 垂直結合子 (!)

データを縦に結合して出力。

例: Name! Tel 

name
tel

- 深度結合子 (%)

データを 3 次元方法へ結合。出力が HTML ならばリンクとなる。

例: Name % Tel 

name
------

tel
-----

### 2.2 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して表示する。また反復子はただ構造を指定するだけでなく、そのネストの関係によって属性間の関連を指定できる。例えば

[科目名]! ,[学籍番号]! ,[評点]!

とした場合には各属性間に関連はなく、単に各々の一覧が表示されるだけである。一方、ネストを利用して

[科目名! [学籍番号, 評点] ]!

とした場合には、その科目毎に学籍番号と評点の一覧が表示されるといったように、属性間に関連が指定される。以下、その種類について述べる。

- 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し表示する。

例: [Name] ,

name1	name2	...	name10
-------	-------	-----	--------

- 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し表示する。

例: [Name]!

name1
name2
...
name10

## 2.3 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、文字スタイル、横幅、文字色、背景、高さ、位置などの情報を付加できる。これらは装飾演算子 (@) によって指定する。

< 属性名 >@{ < 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。

クエリ例: [employee.name@{width=200, color=red}]!

## 2.4 関数

### 2.4.1 image 関数

image 関数を用いると画像を表示することが可能となる。引数には属性名、画像ファイルの存在するディレクトリにパスを指定する。

クエリ例: image(Pict, “./picture”)

### 2.4.2 link 関数 (出力メディアが HTML の場合のみ)

link 関数は、FOREACH 句と同時に用いる。深度結合子と同様にリンクを生成することができる。リンクを生成するために深度結合子を用いる場合は全てを一つの質問文で記述するが、link 関数と FOREACH 句を用いる場合はリンク元のページとリンク先のページを別々の質問文で記述する。リンク元を生成する質問文で link 関数を、リンク先を生成する質問文で

FOREACH 句を記述する。以下に例を示す。

クエリ例:

```
<Q1.ssql>
GENERATE HTML
[e.name % {e.salary, e.syear}]!
FROM employee e
```

これを link 関数と FOREACH 句を用いると、Q2.ssql と Q3.ssql の二つの質問文となる。

```
<Q2.ssql>
GENERATE HTML
[link(e.name, "Q3.ssql", e.id)]!
FROM employee e

<Q3.ssql>
FOREACH e.id
GENERATE HTML
[e.salary, e.syear]!
FROM employee e
```

link 関数ではペアとなる FOREACH 句を含む質問文のファイル名を第二引数で指定する。さらに、link 関数の第三引数と FOREACH 句で同じ属性を指定する。なお、この属性の値を利用して URL を生成するため、指定する属性はリンク先のページを一意に識別できるもの(主キーなど)を選択する。2 つ以上の属性を指定する場合(主キーが 2 つ以上あるようなとき)は、link 関数では第三引数以降に属性を一つずつ指定し、FOREACH 句では属性を(,)で区切って指定する。

## 2.5 ソーティング (並べかえ)

SuperSQL では属性に基づき、昇順もしくは降順にソーティングすることができる。対象になる属性の前に括弧を記述し、その中にソーティングの種類や順序を入れる。ソーティングの種類には asc と desc の二つがあり、ソーティングに順序付けをする場合にはその右側に数字を記述する。数字の小さいものから優先的に処理され、数字指定がない場合は記述された順番に上から処理される。

- 昇順 (ascending order)

従業員名の昇順によってソーティングを行う。

クエリ例: (asc)e.name

- 降順 (descending order)

給料の降順によってソーティングを行う。

クエリ例: (desc)e.salary

## 3. 研究背景

現在、SuperSQL による Web ページの開発には主に SSedit という専用のエディタ (図 1) が用いられる [3]。このエディタの導入により、SuperSQL での Web ページ開発環境は飛躍的に改善された。また、このエディタは検索やファイルの一括検

索、置換などといった様々な拡張機能により、開発者のクエリ作成をより強力に支援する [4] .

