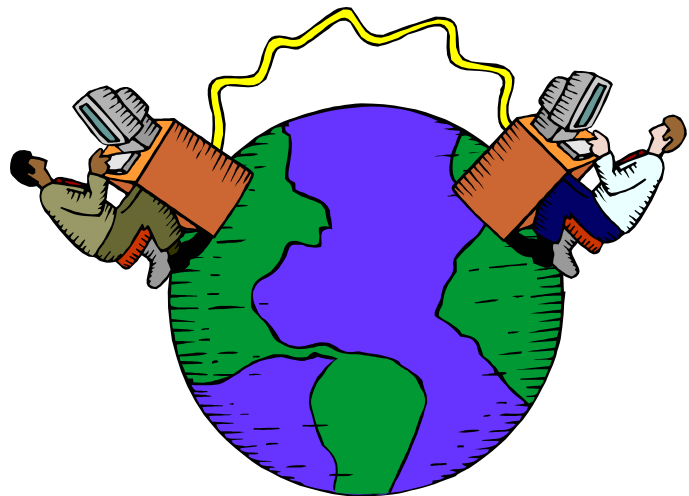


# 意味的に拡張したXMLと それに対する問合せ機構を持つシステムの実現

言語・ソフトウェア工学研究室  
的野晃整

# はじめに



## XML

- 電子化文書の構造化
- タグを用いて自由に意味を付加
- 情報交換のデファクトスタンダード
- 木構造

## 目的

❗ XMLの木構造を超えている

- 複合オブジェクトを表現
- 個別化情報の表現
- マルチメディア情報の統合

XMLの拡張

# XMLの拡張

- 有向グラフの導入
  - 識別子
  - 等値制約
- 情報の共有化
  - 部分文字列の共有
  - 意味的な親の指定
- 構成子の導入
  - 集合、列
- 拡張属性の導入

