



chapter

# 3

## データの取得と処理

本章では、XSLT を使用したカスタム Web 公開で利用できる XSLT スタイルシートの記述の仕方の基本を説明します。あわせて、XSLT スタイルシートの処理に必要な XML データを要求する際に使用するクエリーコマンドについても紹介します。

## 3.1 Web公開エンジンの情報の取得

FileMaker ServerのカスタムWeb公開では、適切なクエリーコマンドとクエリー引数を使用してHTTPリクエストを送信すると、FileMakerのデータベースからデータを取得したり、データベースサーバーやWeb公開エンジンに対して任意の処理を実行させることができます。カスタムWeb公開で使用されるクエリーコマンドはFileMaker固有のものであり、一般的なデータベースの操作に用いられるSQL(Structured Query Language)のように汎用的なものではないという点をまず認識する必要があります。

### 03

#### データの取得と処理

### 3.1.1 ● Web公開エンジンの製品情報を表示する

データベースと連動する方法を解説する前に、XSLTを使用したカスタムWeb公開で利用できるXSLTスタイルシートの記述の仕方の、基本を説明します。まずはカスタムWeb公開エンジンがWeb公開コアから取得したXMLデータを表示するXSLTスタイルシートのサンプルから見ていきましょう。最初のサンプルでは、Web公開エンジンの製品情報が含まれたXMLデータを取得する-processクエリーコマンドを利用します。

#### 【サンプル1】

```
<?xml version="1.0" encoding="UTF-8"?>
<!-- 1行目には必ずXML宣言を記述し、この文書がXMLドキュメントであることを示します -->
<!-- encoding属性でこの文書の文字コードを指定します -->

<!-- XSLTネームスペースの宣言 -->
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
                exclude-result-prefixes="xsl fmrs"
                version="1.0">

  <!-- FileMaker固有の<?xslt-cwp-query?>処理命令でクエリーコマンドおよびクエリー引数を
  指定できます -->
  <!-- -processクエリーコマンドを使用し、-grammarクエリー引数でfmresultset XML文法を
  利用することを指定します -->
  <?xslt-cwp-query params="-grammar=fmresultset&-process"?>

  <!-- <xsl:output>要素のmethod属性を使用して出力形式の種類を指定します -->
  <xsl:output method="xml" />

  <!-- <xsl:template>要素でテンプレートルールを指定し、出力される内容を調整します -->
  <xsl:template match="/">
```

```

<!-- <xsl:copy-of> エlementを使用するとソースとなる XML データの任意の部分をそのまま利用
することができ、要求された XML データを変換せずに出力します -->
<xsl:copy-of select="." />

</xsl:template>
</xsl:stylesheet>

```

FileMakerのカスタムWeb公開で使用するXSLTスタイルシートファイルについては、Web公開エンジンが動作しているコンピュータにある「xslt-template-files」フォルダの中にコピーします。xslt-template-filesフォルダは下記の場所にあります。

#### ○ Windows の場合

```
C:\Program Files\FileMaker\FileMaker Server\Web Publishing\xslt-
template-files
```

#### ○ Mac の場合

```
/ライブラリ/FileMaker Server/Web Publishing/xslt-template-files
```

XSLTスタイルシートのファイル名が「sample001.xml」であり、そのファイルがxslt-template-filesフォルダの直下に配置されている場合、Webブラウザから下記のようなURLでアクセスすると、結果を表示できます(www.example.comの部分は自分のPC環境に応じて置き換えてください。本書では以下同様に、サンプルに用いるURLにはwww.example.comを使用します)。

#### ○ 指定する URL

```
http://www.example.com/fmi/xsl/sample001.xml
```

Webブラウザによっては、出力結果を確認するためにソースを表示しなければならない場合があります。例えば、MacのSafariを使用している場合には、[表示]メニューの[ソースを表示]を選ぶことで出力結果を確認できます。

#### 【サンプル 1 の出力結果 (改行して表示を調整)】

```

<?xml version="1.0" encoding="UTF-8" ?>
<fmresultset xmlns="http://www.filemaker.com/xml/fmresultset" version=
"1.0">
  <error code="0" />
  <product build="01/12/2008" name="FileMaker Web Publishing Engine"
version="9.0.3.316" />
  <datasource database="PRODUCTINFO" date-format="" layout="" table=""

