

TAUEDA2Engine

クライアントサイド・アートアーカイブ検索ライブラリ

for デザイナー

TAUEDA2Engine とは

- 多摩美術大学のアートアーカイブ構想の一角をなす、最も利用者側で動作するウェブブラウザ内で動作するソフトウェア
- アーカイブを検索する利用者が直接利用することはなく、動作していることを感じさせない
- 検索を実行し、検索結果をウェブページに反映させる複雑なプロセスをほとんど自動的に行う
- ユーザインターフェースは、HTML を用いて、一般的なウェブページを作成する感覚で記述可能

TAUEDA2Engine とは

～何ができるか

- アートアーカイブを検索して、検索結果をウェブブラウザで閲覧
- 既存のウェブページに、検索欄を組み込み、検索結果を表示
- アートアーカイブからデータを取得し、d3.js などのビジュアライズエンジンを使用してグラフィカルにインタラクティブに表示
- google 検索や他の検索エンジンと連携したビジュアライズ
- HTML を用いて出力画面のテンプレートだけを記述しておき、具体的な検索結果のデータを自動的に埋め込んで表示

TAUEDA2Engine とは

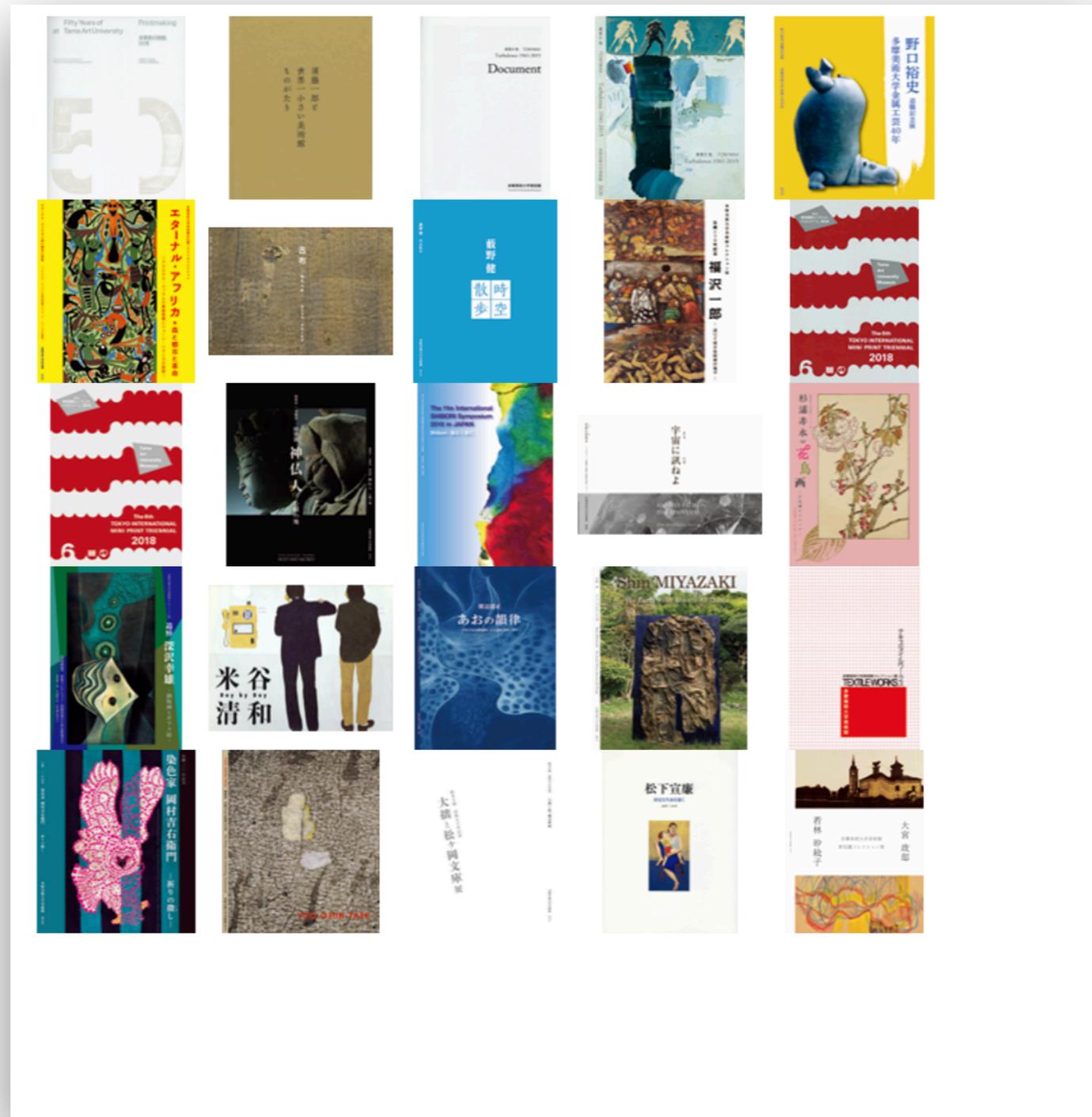
～事例1：ミニマルでダイナミックなウェブページ

ランダム検索

- 屏風絵の世界 — 李朝生活画 — 展 [展覧会名]
Not Edited
[1991/04/09](#) ~ [1991/05/24](#) [開催]
- 東京国際ミニプリントトリエンナーレ2015 [展覧会名]
Not Edited
[2015/09/26](#) ~ [2015/11/08](#) [開催]
- 起源：堀浩哉 [書名]
[堀浩哉退職記念展実行委員会](#), [堀浩哉](#) [著者]
[2014](#) ~ [出版]
- 宮崎進 多摩美術大学退職記念展 [展覧会名]
Not Edited
[1992/06/22](#) ~ [1992/07/10](#) [開催]
- 幽暗のレオナルド・ダヴィンチ [日本語]
Not Edited
[1986](#) ~ [制作]
- 多摩美術大学美術館新収蔵コレクション展 テキスタイルワークス [展覧会名]