# 拡張XMLの問題点

識別子  
等値制約  
構成子  
部分文字列  
意味的な親  
拡張属性

- I. 表現が多い 統一の意味
- II. 記述が面倒 (XMLに準拠)
- III. 問合せ (検索、変換) 表現がない

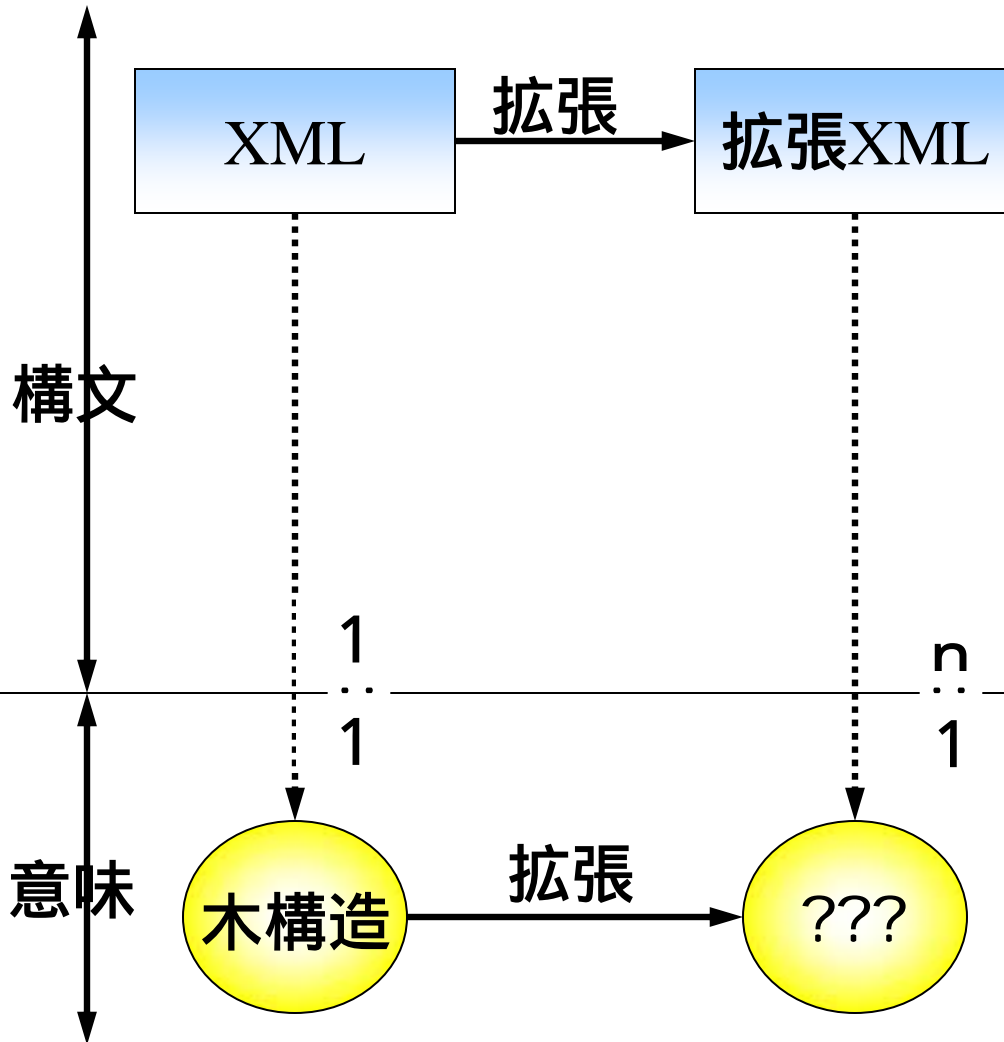


- I. 意味論となるグラフ

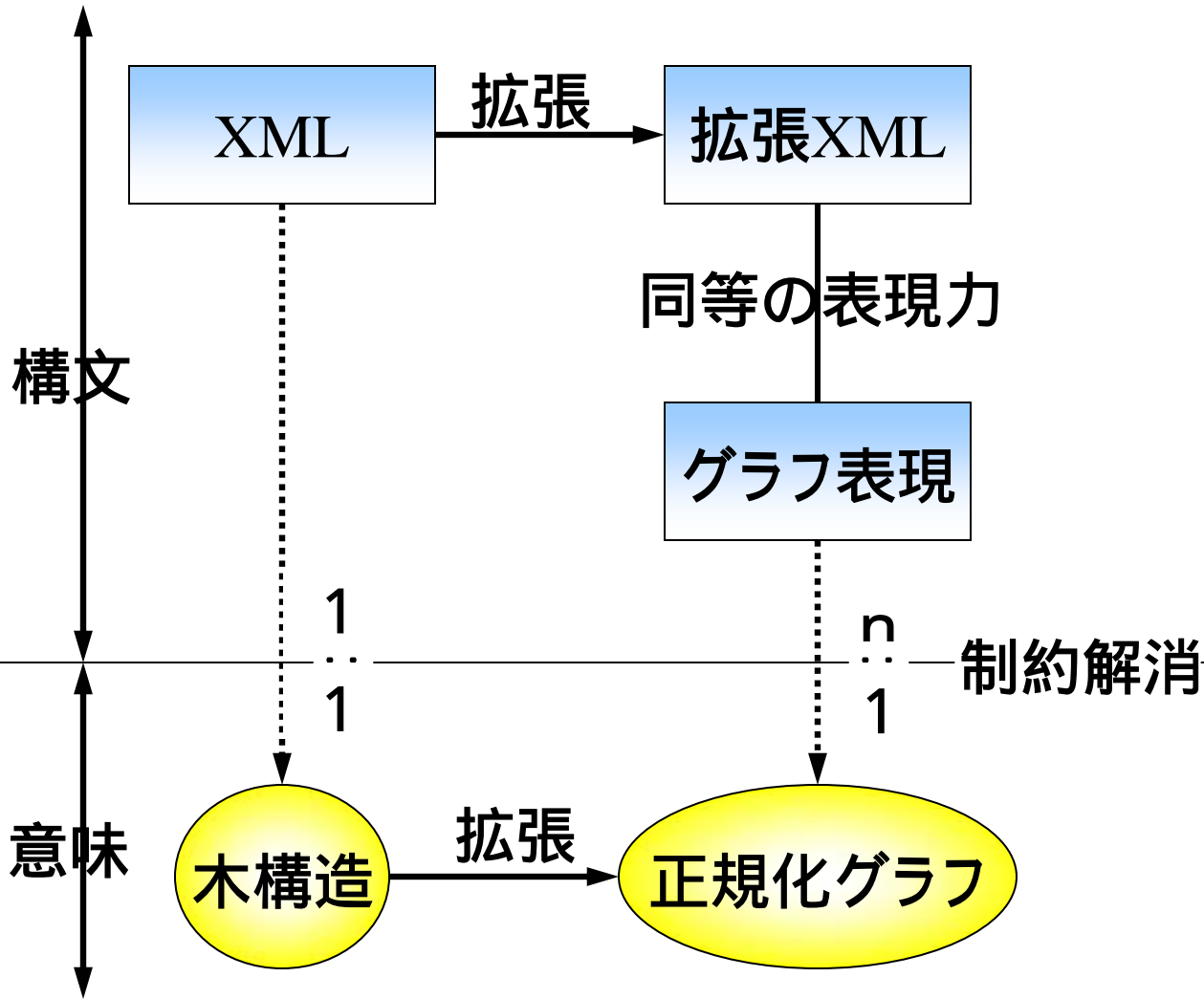


- II. III. 問合せ表現を持つ *Qpit* 言語

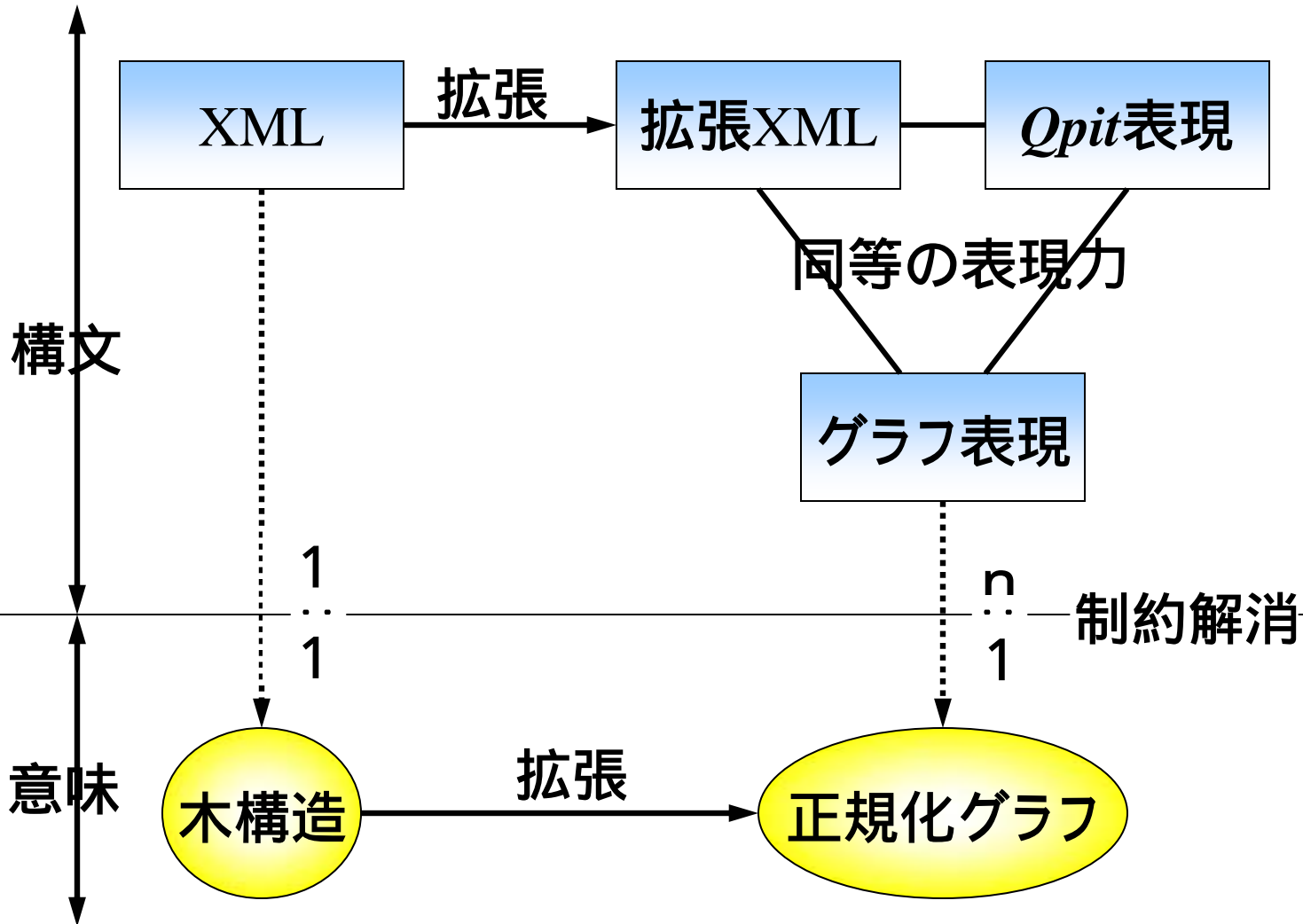
# 関係



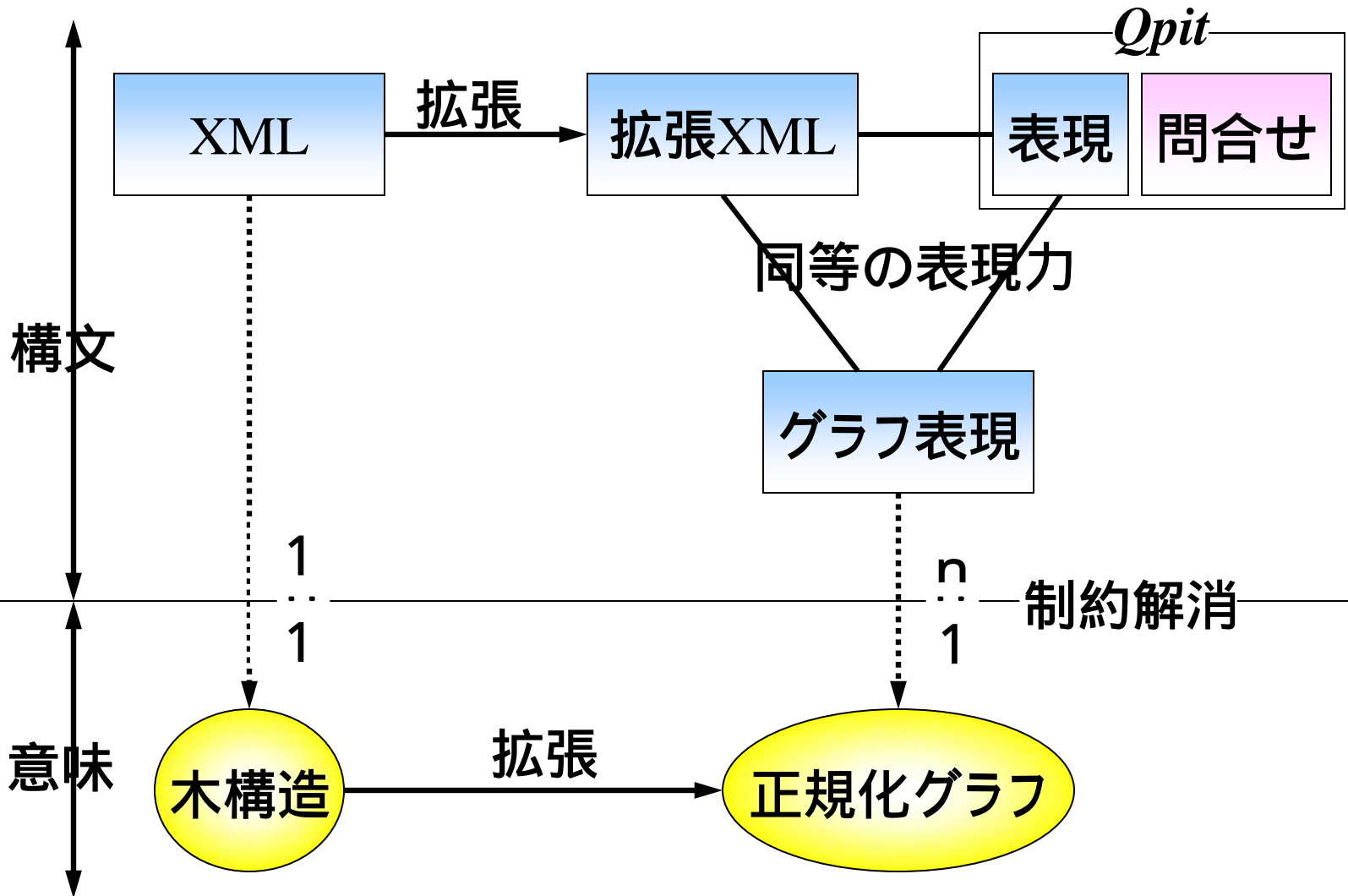
# 関係



# 関係



# 関係





# モデル

```
<family q:constructor="set">
```

```
  <person q:id="sazae">
```

```
    <name>
```

```
      <q:Ref substring="name(0,3)"/>
```

```
      サザエ
```

```
    </name>
```

```
    <husband q:idref="masuo"/>
```

```
  </person>
```

```
  <person q:id="masuo">
```

```
    <name q:id="name">
```

```
      フグ田マスオ
```

```
    </name>
```

```
    <wife q:idref="sazae"/>
```

```
  </person>
```

```
</family>
```

```
<hobby q:parents="#masuo"
```

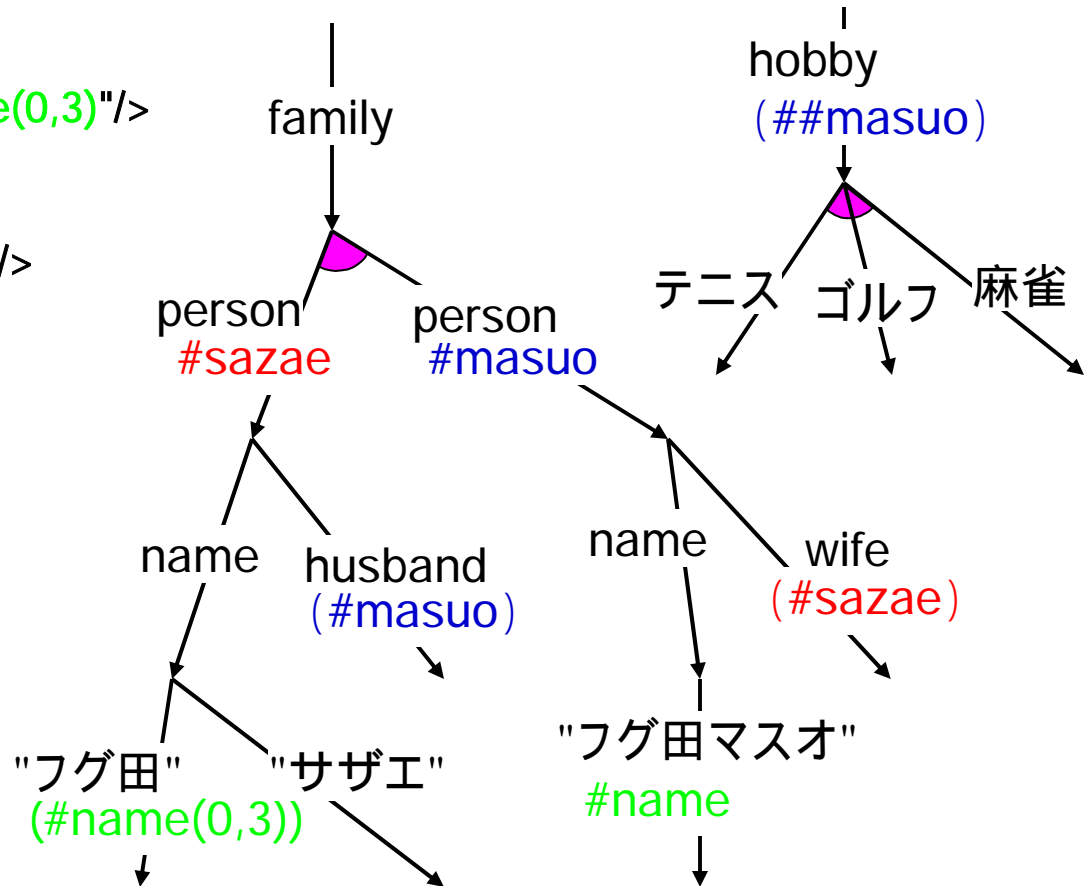
```
  q:constructor="set">
```

```
  <q:Pcdata>テニス</q:Pcdata>
```

```
  <q:Pcdata>ゴルフ</q:Pcdata>
```

```
  <q:Pcdata>麻雀</q:Pcdata>
```

```
</hobby>
```



# グラフ表現

$\langle \text{グラフ} \rangle ::= \{ \langle \text{式} \rangle, \dots \}$

$\langle \text{式} \rangle ::= \langle \text{グラフ変数} \rangle = (\langle \text{id} \rangle, \langle \text{要素名} \rangle, \{ \langle \text{属性名} \rangle = \langle \text{属性値} \rangle, \dots \}, \{ \langle \text{要素値} \rangle, \dots \})$

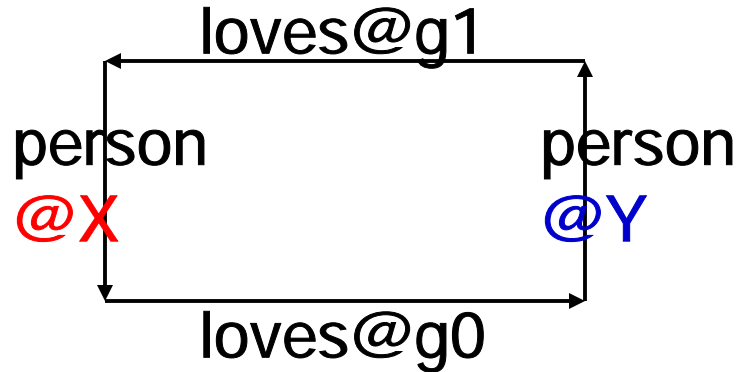
$\langle \text{要素値} \rangle ::= \langle \text{PCDATA} \rangle \mid \langle \text{グラフ変数} \rangle$

