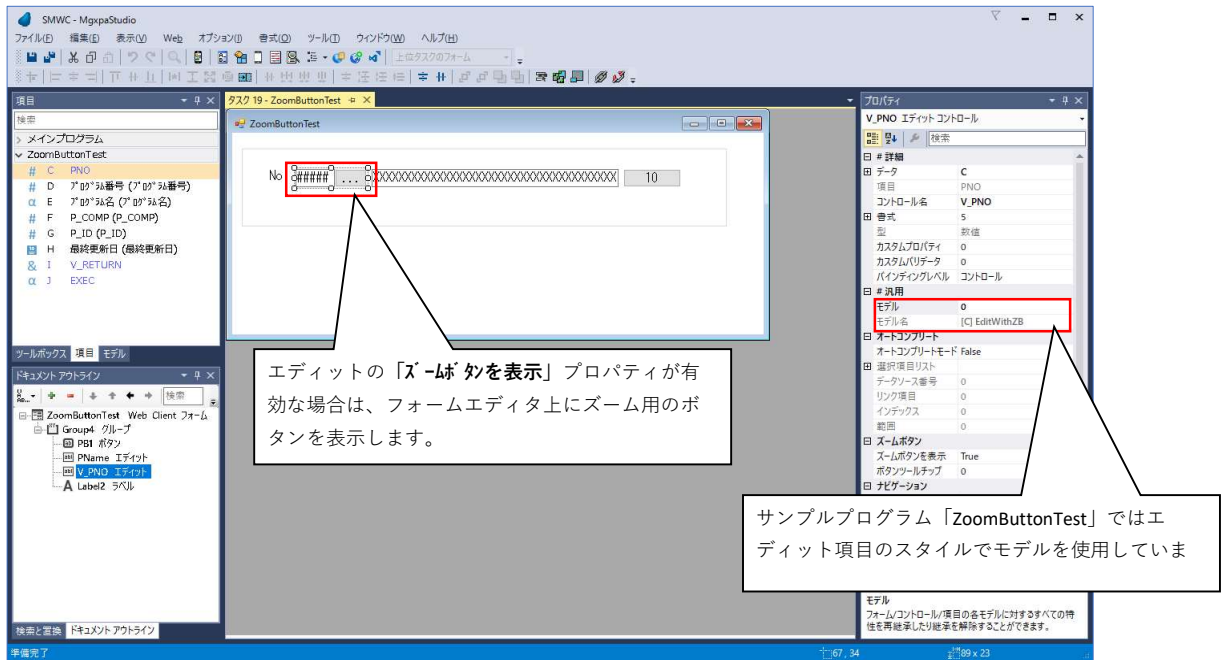


楽々ウェブクライアント/SMSYS Ver 1.01新機能の紹介

1. ズームボタン (ShowZoomButton) に対応

MAGIC xpa Ver 4.8で利用できるようになったエディットのズームボタン (ShowZoomButton) に対応しました。エディットの「ズームボタンを表示」プロパティが有効な場合は、Bootstrapの「input-group」クラスを使用し、ボタンをテキスト入力と一体化させます。更にGUI編集画面からボタンのデザインやアイコンを指定することも可能にしました。

① サンプルプログラムのフォーム



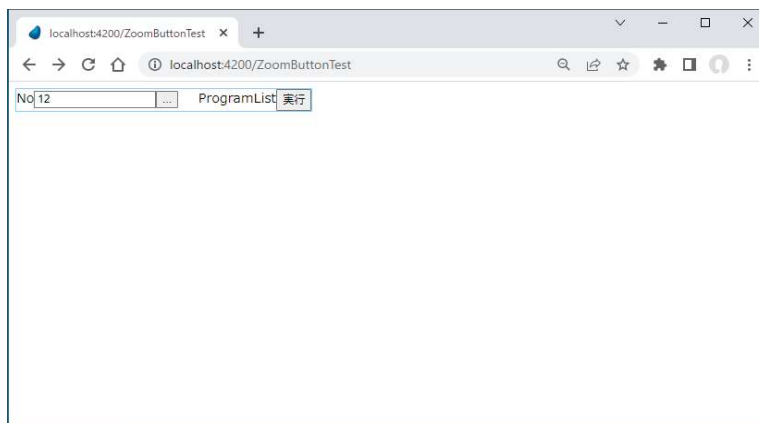
ズームボタンを表示する方法には幾つかの方法があります。

- ① フォームエディタ上でコントロールのプロパティを直接指定する
- ② フォームエディタ上でモデル (ズームボタンを表示) を指定する
- ③ データ項目の該当のスタイルでプロパティを直接指定する
- ④ データ項目の該当のスタイルでモデル (ズームボタンを表示) を指定する