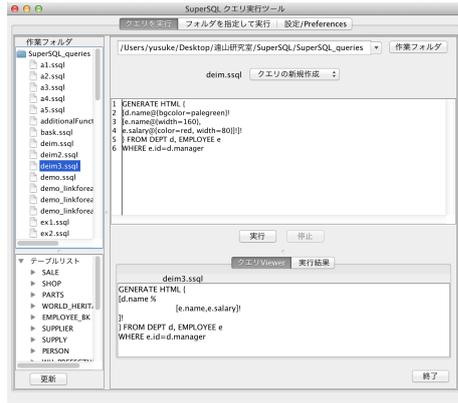


図 1 SSedit

しかし、SuperSQL を用いた Web ページ開発において、開発者はシンタックスエラーを数多く起こす。慶應義塾大学理工学部情報工学科の3年生を対象とした授業であるデータモデリング (担当教員: 遠山) での最終課題で、実際に SSedit を用いて Web ページを作成してもらい、その際の学生の実行ログとエラーログを採取したところ、2015 年春学期の授業で SuperSQL が出力したエラー総数は 10637 であった。これは図 2 のように分類される。また、2014 年春学期と 2014 年秋学期の授業において採取したログについても同様に分類を行うと図 3 のようになり、多少のばらつきはあるがどの学期の授業においても主に見受けられるエラーの種類は同じものであるということが証明された。さらにそれぞれの学期の、全学生の実行回数に対するエラー比率は、23.5%、25.3%、22.9%であり、学生の記述するクエリの 2 割がシンタックスエラーを含むクエリであるということが分かった。

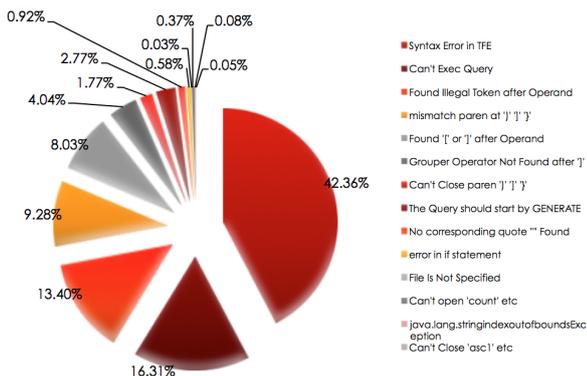


図 2 エラー分類 (2015 年春学期)

例えば、下記のようなシンタックスエラーを含むクエリ例に対して SuperSQL が出力するエラーは図 4 である。学生は縦反復による図 5 のような構造、もしくは横反復による図 6 のような構造の表を出力したかったのであろうと考えられるが、そのためには [] の後ろに (!) もしくは (,) を記述し反復子の形をとらなければならない。この見かけにはわずかな書き忘れのため

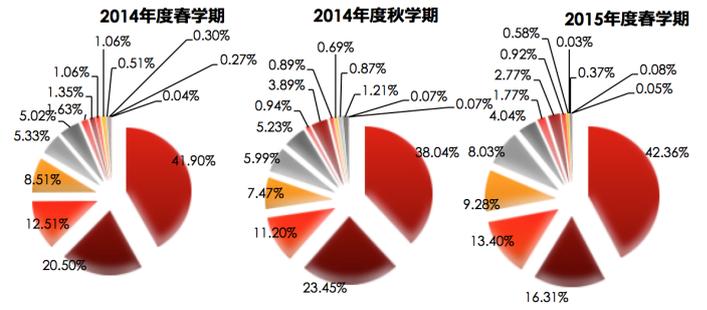


図 3 エラー分類 (3 学期分)

に、学生はログから見るに 10 分ほど似たようなクエリを記述し試行錯誤を繰り返していた。このような事象は、SSedit が開発者に提示するエラーの品質に起因する二次エラーであると言える。そこで、シンタックスエラーを自動で修正し、開発者にその解決策であるクエリの候補を提示することで、エラー表示の分かりにくさから生まれる新たなエラーを回避し、開発速度を向上させることができると考える。