```

```

time-format="" timestamp-format="" total-count="1" />
  <metadata>
    <field-definition auto-enter="no" four-digit-year="no" global="no"
max-repeat="1" name="MAX_CONNECTIONS" not-empty="yes" numeric-
only="no" result="number" time-of-day="no" type="normal"/>
  </metadata>
  <resultset count="1" fetch-size="1">
    <record mod-id="0" record-id="0">
      <field name="MAX_CONNECTIONS">
        <data>FileMaker Web Publishing Engine</data>
      </field>
    </record>
  </resultset>
</fmresultset>

```

XSLTスタイルシートの中身を見ていきます。XSLTスタイルシートはそれ自身がXMLドキュメントでもあるため、1行目には必ず下記のようなXML宣言を記述して、その文書がXMLドキュメントであることを示す必要があります。

```
<?xml version="1.0" encoding="UTF-8"?>
```

サンプルではencoding属性でこの文書の文字エンコーディングを「UTF-8」に指定していますが、XSLTスタイルシートの文字コードがUTF-8ではなく「ISO-2022-JP」や「Shift\_JIS」の場合には、それぞれ対応した文字エンコーディングを指定する必要があります。XMLの仕様ではUTF-8を用いることが推奨されており、本書のサンプルでもUTF-8を指定しています。

**注意**

XML宣言のencoding属性で指定している文字エンコーディングと、XSLTスタイルシートで実際に使われている文字エンコーディングが一致しない場合には、エラーが発生する場合があります。

XML宣言の後は、<xsl:stylesheet>～</xsl:stylesheet>で全体を囲みます。<xsl:stylesheet>エレメント(要素)ではXSLTネームスペース(名前空間)の宣言を記述する必要がありますが、XSLTを使用したカスタムWeb公開では少なくとも下記の内容を記述します。

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
exclude-result-prefixes="xsl fmrs"
version="1.0">

```

XSLTスタイルシートの内容によっては、XSLTネームスペースの宣言の内容を変更する必要がありますが出てくる場合があります。FileMaker XSLT拡張関数を使用したり、データベースのレイアウト情報にアクセスする場合などには、宣言内容を一部追加および修正する必要があります。例えば、FileMaker XSLT拡張関数を使用するときには、XSLTネームスペースの宣言は次のように「xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"」を追加し、exclude-result-prefixes属性の値の最後に1バイトのスペースと文字列「fmxslt」を追加します。

```
<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
                xmlns:fmrs="http://www.filemaker.com/xml/fmresultset"
                xmlns:fmxslt="xalan://com.fmi.xslt.ExtensionFunctions"
                exclude-result-prefixes="xsl fmrs fmxslt"
                version="1.0">
```

次に、サンプルではFileMaker固有の<?xslt-cwp-query?>処理命令を記述しています。この処理命令はオプションで必ずしも利用する必要はありませんが、使用する場合にはXSLTスタイルシートの先頭部分で使う必要があります。

```
<?xslt-cwp-query params="-grammar=fmresultset&-process"?>
```

<?xslt-cwp-query?>処理命令を使うと、XMLデータの要求時に使用するFileMakerクエリー文字列をあらかじめ定義しておくことができます。ここで指定した内容は、URLで指定されたFileMakerクエリー文字列内のクエリーコマンドやクエリー引数よりも優先され、外部からの不正使用を防止できるようになっています。さらに、URLでFileMakerクエリー文字列を指定する必要がなくなってURLを短くできるので、なるべくこの処理命令を利用することが推奨されます。

なお、サンプルのXSLTスタイルシートで<?xslt-cwp-query?>処理命令の記述を省略した場合、下記のようなURLでアクセスする必要があります。

```
http://www.example.com/fmi/xsl/sample001.xsl?-grammar=fmresultset&-process
```

カスタムWeb公開エンジンは、<?xslt-cwp-query?>処理命令で定義されたFileMakerクエリー文字列の内容と、URLで指定されたFileMakerクエリー文字列の内容に基づいて、XMLデータを取得するためのリクエストを内部的に生成します。

-processクエリーコマンドは、XSLTを使用したカスタムWeb公開でのみ利用できるコマンドであり、データベース処理が不要な場合に使用されます。このクエリーコマンドは-grammarクエリー引数を必要とします。-grammarクエリー引数はXML文法を指定するために利用され、