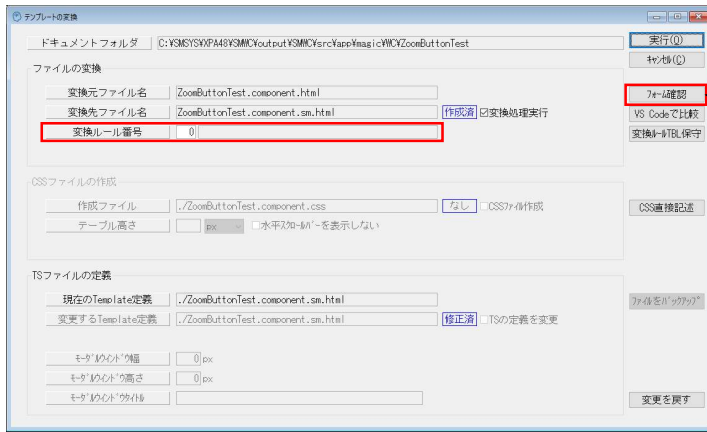
楽々ウェブクライアントでは上記の何れのパターンにも対応しました。

② 変換処理について

そのままサンプルを実行した場合のブラウザ表示結果は次のようになります。

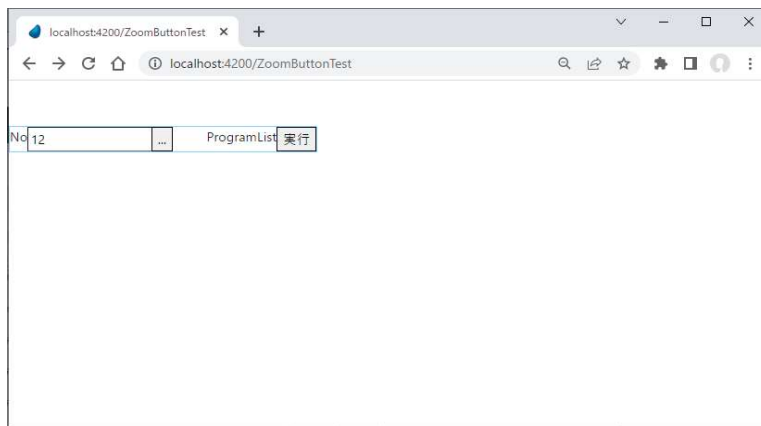


楽々ウェブクライアントで何も指定せずに変換してみます。



変換ルール番号や、フォーム確認からのコントロール属性を入力せずに変換します。

変換後の実行結果は次の様になりました。



ブラウザの表示結果だけでは、さして大きな違いは見られないように思えます。

HTMLを比較してみます。

```
14 <div style="display: flex; flex-direction: row">
15 <label
16   [magic]="mgc.Label2"
17   class="lable_overflow mgFont1"
18 >
19   No
20 </label>
21 </div>
22 <div>
23   <input
24     style="width: 70%"
25     [formControlName]="mgc.V_PNO"
26     type="number"
27     [magic]="mgc.V_PNO"
28     mgFormat
29   />
30   <mgError [magic]=mgc.V_PNO </mgError>
31   <button
32     style="min-width: auto; margin: auto"
33     [magic]="mgc.V_PNO"
34   >...</button>
35 </div>
36 <label [magic]="mgc.PName">
37   {{mg.getValue(mgc.PName) }}
38 </label>
39 <button [magic]="mgc.PB1">
40   {{mg.getValue(mgc.PB1)}}
41 </button>
42 </div>
43 </div>
```