クエリ例:

```
GENERATE HTML
{ [e.dept, {e.floor! e.name! e.category}] }
FROM EMPLOYEE e;
```

```
*** Grouper Operator Not Found after ']' ***
Error[TFEparser]: Syntax Error in TFE
{[e.dept,{e.floor!e.name!e.category}] >>>> } <<<<<
結果: 失敗
```

図 4 出力されるエラー

Foods	1F
	Miller
	Hobby and Life
Wears	2F
	Wilson
	Fashion
Books	1F
	Smith
	Hobby and Life

図 5 縦反復による表

Foods	1F			1F
	Miller			Smith
	Hobby and Life			Hobby and Life
		Wears	2F	
			Wilson	
			Fashion	

図 6 横反復による表

#### 4. エラー分析

3 章で述べたように、SuperSQL を用いて Web ページを開発する際に主に見受けられるエラーは同じ種類のものであるということから、開発者が起こしがちなシンタックスエラーはある程度予測することができる。この章では、SuperSQL が出力するエラーの種類とは別に、実際に開発者が起こしたエラーをクエリログから分析し、分類する。表 1 に開発者が起こした実際のシンタックスエラーパターンを示す。

表 1 シンタックスエラーパターンの分類

種類	具体例
反復子の後の (!)(,) 書き忘れ	[e.name, e.salary]
[]! [], の後の結合子の書き忘れ	[e.name]! [e.salary]
余分な結合子	e.name,! e.salary! e.dept
装飾子の書き方	e.namebgcolor=red, width=100
装飾子の後に装飾子を付ける	e.name@bgcolor=red@width=100
ソーティングを属性以外に付ける	(asc[e.name]!)
括弧の使い方の間違い	(e.name)!
; と:の間違い	from employee e:
{ } や [] の前に結合子が足りない	e.nameee.salary
属性や関数の書き間違い	[e.nane]! (decs)e.name