$\langle \text{グラフ変数} \rangle$  は  $\langle \text{式} \rangle$  間の等値制約の役割を果たしている。

$\langle \text{id} \rangle$  は  $\langle \text{式} \rangle$  の識別子である。

$\langle \text{PCDATA} \rangle$  は原子値である。

# 等値制約による相思相愛



```
<person q:constraint="X">  
  <loves>  
    <q:Ref conref="Y"/>  
  </loves>  
</person>  
<person q:constraint="Y">  
  <loves>  
    <q:Ref conref="X"/>  
  </loves>  
</person>
```

$X = (\_, \text{person}, \{\}, \{g_0\})$

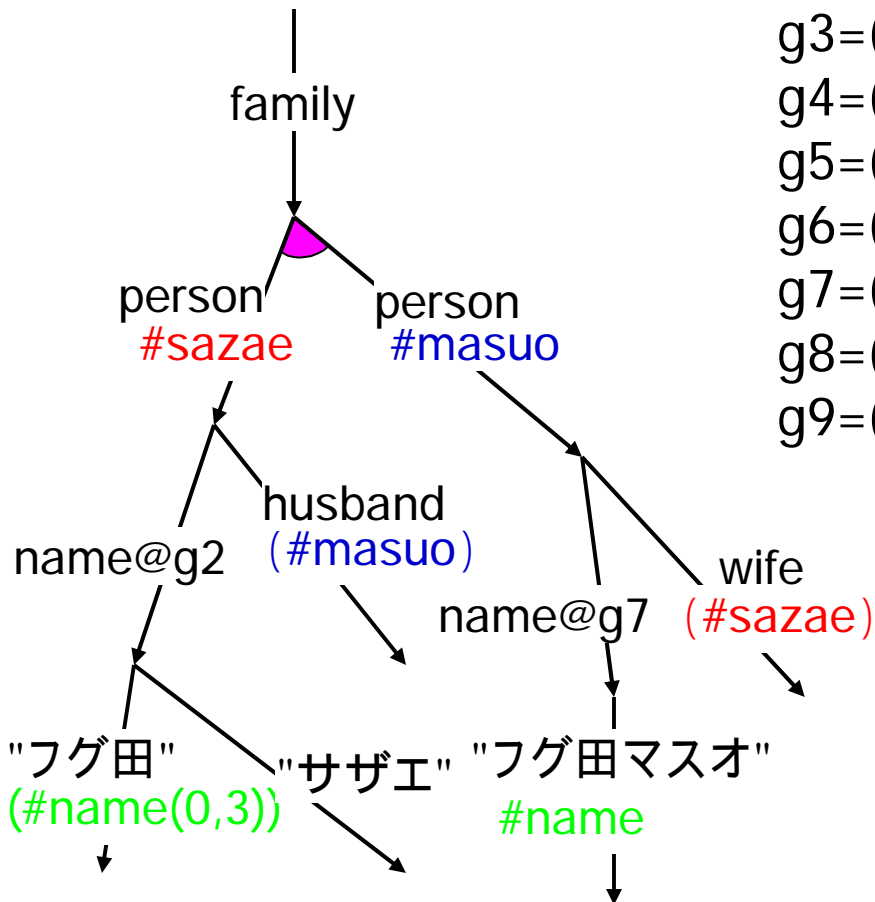
$g_0 = (\_, \text{loves}, \{\}, \{Y\})$

$Y = (\_, \text{person}, \{\}, \{g_1\})$

$g_1 = (\_, \text{loves}, \{\}, \{X\})$

# グラフ表現 (複合オブジェクト)

```
g0=(_, family, {constructor="set"}, {g1, g6})
g1=(sazae, person, {}, {g2, g5})
g2=(_, name, {}, {g3, g4})
g3=(_, PCDATA, {substring="#name(0,3)"}, {})
g4=(_, PCDATA, {}, {"サザエ"})
g5=(_, husband, {idref="masuo"}, {})
g6=(masuo, person, {}, {g7, g9})
g7=(_, name, {}, {g8})
g8=(name, PCDATA, {}, {"フグ田マスオ"})
g9=(_, wife, {idref="sazae"}, {})
```



# *Qpit* の特徴

## *Qpit* 表現

- 項とパスによる表現
- 高レベルな記述
- グラフ・拡張XMLと同等の表現力

## 問合せ表現

- パスによるフィルタリング表現
- 変数と正規表現の導入
- ルールによる構造の変更

# Qpit 表現

## • 項

- ラベル
  - A Aは要素名
  - "A" Aは原子値
- 等値制約...@A
- 識別子...#A
- 構成子(sequence|set)
- 参照...(A)
  - #A 識別子参照
  - @A 等値制約参照
  - #A(0,3) 部分文字列参照
- 拡張属性...(A)

## 例)

"今日はいいい天気だ"

人(name="太郎" age="20")

hobby(set)

@X

(#taro)

(#section1(23, 54))

# *Qpit* 表現

- パス(項の集合)
  - $A / B$  Aは親の項、Bは子の項
  - $/ A$  Aは親の項を持つ
  - $A /$  Aは子の項を持つ
  - $A = B$  AとBは存在が同じ項
  - $[A, B]$  AとBは兄弟項

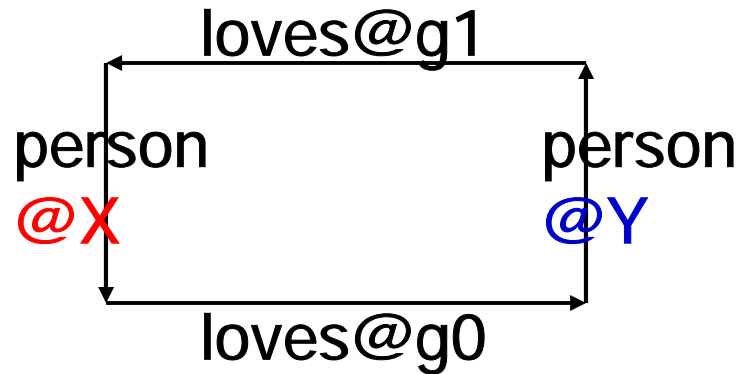
例)

$/岡山県/総社\#soja/岡山県立大学\#OPU/$

$人/恋人=人/名前/"花子"$

$夫婦/[夫, 妻]/息子/$

# *Qpit* 表現 (相思相愛)



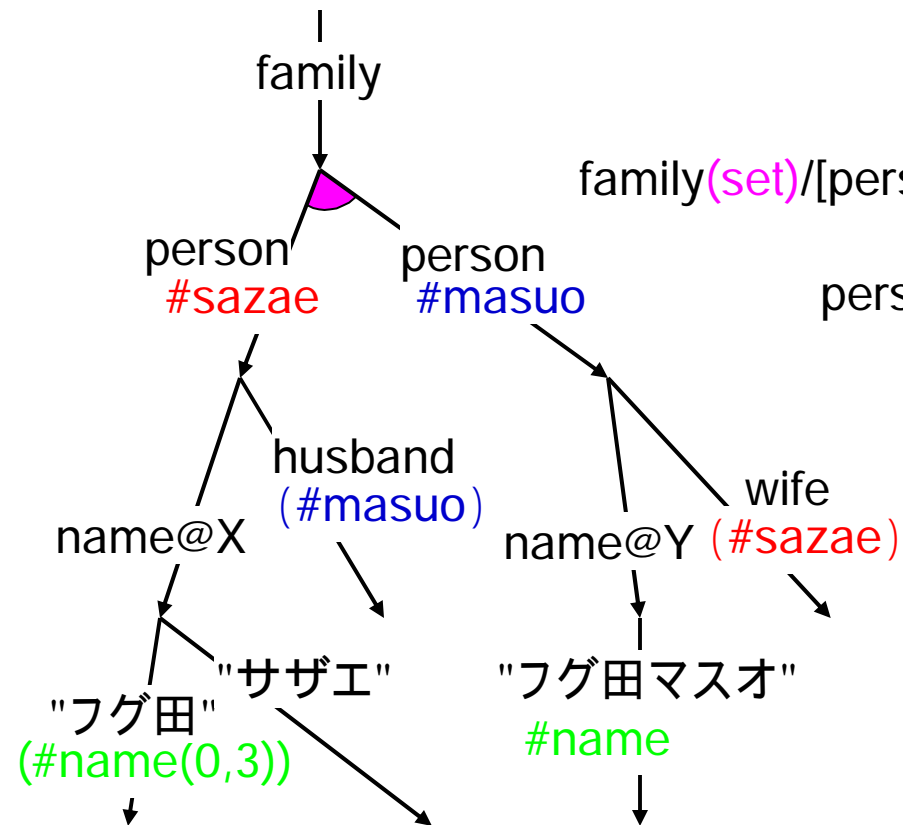
person@X/loves@g0/person@Y/loves@g0/(@X)

person@X/loves@g0/(@Y)

person@Y/loves@g1/(@X)



# Qpit 表現 (複合オブジェクト)



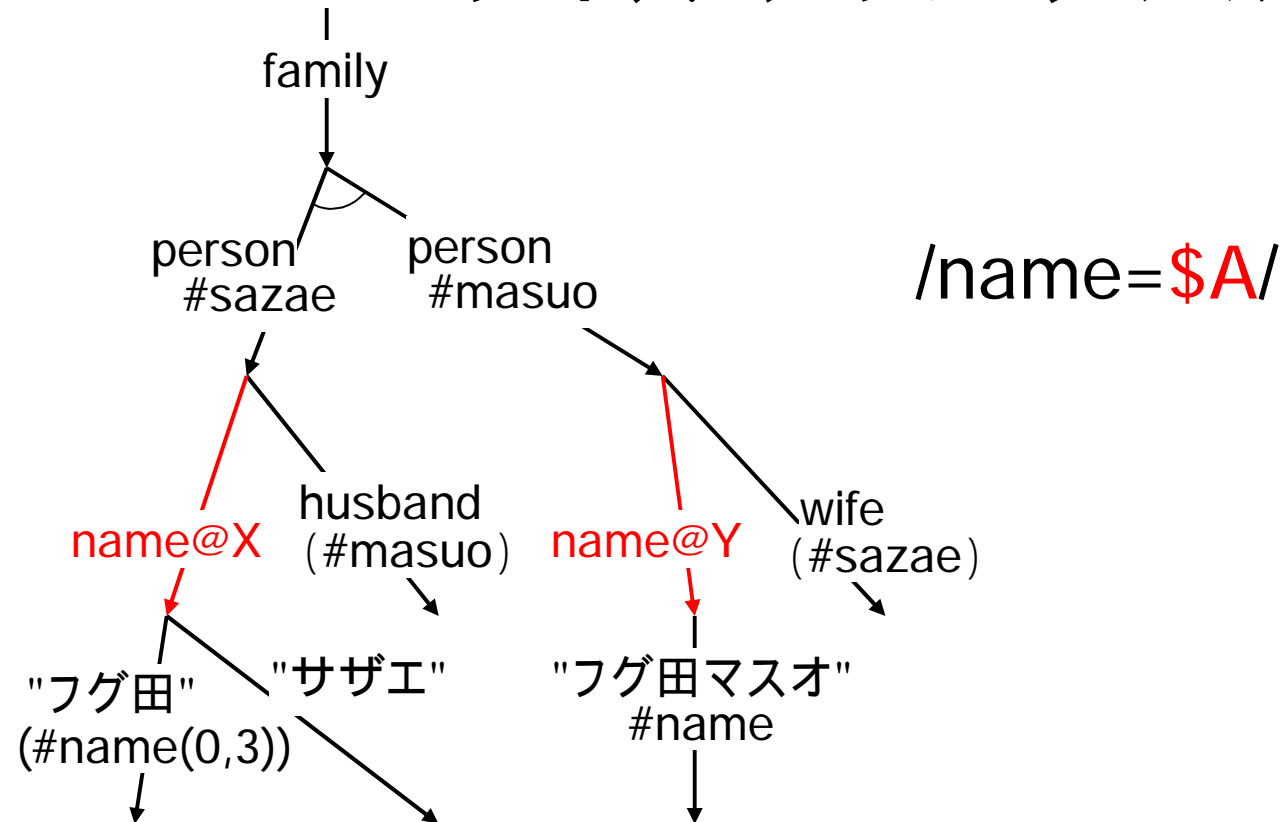
```
family(set)/[person#sazae/[name@X/[(#name(0,3)), "サザエ"],  
husband(#masuo)],  
person#masuo/[name@Y/"フグ田マスオ"#name,  
wife(#sazae)]]
```

```
family(set)/person#sazae/name@X/[(#name(0,3)), "サザエ"]  
family(set)/person#sazae/husband(#masuo)  
family(set)/person#masuo/name@Y/"フグ田マスオ"#name  
family(set)/person#masuo/wife(#sazae)
```