MAGICが出力したHTML

```
1 <div novalidate [formGroup]="screenFormGroup">
2   <div style="display: flex; flex-direction: column" [magic]="mgc.ZoomButtonTest">
3     <div style="display: flex; flex-direction: row">
4       <div [magic]="mgc.Group4" class="container_border" style="display: flex; flex-direction: column">
5         <div style="display: flex; flex-direction: row">
6           <label [magic]="mgc.Label2" class="lable_overflow mgFont1">
7             No
8           </label>
9           <div>
10            <div class="input-group">
11              <input [formControlName]="mgc.V_PNO" type="number" [magic]="mgc.V_PNO" mgFormat />
12              <mgError [magic]=mgc.V_PNO>
13              </mgError>
14              <button type="button" [magic]="mgc.V_PNO">
15                ...
16              </button>
17            </div>
18          </div>
19          <label [magic]="mgc.PName">
20            {{mg.getValue(mgc.PName) }}
21          </label>
22          <button [magic]="mgc.PB1">
23            {{mg.getValue(mgc.PB1)}}
24          </button>
25        </div>
26      </div>
27    </div>
28  </div>
29 </div>
```

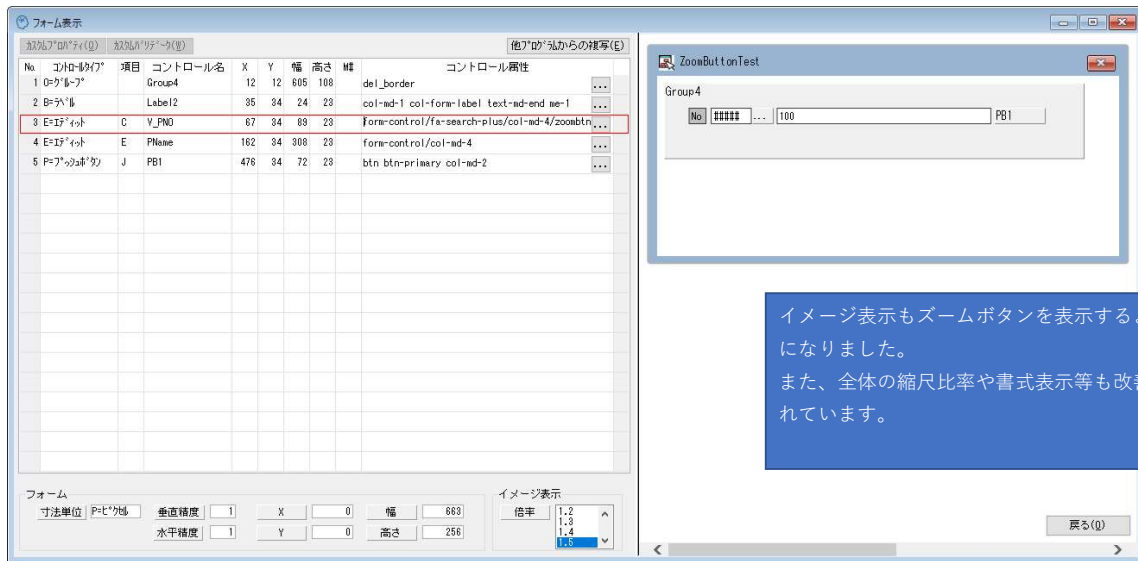
inputタグとbuttonタグを括るためのdiv要素を追加し、「input-group」クラスをセットしている

楽々ウェブクライアントが変換したHTML

このようにして、楽々ウェブクライアントでは、bootstrapのInput group(インプットグループ)クラスを使用してズームボタンを実現します。これにより、テキスト入力とボタンが一体として配置されるようになります。