## 5. 実装例

4章で述べたエラー分析の結果(表1)に基づきシンタックスエラーをパターン化し記憶．そのパターンを用いたクエリ自動修正機構の実装例を提示する．

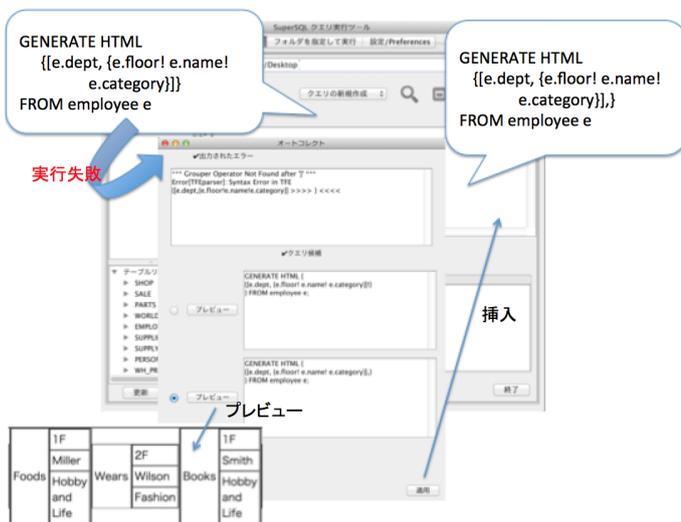


図 7 実装例

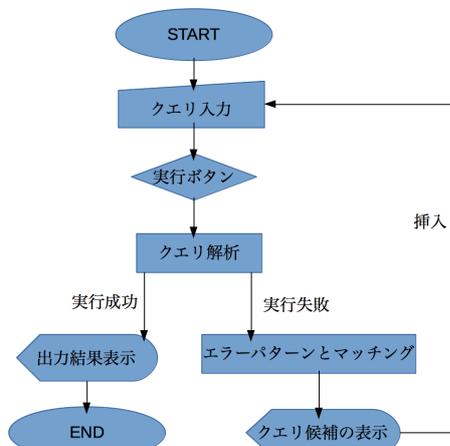


図 8 フローチャート

図 7 に示すように、シンタックスエラーが含まれるクエリが実行されその実行が失敗となる．その場合に記述されたクエリを解析することで、反復子の後の (!)(,) の書き忘れや余分な結合子などといった表 1 に基づくシンタックスエラーパターンとマッチング．マッチングした結果からシンタックスエラーの解決策であるクエリの候補を予想し開発者に提示する．提示されたクエリの横にあるラジオボタンを選択し適用ボタンをクリックすることで、選択した候補がクエリ画面に挿入される．これにより開発者はクエリのデバッグのために似たようなクエリを書いては実行するという手間をかけることなく開発を続けることができる．また、提示されたクエリの候補が開発者の意図に沿う出力をするかどうか問題となってくる．そこで、選択したクエリ候補をクエリ画面に挿入する前に出力結果を確認できるようにするため、プレビュー機能を用意する．プレビュー機能では選択されたクエリの候補の出力結果を別ウィンドウで開発者に提示する．図 8 に自動修正機構のフローチャートを示す．

## 6. 評価

現在、評価に関しては調査中であるが、本研究では 2 つの方法で評価を行う予定である．1 つ目はエラー比率、開発時間による性能評価．2 つ目は既存のオートコレクトツールとの性能比較による評価である．

### 6.1 エラー比率、開発時間による性能評価

3章で述べたように現在の SuperSQL による Web ページ開発の際の実行回数に対するエラー比率は約 20%であり、クエリ自動修正機構を導入し開発を行った場合のエラー比率と比較し評価する．また同様に、開発時間についてもクエリ自動修正機構の導入前と導入後での変化を比較し評価する．

### 6.2 既存のオートコレクトツールとの比較による評価

php が生成する html を自動で修正するシステムや [5] や html の構文のチェックを行う Another HTML-lint [6] , html の構文をチェックし自動で修正するプラグインを Jedit に対して提供する Jchecker X [7] , Web ページ作成の際に誤りのある html の構文を自動で修正する機能を持つホームページ・ビルダー [8] などと性能を比較し評価する．

- 構文チェック機能
- 自動修正機能
- プレビュー機能

などといった点で評価することを予定している．

## 7. おわりに

### 7.1 結論

開発者に表示されるエラーの品質に起因する二次エラーの問題を解決するために、SuperSQL による Web ページ開発時の実際のエラーログを取得し解析することで開発者のエラーパターンを分類．その分類に基づき、SuperSQL クエリのシンタックスエラーを自動で修正し、開発者にその解決策であるク

エリの候補を提示する自動修正機構を提案する .

## 7.2 今後の課題

ログを更に分析しシンタックスエラーパターンを充実させることで自動修正機構の改善に繋がると考える . また , 現在はクエリに 1 つのシンタックスエラーが存在する単独エラーが検出された場合についてクエリの自動修正を行っているが , クエリに 2 つ以上のシンタックスエラーが存在する複合的なエラーの検出 , 自動修正についで検討する必要があると考え , さらに解決策となり得るクエリ候補が多数ある場合の表示の仕方や候補の切り捨て等についても検討事項である .

## 文 献

- [1] SuperSQL: <http://SuperSQL.db.ics.keio.ac.jp/>
- [2] M . 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
- [3] 木谷 将人, 五嶋 研人, 遠山 元道. "SuperSQL クエリ作成支援システムの開発 " データ工学ワークショップ, DEIM2014.
- [4] 星野 友亮, 五嶋 研人, 遠山 元道. "SuperSQL クエリ作成支援システム SSedit の拡張 " データ工学ワークショップ, DEIM2015.
- [5] Xiaoyin Wang, Lu Zhang, Tao Xie, Yingfei Xiong, Hong Mei: "Automating Presentation Changes in Dynamic Web Applications via Collaborative Hybrid Analysis " , SIGSOFT '12/FSE-20, November 11-16, 2012, Cary, North Carolina, USA.
- [6] Another HTML-lint: <http://www.htmlint.net/html-lint/index.html>
- [7] Jchecker X: [http://www.artman21.com/jp/jchecker\\_x/](http://www.artman21.com/jp/jchecker_x/)
- [8] ホームページ・ビルダー: <https://www.justsystems.com/jp/products/hpb/>