Not Edited
[2017/04/05](#) ~ [2017/07/02](#) [開催]
- 現代の東南アジア美術：それぞれの視点 [書名]
[チェ・キョンファ](#), [パトリック・D.フローレス](#), [エイリオン・ク](#), [ジョアンナ・リー](#), [デヴィ・マリアント](#), [ニランジャン・ラジャ](#), [ハップ・トゥーイ](#), [ロー・ジー](#)
[ー・ウィー](#) [執筆]
[2003](#) ~ [出版]
- エターナル・アフリカ*森と都市と革命 —アミルカル・カブラルの革命思想とジョージ・リランガの芸術— [展覧会名]
Not Edited
[2019/07/27](#) ~ [2019/10/14](#) [開催]
- 大宮政郎若林砂絵子：多摩美術大学美術館新収蔵コレクション展 [書名]
[小林宏道](#), [吉田公子](#) [執筆]
[2016/04/06](#) ~ [出版]
- 多摩美術大学博士課程展 2017 [展覧会名]
Not Edited
[2017/03/08](#) ~ [2017/03/23](#) [開催]

TAUEDA2Engine とは

～事例2：画像のみを取得してタイル表示



TAUEDA2Engine とは

～事例3：美術館のカatalog検索

多摩美術大学美術館
Tama Art University Museum
ご利用案内 → 通信販売

年代
すべて

在庫状況
すべて

フリーワード
任意単語

リセット

2020年7月4日 - 9月22日 **会期**
町田恵美, 関川歩, 土屋誠一, 石田尚志, 鶴岡真弓, 照屋勇賢, 金城満, 銘苅靖, 熊谷博人, 住友文彦, 建島哲 **執筆**
多摩美術大学美術館
未入力
未入力
非売品

アクティビティ ご利用案内 ▾ 展覧会 ▾ コレクション ▾ 施設概要 ▾



真喜志勉 Tom Max : turbulence 1941-2015
2020年7月4日 - 9月22日 **会期**
町田恵美, 豊見山愛, 比嘉良治, 石田尚志, 港千尋, 関川歩 **執筆**
多摩美術大学美術館
2020 **出版**
104ページ **頁**
在庫あり
一般 2000円 (学生1500円)