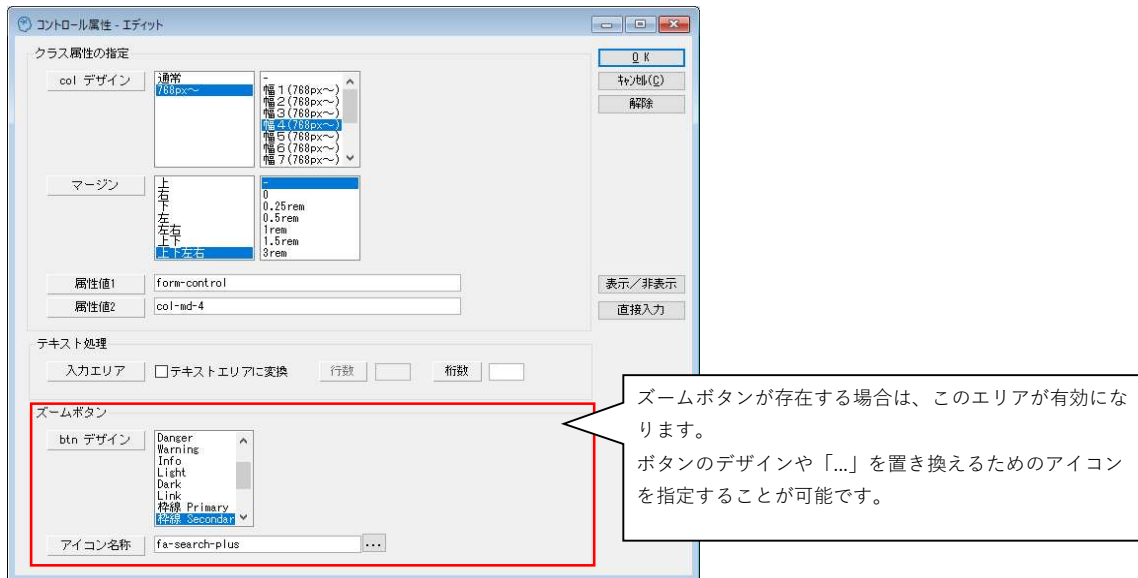
③ボタンのデザイン変更など

フォームやコントロールに対してデザインを施した例を見えます。



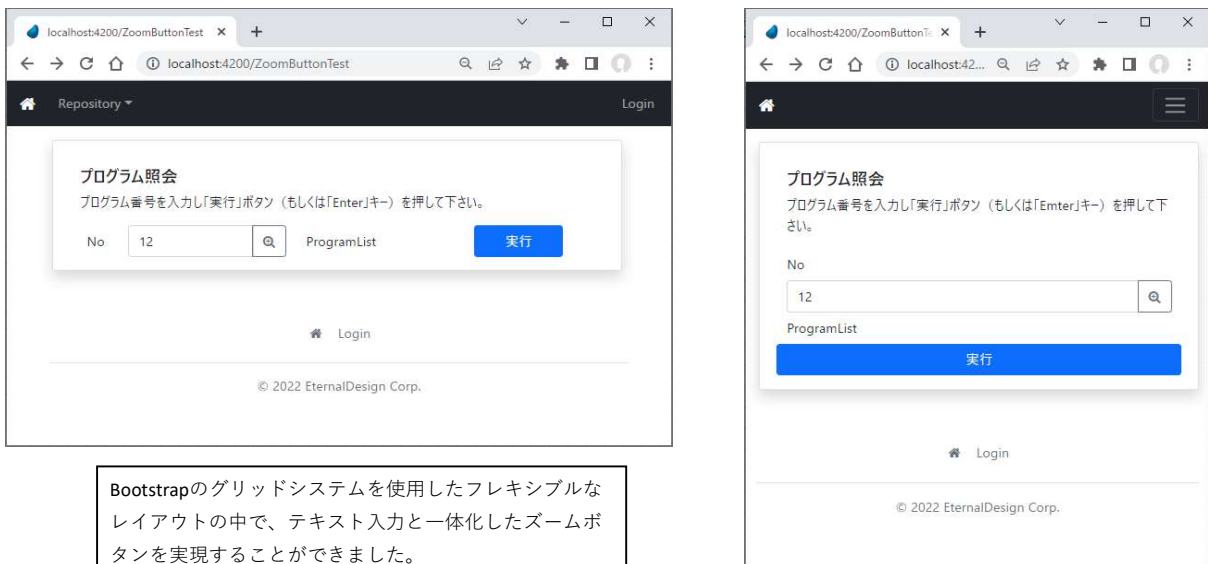
イメージ表示もズームボタンを表示するようになりました。
また、全体の縮尺比率や書式表示等も改善されています。