# 問合せ表現

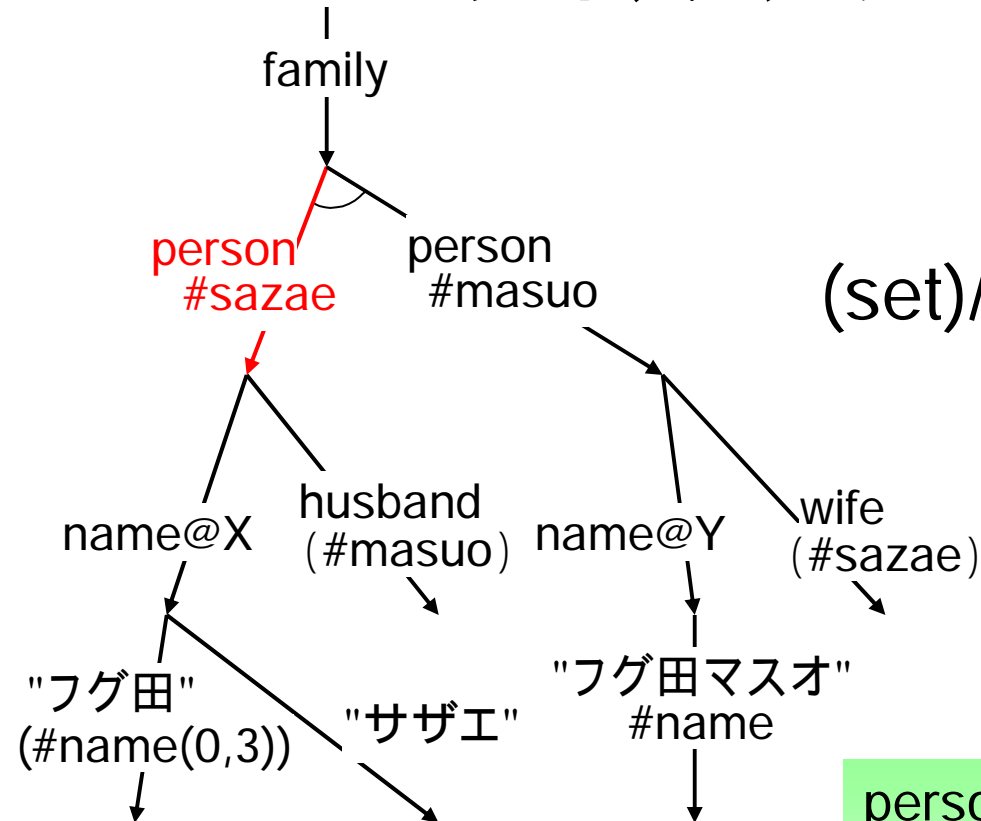
- ルール  
変更のパス式  $\leq$  フィルタリングパス式
- 問合せ  
?- パス式(ルールのヘッドに用いたパス式)
- フィルタリングパス式
  - パス
  - 変数(項の集合を束縛)
  - 正規表現

# フィルタリングパス式



親を持つ。  
かつ、子を持つ。  
かつ、nameというラベルである項  
の集合を変数\$Aに束縛する。

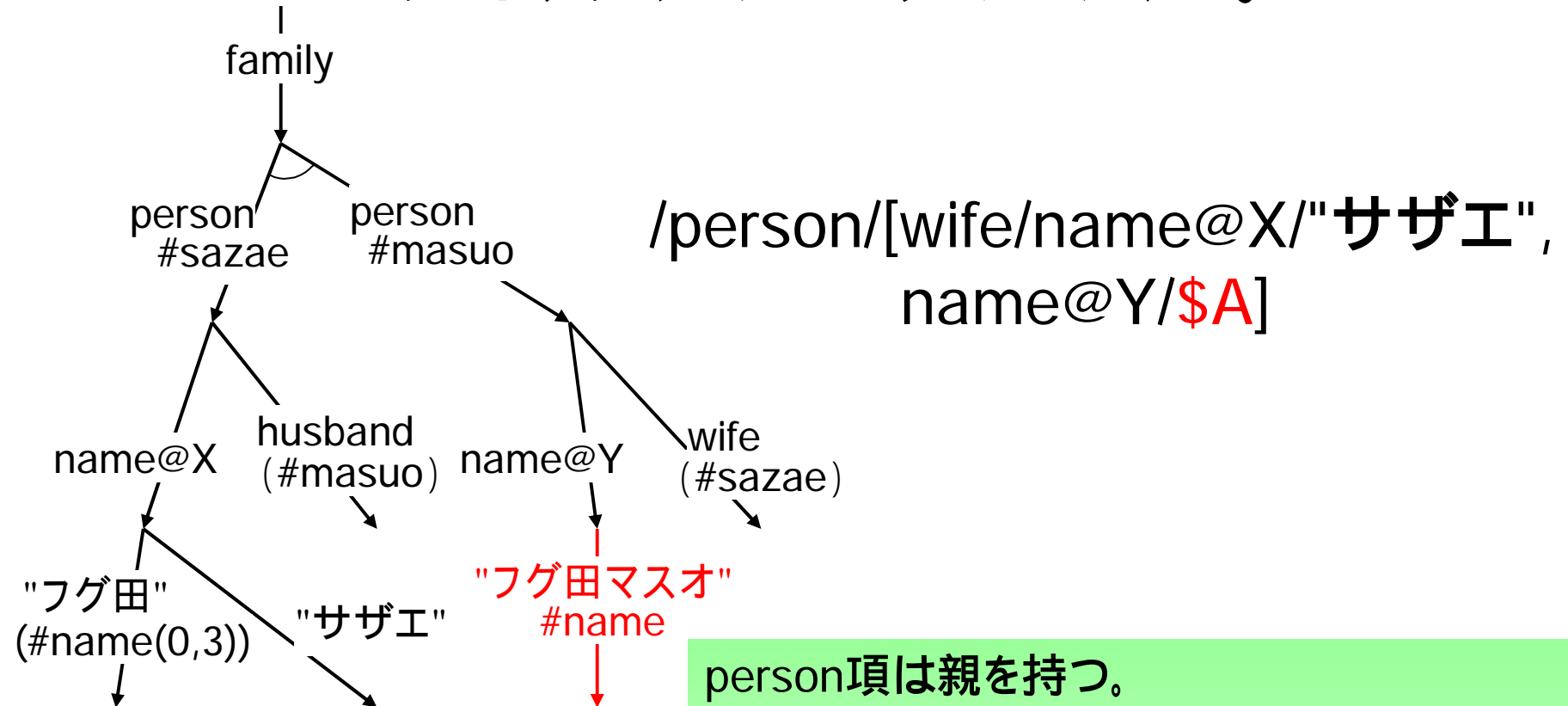
# フィルタリングパス式



(set)/person=**\$A**/name/"サザエ"

person項は集合の関係にある。  
かつ、person項はname項の子を持つ。  
かつ、name項は"サザエ"項の子を持つ  
person項の集合を変数**\$A**に束縛する。

# フィルタリングパス式



person項は親を持つ。  
かつ、person項はwife項の子を持つ。  
かつ、person項はname@Y項の子を持つ。  
かつ、wife項はname@X項の子を持つ。  
かつ、name@X項は"サザエ"項の子を持つ。  
name@Y項の子の集合を変数Aに束縛。

# ルールと問合せ

変更のパス式  $\leq$  フィルタリングパス式  
?- ルールのヘッドに用いたパス式

?- 名前(set)/\$A

名前(set)/\$A  $\leq$  family/person/name/\$A

解

名前

"フグ田サザエ" "フグ田マスオ"

?- サザエの夫/\$B

サザエの夫/\$B

$\leq$  /person/[wife/name@X/"サザエ",  
name@Y/\$B]

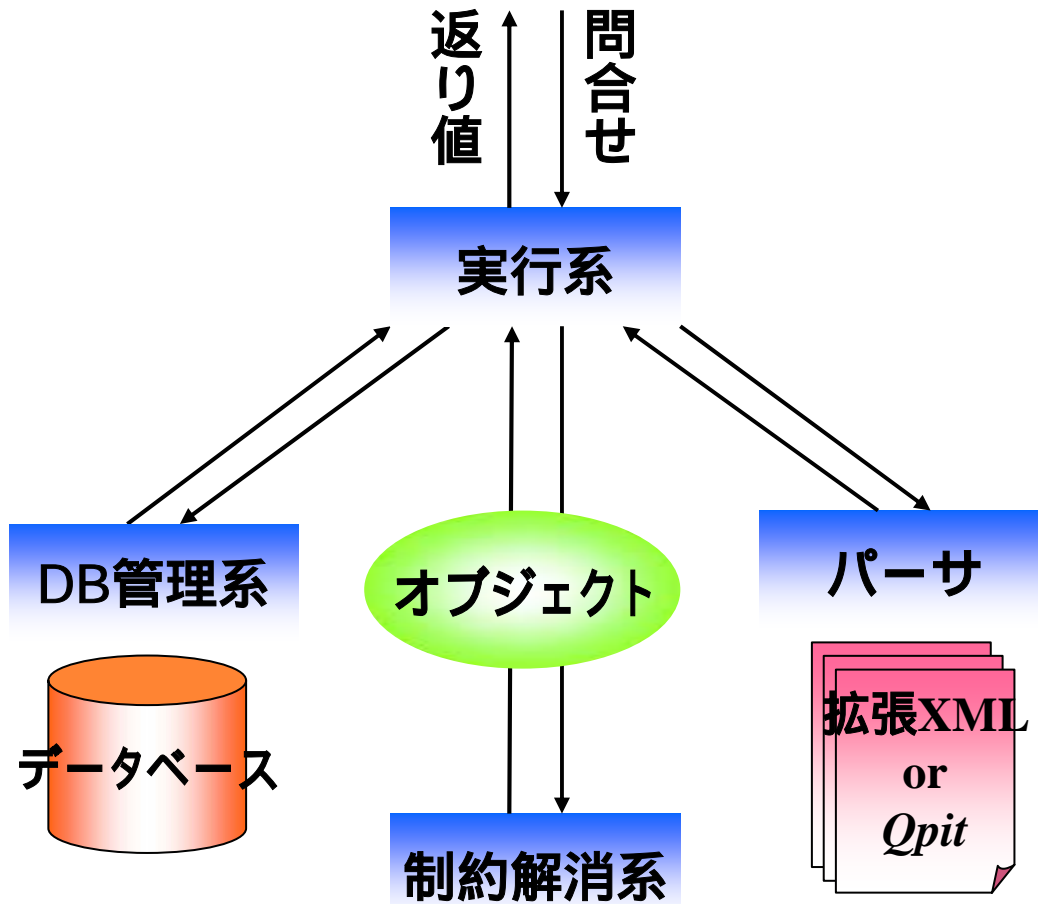
解

サザエの夫

"フグ田マスオ"

