



GSLetterNeo Vol.111

2017年10月

年表化プログラム orca のデータのカスタム表示

松原 伸人

matubara@sra.co.jp

◆ はじめに

Vol.100 で 時間情報を持つテキストの年表化プログラム orca について紹介し、Vol.102 と Vol.103 では年表記述スクリプト MC を紹介し、Vol.104 と Vol.105 では MC で書いたデータをどのように orca で年表表示しているのかを説明しました。

多くのデータは Vol.102 で紹介したような「テキスト」欄が 1 個だけあるようなデータではありません。(例 1)

年月,テキスト

2008-08,Vol.1 アジャイルな計画 (1) / 野島 勇 / 計画 アジャイルソフトウェア 適応 手法 見直し 従来 不透明 見積り 距離 変化
2008-09,Vol.2 アジャイルな計画 (2) / 野島 勇 / ストーリ ストーリポイント 距離 値 購入 理想 商品 手間 時間 クレジットカード
2008-10,Vol.3 アジャイルな計画 (3) / 野島 勇 / 速度 距離 ストーリポイント ストーリ プランニング ポーカー カード 値 反復 議論
2008-11,Vol.4 ユースケースモデリング / 土屋 正人 / ユースケース アクタ レベル Jacobson システム 記述 条件 Cockburn 利害 粒
2008-12,Vol.5 ふりかえり / 野島 勇 / 質問 自分 面 今年 気持ち 出来事 大切 状態 紙 ふりかえり

例 1 タイトルと著者とキーワードが 1 項目に書かれている

[GSLetterNeo Web 年表](#) には「テキスト」欄に、記事のタイトルと著者とキーワードを書いています。よくあるデータフォーマットは、タイトルと著者とキーワードを 3 項目に分けて書いてあり、また管理したり分析したりするような場合、項目が分かれている方が扱いやすいです。(例 2)

orca で既にあるデータを表示するためにフォーマットを変更することもできますが、今回はデータに手を加えずに、フォーマットに合わせて orca プログラムの一部を書き換えたり、自前のコンテンツを追加したりする 2 つの方法を紹介します。

◆方法 1 コンテンツの実装を上書きして書き換える

orca は JavaScript で実装してあります。

JavaScript では、実装してある関数を上書きして書き換えることができます。

コンテンツのビジュアルを生成する関数は Orca.js の createContent です。

年月,タイトル,著者,キーワード

2008-08,Vol.1 アジャイルな計画 (1),野島 勇,計画 アジャイルソフトウェア 適応 手法 見直し 従来 不透明 見積り 距離 変化
2008-09,Vol.2 アジャイルな計画 (2),野島 勇,ストーリ ストーリポイント 距離 値 購入 理想 商品 手間 時間 クレジットカード
2008-10,Vol.3 アジャイルな計画 (3),野島 勇,速度 距離 ストーリポイント ストーリ プランニング ポーカー カード 値 反復 議論
2008-11,Vol.4 ユースケースモデリング,土屋 正人,ユースケース アクタ レベル Jacobson システム 記述 条件 Cockburn 利害 粒
2008-12,Vol.5 ふりかえり,野島 勇,質問 自分 面 今年 気持ち 出来事 大切 状態 紙 ふりかえり

例 2 タイトルと著者とキーワードが 3 項目に書かれている

createContent は、MC ファイルを読み込んで CSV のヘッダー部分に書いてあるデータの種類に応じたビジュアル要素を作成します。

この関数を、タイトルと著者とキーワードの 3 つの項目から 1 個のビジュアル要素を作成するように上書きするのが次のコードです。

```

<script type="text/javascript">
function createMyContent (event) {
  var content = document.createElement('div'),
      titleElement = document.createElement('div'),
      authorElement = document.createElement('div'),
      keywordElement = document.createElement('div');
  content.className = 'content';
  titleElement.className = 'title';
  authorElement.className = 'author';
  keywordElement.className = 'keyword';
  titleElement.innerHTML = event.data.タイトル;
  authorElement.innerHTML = event.data.著者;
  keywordElement.innerHTML = event.data.キーワード;
  content.appendChild(titleElement);
  content.appendChild(authorElement);
  content.appendChild(keywordElement);
  return content;
}

Orca.Event.prototype.createContent = function () {
  return createMyContent(this);
};
</script>

```

上書きで書き換えるのが分かりやすいですが、もとの振る舞いを期待していたプログラムが動かなくなる可能性があります。

色々なコンテンツを扱いたいときには向いていません。

orca にはあらかじめテキストと画像と動画の 3 種類のコンテンツを表示するプログラムが実装してあります。

元々のテキストや画像や動画を扱う実装をそのまま使いたいようなときはコンテンツの種類を追加して書き換えます。

◆方法 2 コンテンツの種類を追加して書き換える

テキストなどのデータの種類の定義とビジュアル要素の対応関係は、contentTypeDictionary に書いてあります。

contentTypeDictionary にタイトルと著者とキーワードそれぞれに対応するビジュアル要素を作成する関数を追加します。

今回の例では、タイトルでも著者でもキーワードでも同じ上記の createMyContent 関数でビジュアル要素を作成するようにしました。

```

<script type="text/javascript">
Orca.Event.prototype.contentTypeDictionary.タイトル = function (event) {
  return createMyContent(event);
};
Orca.Event.prototype.contentTypeDictionary.著者 = function (event) {
  return createMyContent(event);
};
Orca.Event.prototype.contentTypeDictionary.キーワード = function (event) {
  return createMyContent(event);
};
</script>

```

タイトルと著者とキーワードのビジュアルの定義はスタイルシートに書いています。

```

<style type="text/css">
.content {
  font-size: 12px;
  line-height: 12px;
  padding: 0 0 4px 0;
}
.content > .title {
  font-size: 14px;
  font-weight: bold;
}
.content > .author {
  font-weight: bold;
}
.content > .keyword {
  font-style: italic;
}
</style>

```

夢を。



GSLetterNeo Vol. 111

2017 年 10 月 20 日発行

発行者 ● 株式会社 SRA 先端技術研究所

編集者 ● 土屋正人

株式会社SRA

〒171-8513 東京都豊島区南池袋 2-3-2-8

夢を。Yawaraka Innovation
やわらかいのべしょん