コントロール属性設定画面も拡張されました。
ボタンのデザイン等も簡単に指定できるようになりました。



ズームボタンが存在する場合は、このエリアが有効になります。
ボタンのデザインや「...」を置き換えるためのアイコンを指定することが可能です。

変換例を示します。



Bootstrapのグリッドシステムを使用したフレキシブルなレイアウトの中で、テキスト入力と一体化したズームボタンを実現することができました。

HTML変換結果

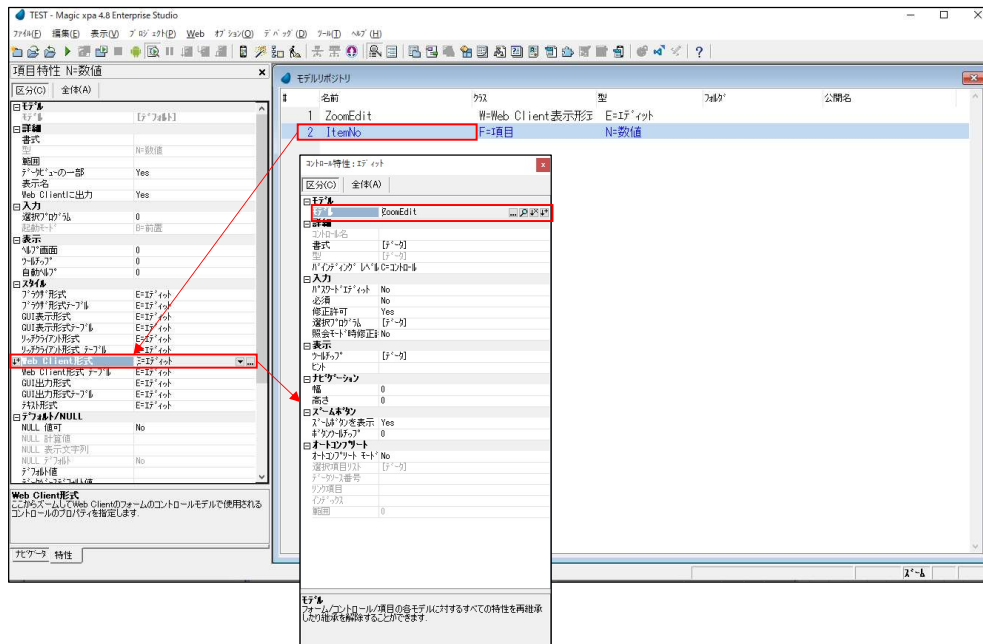
```
ZoomButtonTest.components.sm.html - SMWC - Visual Studio Code
ZoomButtonTest.components.sm.html x
Go to component
1 <div novalidate [formGroup]="screenFormGroup">
2   <div [magic]="mgc.ZoomButtonTest">
3     <div class="row">
4       <div class="card bg-white shadow mx-3 mb-3 px-3 pt-3" style="width: 95%;">
5         <div class="card-body">
6           <h5 class="card-title">プログラム照会</h5>
7           <p class="card-text">プログラム番号を入力し「実行」ボタン（もしくは「Enter」キー）を押して下さい。</p>
8           <div class="row">
9             <div [magic]="mgc.Group4">
10              <div class="row">
11                <label [magic]="mgc.Label2" class="lable_overflow mgFont1 col-md-1 col-form-label text-md-end me-1">
12                  No
13                </label>
14                <div class="col-md-4">
15                  <div class="input-group">
16                    <input [formControlName]="mgc.V_PNO" type="number" [magic]="mgc.V_PNO" mgFormat class="form-control"/>
17                    <mgError [magic]=mgc.V_PNO>
18                    </mgError>
19                    <button type="button" class="btn btn-outline-secondary" [magic]="mgc.V_PNO">
20                      <i class="fa fa-search-plus"></i>
21                    </button>
22                  </div>
23                </div>
24                <label [magic]="mgc.PName" class="col-md-4 col-form-label">
25                  {{mg.getValue(mgc.PName)}}
26                </label>
27                <button [magic]="mgc.PB1" class="btn btn-primary col-md-2">
28                  {{mg.getValue(mgc.PB1)}}
29                </button>
30              </div>
31            </div>
32          </div>
33        </div>
34      </div>
35    </div>
36  </div>
37 </div>
38
```