解はパスの集合

# Qpitシステム



- 実行系
  - 検索
  - 各処理系に命令
- パーサ
  - 構文解析
  - アンパース
- DB管理系
  - データ管理
  - 検索
  - 保存
- 制約解消系
  - 制約解消

# まとめ

## 目的

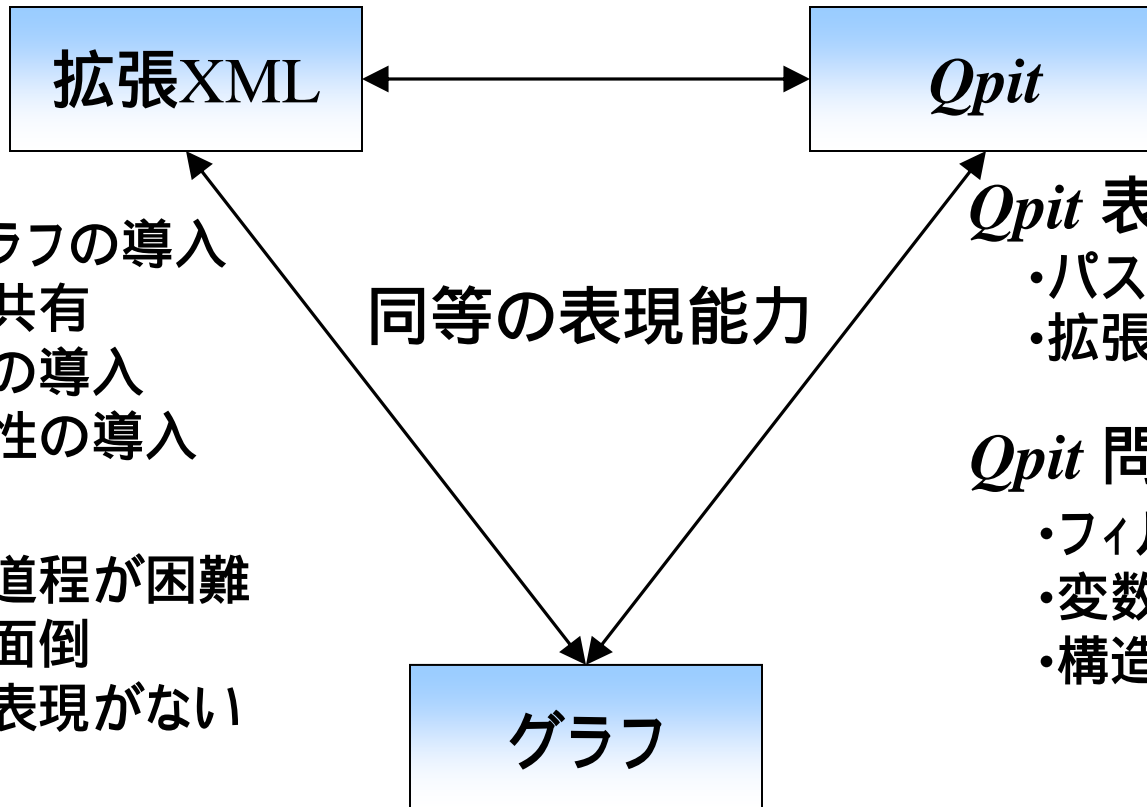
- 複合オブジェクトを表現
- 個別化情報の表現
- マルチメディア情報の統合

 XMLの木構造を超えている

XMLの拡張



# まとめ



制約解消した結果、正規化されたグラフが意味となる

## 拡張

- 有向グラフの導入
- 情報の共有
- 構成子の導入
- 拡張属性の導入

## 問題点

- 意味の道程が困難
- 記述が面倒
- 問合せ表現がない

*Qpit*

## *Qpit* 表現

- パスによる高次な記述
- 拡張XMLの意味を反映

## *Qpit* 問合せ表現

- フィルタリング表現
- 変数と正規表現の導入
- 構造の変更

グラフ

# 今後の課題

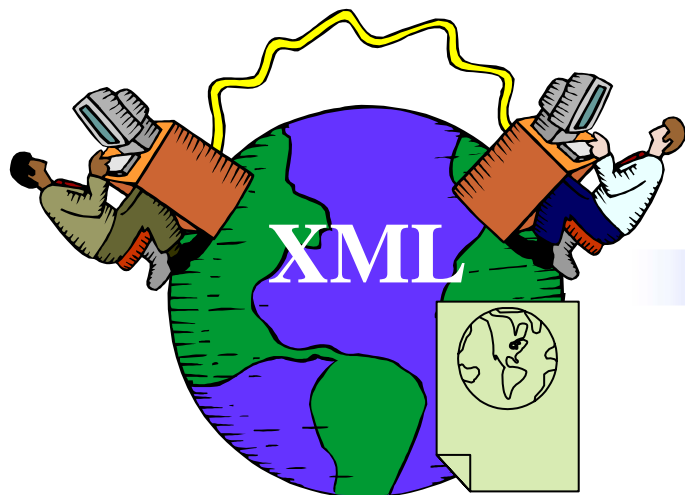
- パタンの順序や項の評価結果
- 問合せ機能の強化
  - より複雑な変更
  - 情報の更新、追加、削除
  - 複数のルール
- XML1.0への完全な準拠
- *Qpit*APIの機能強化
- コンテンツデータベースの実現

# 意味的に拡張したXMLと それに対する問合せ機構を持つシステムの実現

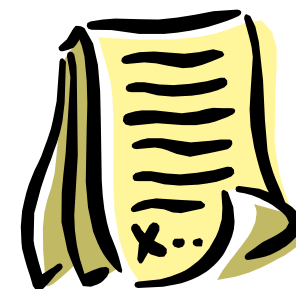
言語・ソフトウェア工学研究室  
的野晃整

# はじめに

## 情報交換



## XMLの拡張



## Qpit言語

- 意味論を自由に与える
- 応用の拡大
- デファクトスタンダード
- 木構造
- 型がない

- 有向グラフ
- 集合を表現
- 情報の共有
- 属性の拡張
- XMLに準拠
- 記述が低次元
- 問合せ機構がない

- 拡張の意味論を反映
- 高レベルな記述
- パスによる表現
- 情報のフィルタリング
- 問合せ能力

# XMLの拡張

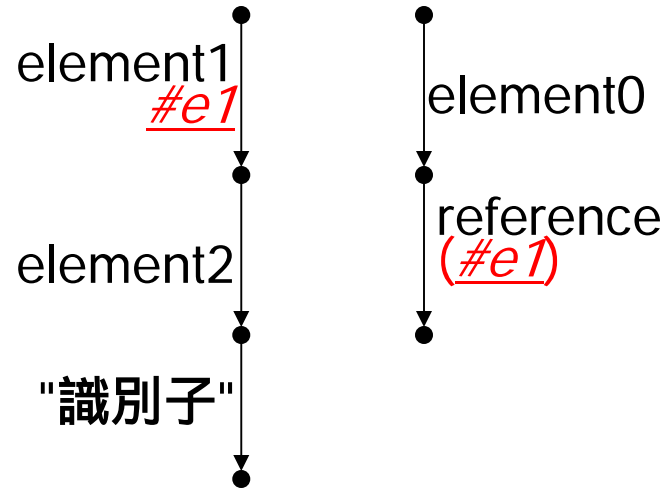
## 要件

- マルチメディア情報の統合
- 情報の共有化
- 複合オブジェクトの表現
  
- 有向グラフ…………… 識別子、等値制約の導入
- 集合を表現…………… 構成子の導入
- 部分情報の共有……… 部分文字列、意味的な親の導入
- 属性の拡張…………… 拡張属性の導入

# 有向グラフ (識別子)

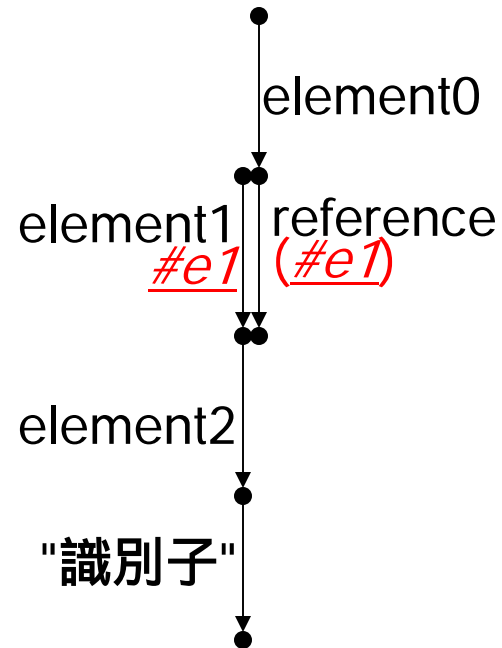
```
<element1 id="e1">  
  <element2>  
    識別子  
  </element2>  
</element1>
```

```
<element0>  
  <reference idref="e1"/>  
</element0>
```



# 有向グラフ (識別子)

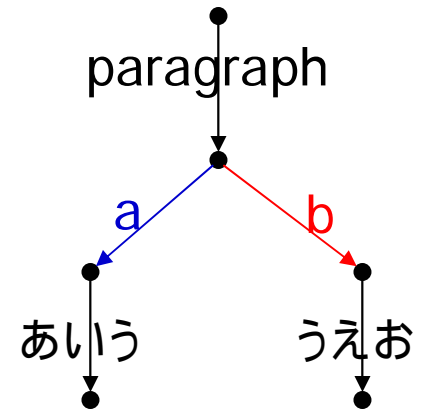
```
<element1 id="e1">  
  <element2>  
    識別子  
  </element2>  
</element1 >  
  
<element0>  
  <reference idref="e1"/>  
</element0>
```



# 部分情報の共有

意味

<paragraph>あいうえお</paragraph>  
          a      b



~~<paragraph><a>あい<b>う</a>えお</b></paragraph>~~



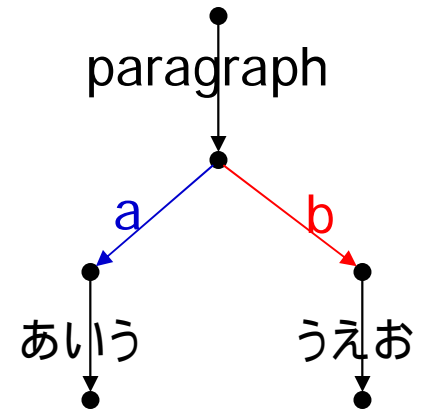
非整形式



# 部分情報の共有

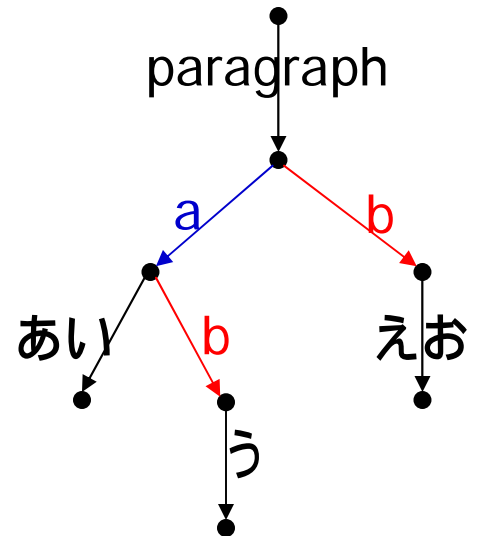
意味

<paragraph>あいうえお</paragraph>  
a b



はじめのbがaの子になっている

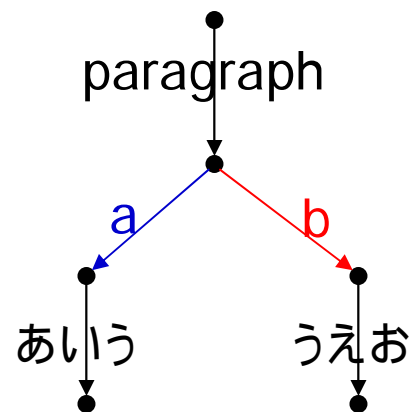
<paragraph>  
<a>あい<b>う</b></a><b>えお</b>  
</paragraph>



# 部分情報の共有 (親要素の指定)

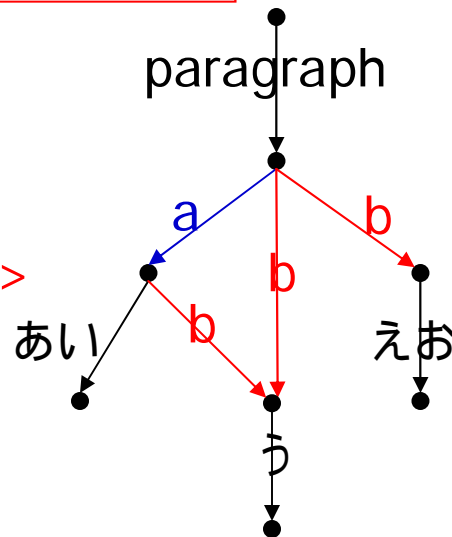
意味

<paragraph>あいうえお</paragraph>  
          a      b



情報量の増加とともに複雑化

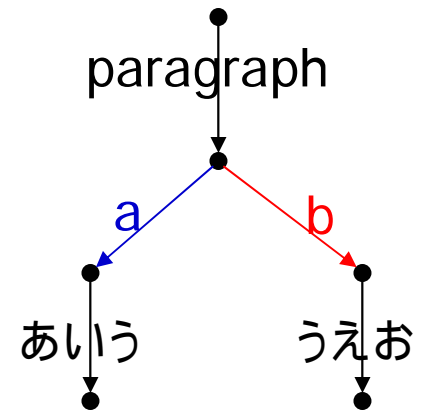
<paragraph id="000">  
  <a>あい<b parents="#000">う</b></a><b>えお</b>  
</paragraph>



# 部分情報の共有

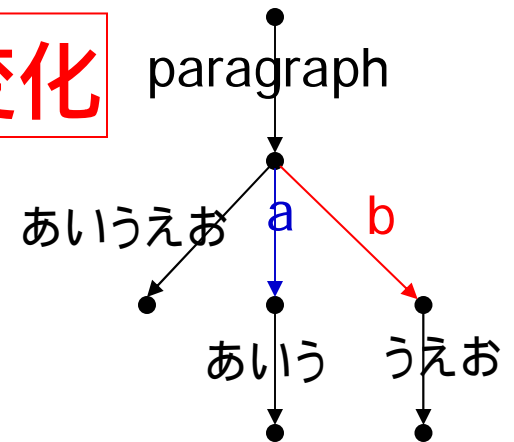
## 意味

<paragraph>あいうえお</paragraph>  
          a      b



**共有が増えると識別子名が変化**

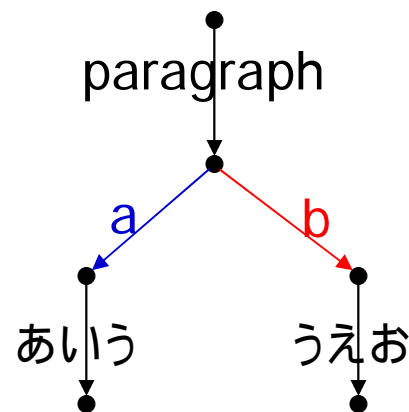
```
<paragraph xmlns:q="Qpit">  
  <q:Id id="000">あい</q:Id>  
  <q:Id id="001">う</q:Id>  
  <q:Id id="002">えお</q:Id>  
  <a><q:Reference idref="000"/><q:Reference idref="001"/></a>  
  <b><q:Reference idref="001"/><q:Reference idref="002"/></b>  
</paragraph>
```



# 部分情報の共有 (部分文字列参照)

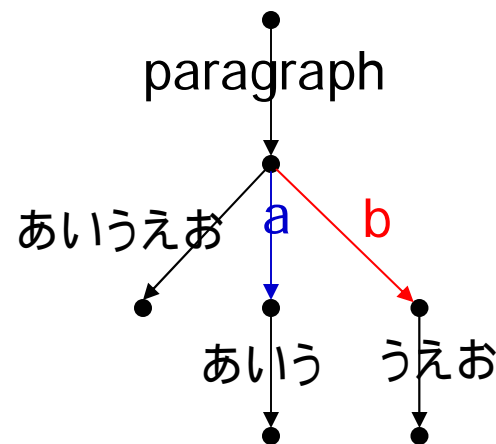
意味

<paragraph>あいうえお</paragraph>  
          a      b



文字数を数える必要がある

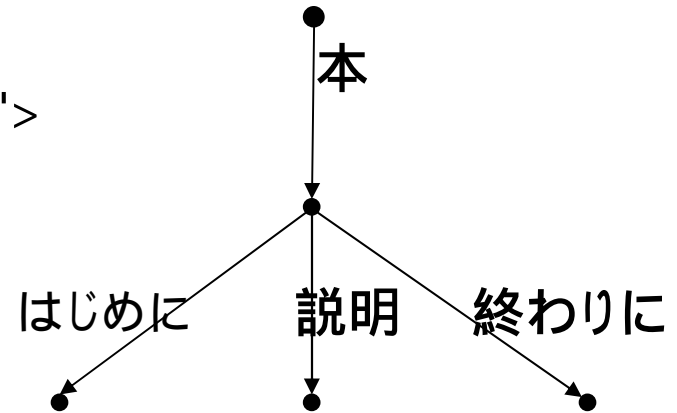
```
<paragraph xmlns:q="Qpit">
  <q:Pcdata id="003">あいうえお</q:Pcdata>
  <a><q:Reference substring="#003(0,3)"/></a>
  <b><q:Reference substring="#003(2,5)"/></b>
</paragraph>
```



# 構成子の導入

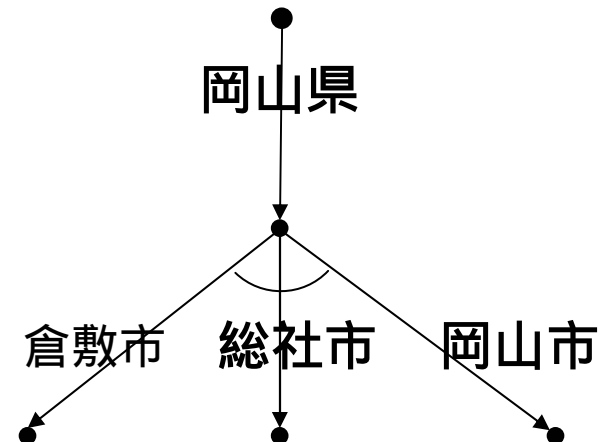
- 列 (文書系で多用)

```
<本 constructor="sequence">  
  <はじめに/>  
  <説明/>  
  <終わりに/>  
</本>
```



- 集合 (データ系で多用)

```
<岡山県 constructor="set">  
  <倉敷市/>  
  <総社市/>  
  <岡山市/>  
</岡山県>
```



# 拡張属性

## 同名の属性



### 非整形式

<鬼ノ城 季節="春" 季節="夏"  
季節="秋" 季節="冬"/>

拡張属性: <鬼ノ城 condition=" 季節='春' 季節='夏'  
季節='秋' 季節='冬' "/>

## 不等式

意味: <成人 年齢="20歳以上"/>

拡張属性: <成人 condition="年齢>'20歳' "/>

# グラフ

グラフ変数 = ( 識別子, 要素名, { 属性 }, { 部分要素 } )

- グラフの連立方程式の解がその情報の意味
- 拡張XMLはグラフのシンタックスシュガー

## 拡張XML

```
<person id="matono">  
  <name>的野</name>  
  <tel>090-xxxx-xxxx</tel>  
</person>
```

## グラフ表現

```
g0=(matono, person, {}, {g1, g3})  
g1=(_, name, {}, {g2})  
g2=(_, pcddata, {}, {"的野"})  
g3=(_, tel, {}, {g4})  
g4=(_, pcddata, {}, {"090-xxxx-xxxx"})
```