2. スタイル情報に使用されたモデルのリファレンス

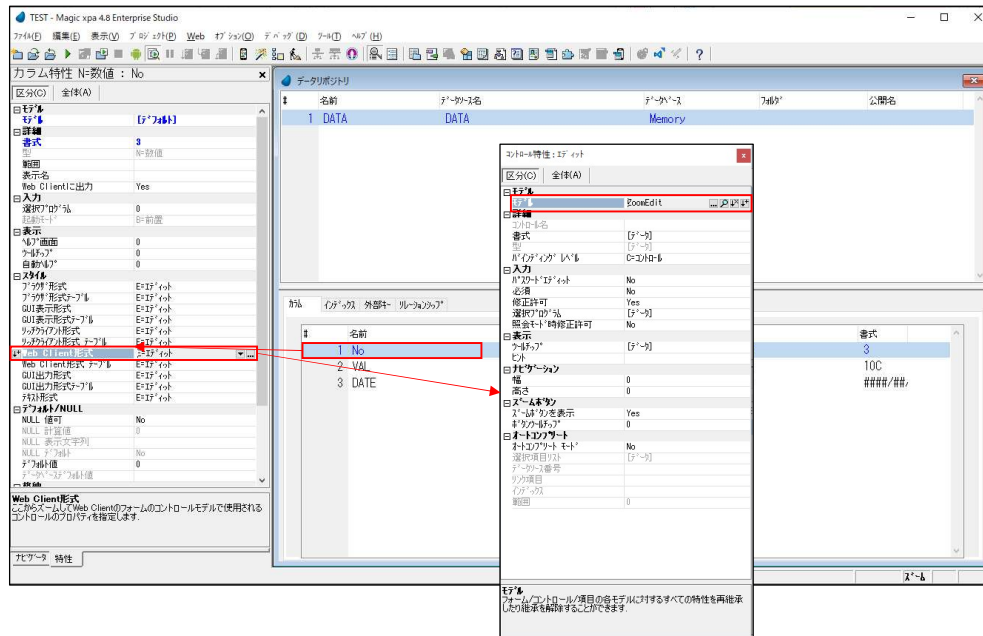
モデルのリファレンスを、スタイル情報(モデル、データ、プログラム項目)で使用しているモデルに対しても取得するようにしました。但し、現状は「GUI表示形式」「GUI表示形式テーブル」「リッチクライアント表示形式」「リッチクライアント表示形式テーブル」「WebClient形式」「WebClient形式テーブル」のみに限定しています。これらのスタイル情報にモデルの利用を取得したときは、「モデル」一覧画面のクロスリファレンスボタン「...」から確認できるようになりました。

<検証データ>

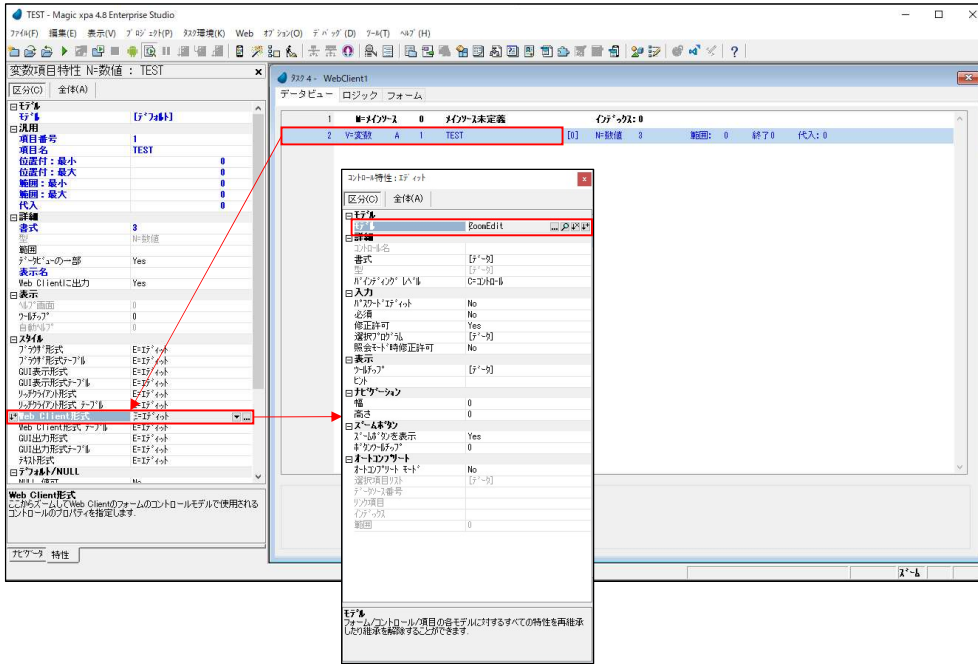
イ) 項目タイプのモデルのスタイル情報で表示形式のモデルを指定しているケース



ロ) データの列のスタイル情報で表示形式のモデルを指定しているケース



ハ)プログラムの変数のスタイル情報で表示形式のモデルを指定しているケース



<リファレンスの確認方法>

モデル一覧プログラムを起動します。
 「CR」カラムのリファレンスボタン(関連がある場合に表示)を押します。
 選択ポップアップが出た場合は「S:スタイル」を選択します。