# *Qpit*表現

- 拡張項、パスによる表現
- アドレッシング、フィルタリング
- 変数と正規表現の導入
- 問合せ表現
- 情報の操作

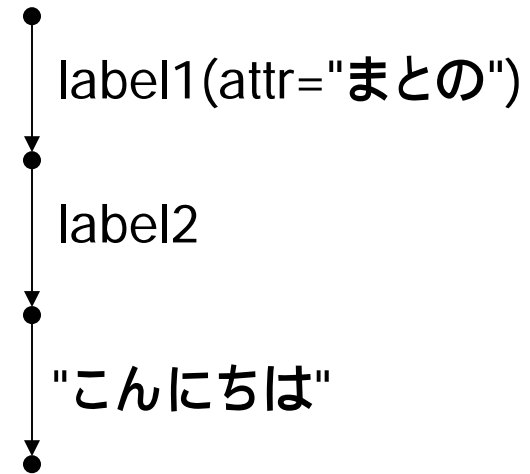


# Qpit表現 (要素の連結)

## 拡張XML

```
<label1 attr="まとの">  
  <label2>  
    こんにちは  
  </label2>  
</label1>
```

## モデル



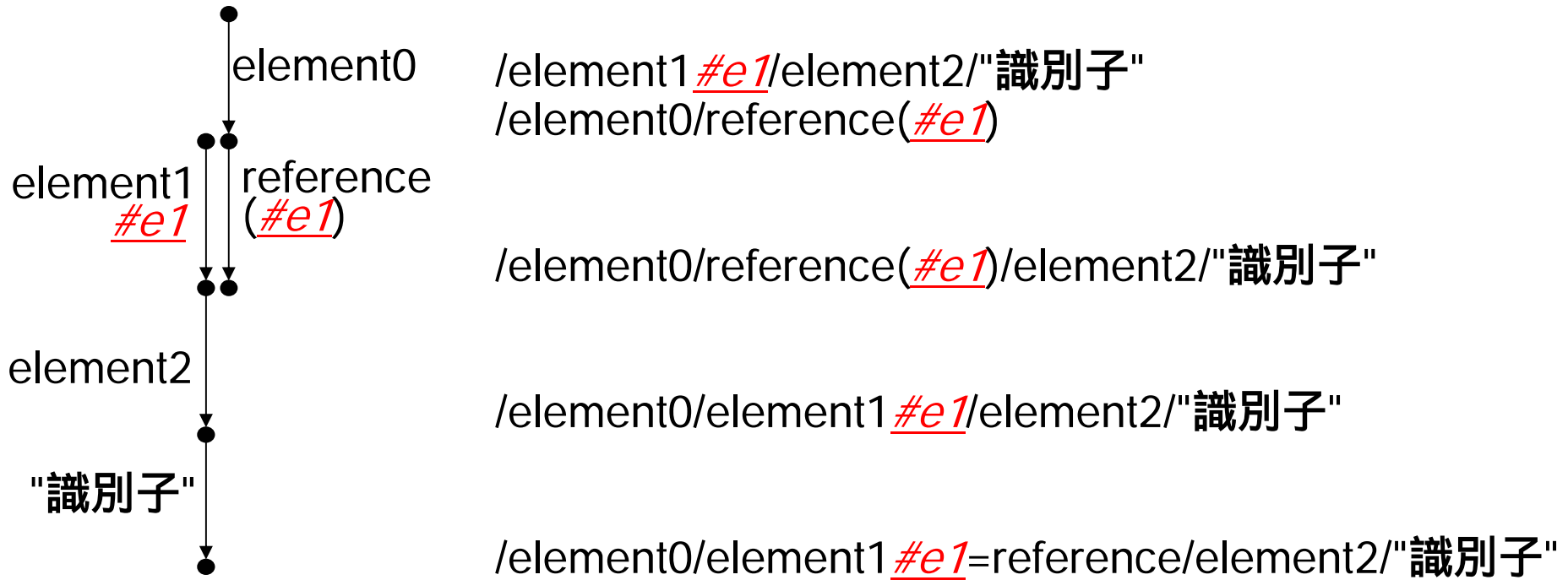
## Qpit表現

`/label1 (attr="まとの")/label2/"こんにちは"`

拡張項

パス

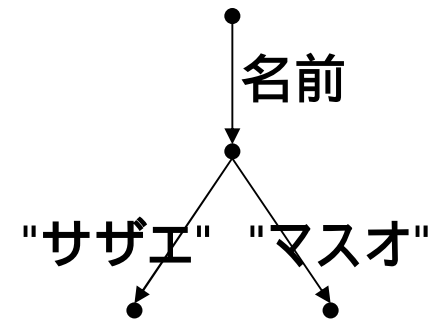
# Qpit表現 (識別子)



# 問合せによる情報の操作

変更情報 <= 検索のためのフィルタリングパス

名前/\$X <= /family/person/name/\$X



名前/[\$A, \$B] <= /family/[ person\$Y/husband\$Z/name/\$A,  
person\$Z/wife\$Y/ name\$B]

# 意味的に拡張したXMLと それに対する問合せ機構を持つシステムの実現

言語・ソフトウェア工学研究室  
的野晃整

# サザエ夫婦の例 (*Qpit*)

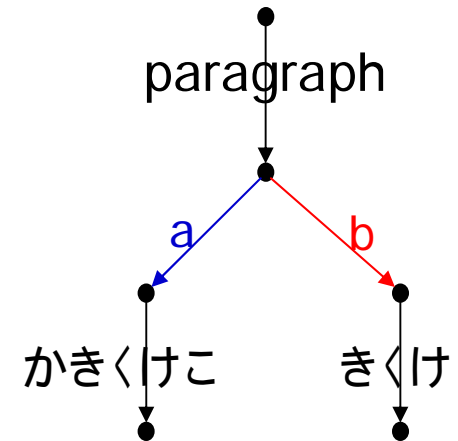
```
<family xmlns:q="Qpit">
  <person q:id="sazae">
    <name>サザエ</name>
    <husband q:idref="masuo"/>
  </person>
  <person q:id="masuo">
    <name>マスオ</name>
    <wife q:idref="sazae"/>
  </person>
</family>
```

```
/family/person#sazae/name/"サザエ"
/family/person#sazae/husband(#masuo)
/family/person#masuo/name/"マスオ"
/family/person#masuo/wife(#sazae)
```

```
/family/[person#sazae/[name/"サザエ",
                        husband(#masuo)],
          person#masuo/[name/"マスオ",
                        wife(#sazae)]]
```

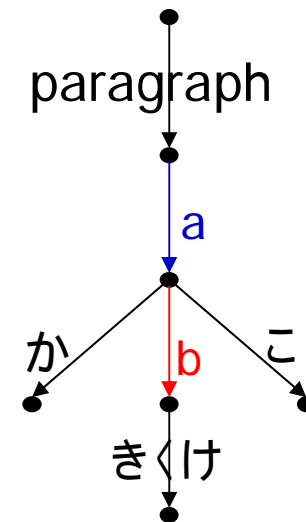
# 意味的な親の指定

<paragraph>かきくけこ</paragraph>



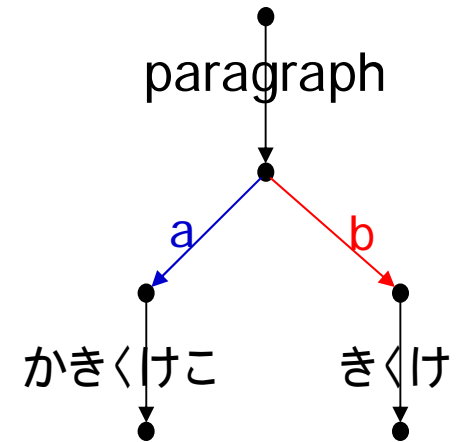
<paragraph>  
<a>か<b>きくけ</b>こ</a>  
<paragraph>

内容に矛盾している

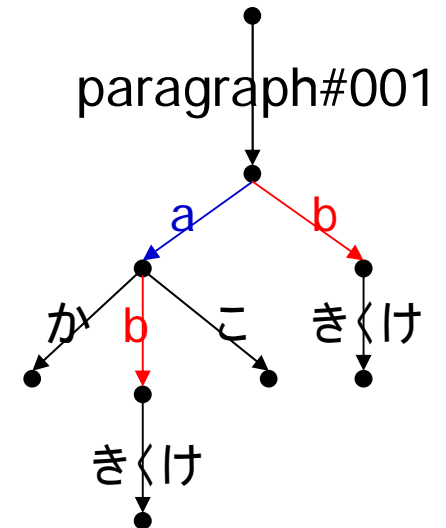


# 意味的な親の指定

<paragraph>かきくけこ</paragraph>



<paragraph q:id="001" xmlns:q="Qpit">  
 <a>か<b q:parents="#001">きくけ</b>こ</a>  
</paragraph>



# 識別子と等値制約の違い

- 識別子 …… 確定している要素のみ
- 等値制約 …… 確定していない要素も



# グラフ表現

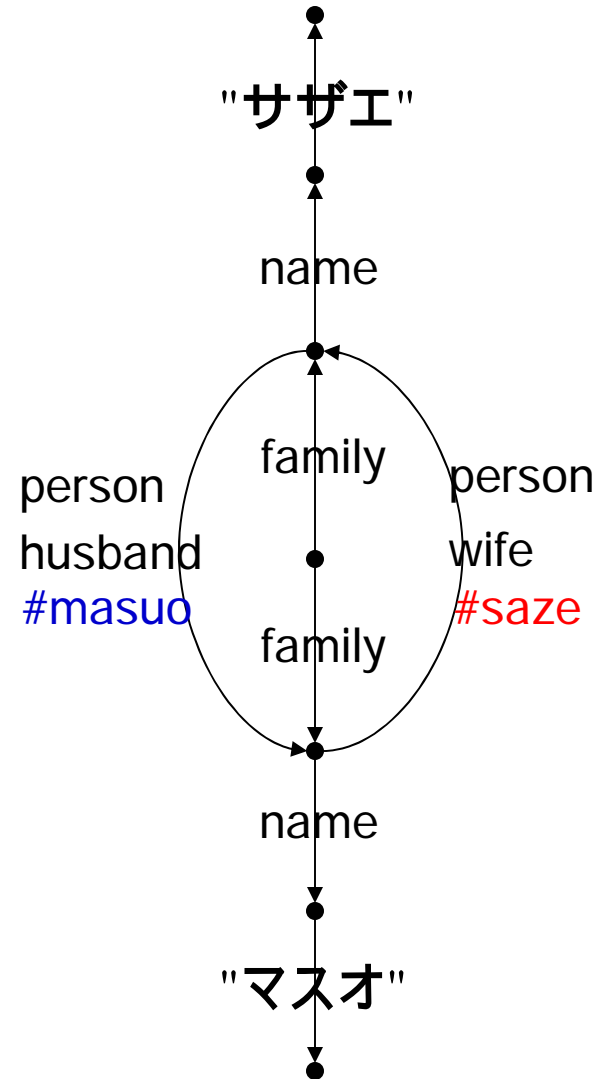
$g=(id, label, \{attributes\}, \{children\})$

```
<family xmlns:q="Qpit">
  <person q:id="sazae">
    <name>サザエ</name>
    <husband q:idref="masuo"/>
  </person>
  <person q:id="masuo">
    <name>マスオ</name>
    <wife q:idref="sazae"/>
  </person>
</family>
```

```
g0=(_, family, {xmlns:q="Qpit"}, {g1, g5})
g1=(sazae, person, {}, {g2, g4})
g2=(_, name, {}, {g3})
g3=(_, q:Pcdata, {}, {"サザエ"})
g4=(, husband, {q:idref="masuo"}, {})
g5=(masuo, person, {}, {g6, g8})
g6=(, name, {}, {g7})
g7=(, q:Pcdata, {}, {"マスオ"})
g8=(, wife, {q:idref="sazae"}, {})
```

# 有向グラフを表現

```
<family xmlns:q="Qpit">
  <person q:id="sazae">
    <name>サザエ</name>
    <husband q:idref="masuo"/>
  </person>
  <person q:id="masuo">
    <name>マスオ</name>
    <wife q:idref="sazae"/>
  </person>
</family>
```

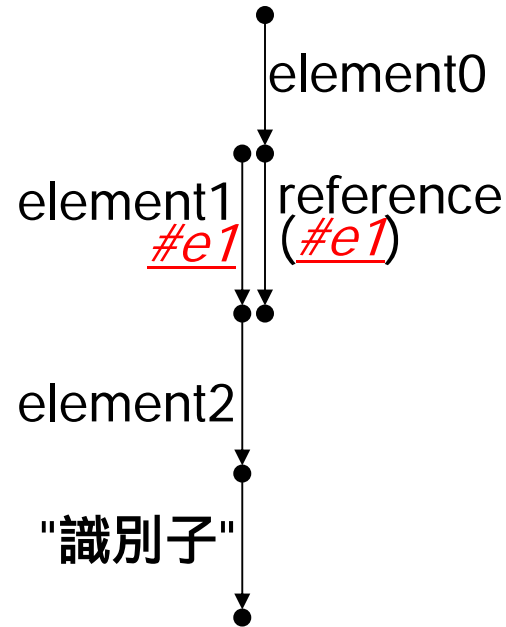


# Qpit表現 (識別子)

## 拡張XML

```
<element1 id="e1">
  <element2>
    識別子
  </element2>
</element1>

<element0>
  <reference idref="e1"/>
</element0>
```



## Qpit表現

- /element1 #e1/element2/"識別子"  
/element0/reference(#e1)
- /element0/reference(#e1)/element2/"識別子"
- /element0/element1 #e1/element2/"識別子"
- /element0/element1 #e1=reference/element2/"識別子"

# 拡張属性

## 同名の属性

意味: <鬼ノ城 季節="春" 季節="夏"  
季節="秋" 季節="冬"/>

非整形形式: <鬼ノ城 季節="春, 夏, 秋, 冬"/>

拡張属性: <鬼ノ城 condition=" 季節='春' 季節='夏'  
季節='秋' 季節='冬' "/>

## 不等式

意味: <成人 年齢="20歳以上"/>

非整形形式: <成人 年齢>"20歳"/>

拡張属性: <成人 condition="年齢>'20歳' "/>

# フィルタリングパス式

```
<family xmlns:q="Qpit">  
  <person q:id="sazae">  
    <name>サザエ</name>  
    <husband q:idref="masuo"/>  
  </person>  
  <person q:id="masuo">  
    <name>マスオ</name>  
    <wife q:idref="sazae"/>  
  </person>  
</family>
```

family/person#sazae/name\$X/

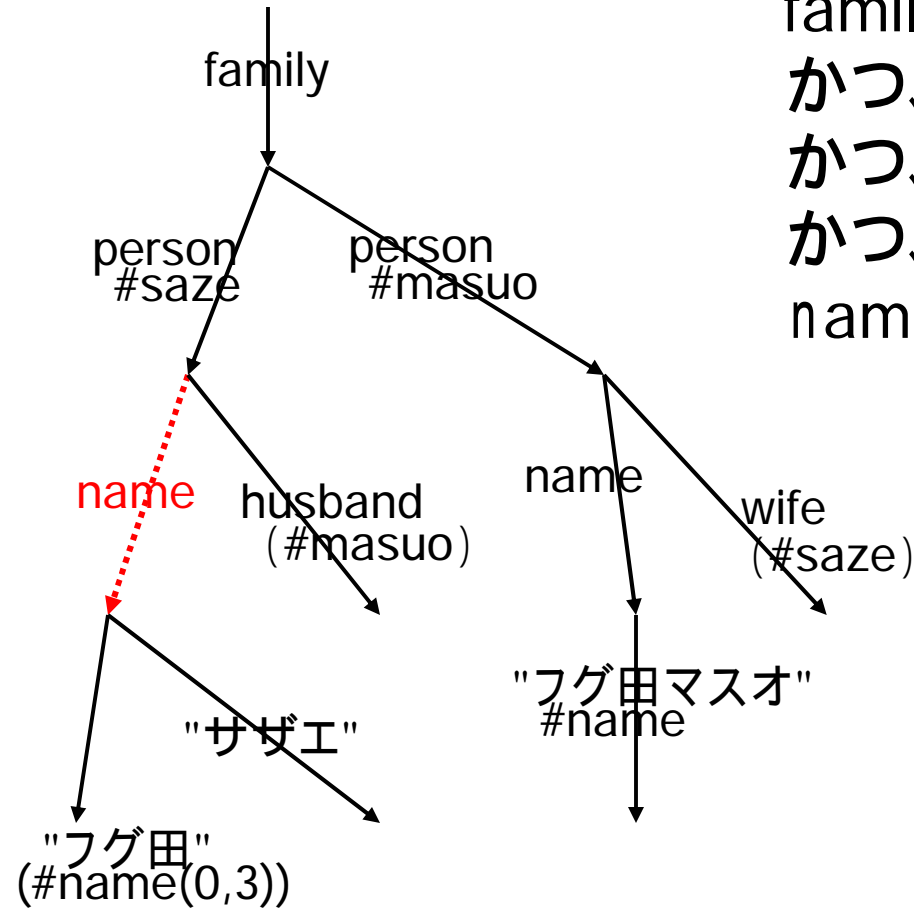
family/person/name\$X/

family/person\$X/name/"サザエ"

!\*/person\$X/!\*/"サザエ"

# フィルタリングパス式

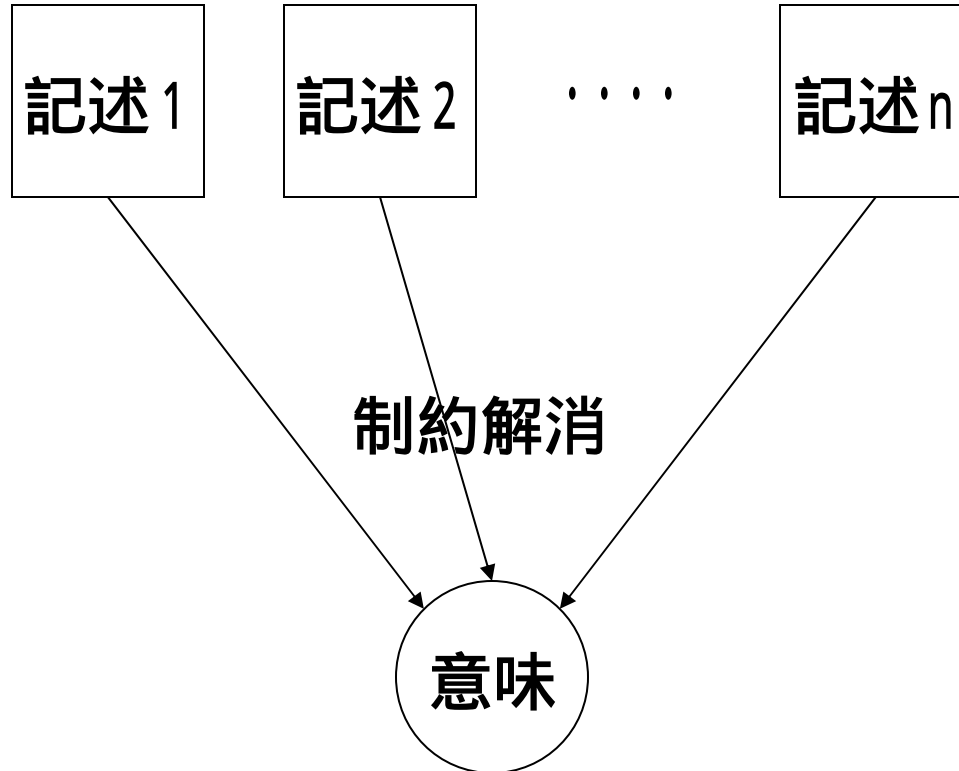
family項は親を持たない、  
かつ、family項はperson項の子を持つ  
かつ、person項の識別子はsazaeである  
かつ、person項はname項の子を持つ  
name項を変数\$Xに束縛



family/person#sazae/name\$X/

# グラフ

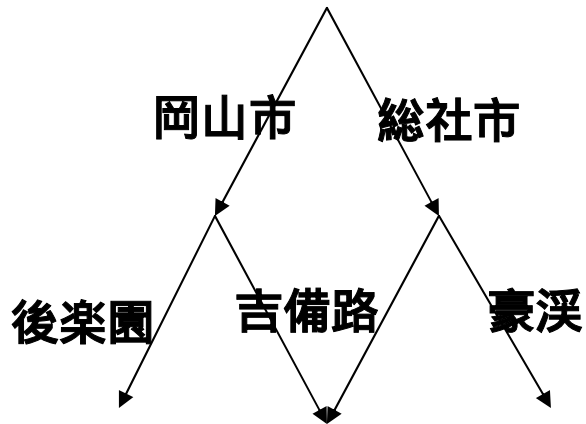
グラフ変数 = ( 識別子, 要素名, { 属性 }, { 要素値 } )



# グラフ表現



# グラフ



$g_0 = (, \text{岡山市}, \{\}, \{g_2, g_3\})$

$g_1 = (, \text{総社市}, \{\}, \{g_3, g_4\})$

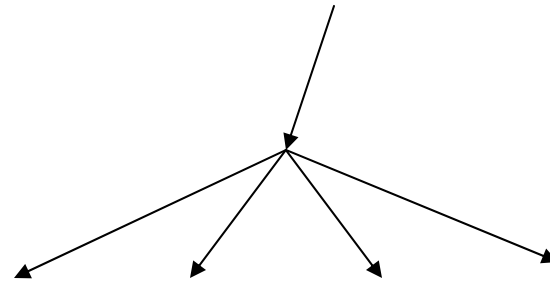
$g_2 = (, \text{後楽園}, \{\}, \{g\})$

$g_3 = (, \text{吉備路}, \{\}, \{g\})$

$g_4 = (, \text{豪溪}, \{\}, \{g\})$

# モデル

```
<県 name="岡山" xmlns:q="Qpit">
  <市 name="岡山">
    <観光地 name="造山古墳" q:id="tukuri"/>
    <観光地 name="岡山後楽園"/>
    <観光地 name="吉備津神社" q:id="kibitu"/>
  </市>
  <市 name="総社">
    <観光地 name="豪溪"/>
    <観光地 name="鬼ノ城" q:id="kinojo"/>
    <観光地 name="岩屋" q:id="iwaya"/>
    <観光地 name="備中国分寺" q:id="koku"/>
  </市>
  <観光地 name="吉備路">
    <q:Ref idref="tukuri"/>
    <q:Ref idref="kibitu"/>
    <q:Ref idref="kinojo"/>
    <q:Ref idref="iwaya"/>
    <q:Ref idref="koku"/>
  </観光地/>
</県>
```



県 (name="岡山")  
市 (name="総社")  
市 (name="岡山")  
市 (name="倉敷")

# まとめ

- XMLの拡張
  - 有向グラフ
  - 集合を表現
  - 部分情報の共有
  - 属性の拡張
- *Qpit*言語
  - パスによる表現
  - 問合せ
    - フィルタリング
    - 変数と正規表現
    - 構造の変更