野口裕史退職記念展：多摩美術大学金属工芸40年
2019年10月26日 - 11月24日 **会期**
建島哲, 野口裕史 **執筆**
多摩美術大学工芸学科金属プログラム研究室
2019 **出版**
58ページ **頁**
在庫あり
1000円



エターナル・アフリカ：森と都市と革命：アミルカル・カブラルの革命思想とジョージ・リランガの芸術：多摩美術大学美術館白石頭二アフリカコレクション
2019年7月27日 - 10月14日 **会期**
石塚正英, 小林宏道, 小川忠博, 山本富美子, 白石頭二 **執筆**
多摩美術大学美術館
2019 **出版**
119ページ **頁**
在庫あり
1500円

TAUEDA2Engine とは

～事例4：美術館のカタログ詳細

多摩美術大学 美術館 → ご利用案内 → 通信販売



管理 ID TAUMC

管理名称 真喜志勉 TOM MAX Turbulence 1941-2015

作成日時 2021/05/21 10:13:25.996

更新日時 2021/07/26 19:55:36.469

改訂番号 15



資料区分 展覧会図録

タイトル 真喜志勉 Tom Max : turbulence 1941-2015 書名
真喜志勉 : turbulence 1941-2015 別書名

年月日 2020 出版

場所 未入力

クレジット 町田恵美 執筆

豊見山愛 執筆

比嘉良治 執筆

石田尚志 執筆

港千尋 執筆

関川歩 執筆

関川歩 編集

町田恵美 編集

百瀬梓 デザイン

ミヤギフトシ 翻訳

ベン・デイビス 翻訳

岡本尚文 写真撮影 (表紙、P6、P8-19、P23-51、P81-83、P89-90、P105)

川村はるな 編集補助

淵田雄 編集協力

渡辺真弓 編集協力

株式会社オフィスエンズ 印刷

坂巻広志 製版

パブリッシャー 多摩美術大学美術館

作家発表 未入力

作家署名 未入力

サイズ 縦高 250mm 横幅 188mm 奥行 15mm 指定なし

継続時間 未入力



TAUEDA2Engine とは

～利用手順

- HTML 内で jQuery をあらかじめ読み込む (TAUEDA2Engine が jQuery に依存しているため)
- 次に、taueda2.js もしくは、taueda2.min.js を読み込む
- HTML 内に TAUEDA2Engine の指定する書式に従って、ユーザーインターフェースのテンプレートとなるタグを記述する
- <script> タグ内、もしくは、外部の javascript ファイルに TAUEDA2Engine が必要とする最小限のコードを記述する
- ウェブブラウザで HTML を開いて閲覧する

TAUEDA2Engine

クライアントサイド・アートアーカイブ検索ライブラリ

for プログラマー

TAUEDA2Engine とは

- 多摩美術大学のアートアーカイブ構想の一角をなす、最もクライアントサイドで動作するソフトウェア
- アートアーカイブをWebブラウザを通して閲覧するときに、ウェブブラウザ上で動作する JavaScript で記述されたライブラリ
- アートアーカイブの検索を担うクラスと、検索結果を描画するクラスから構成
- 検索実行のロジック部分と、検索画面のユーザインターフェースの構築を完全に分離

含まれるクラス

- TAUEDA2Engine …検索を実行して、結果を保持するクラス
- TAUEDA2Renderer …HTML で記述されたユーザインターフェースを解析して、検索結果を描画するクラス
- TAUEDA2Utility …ソフトウェア開発で頻繁に使う便利機能をまとめたクラス
- TAURequestFailedException …通信に関する例外
- TAUInvalidResultException …通信結果のデータに関する例外

TAUEDA2Engine クラス

～基本的な操作の流れ

```
// インスタンスを作成
```

```
const eda2 = new TAUEDA2Engine();
```

```
// キーワード検索を実行
```

```
eda2.search((begin, length, error) => {
```

```
    // 結果を取得
```

```
    console.log(eda2.getTitle(begin, "", true, true));
```

```
    return;
```

```
}, '日本画');
```

TAUEDA2Engine クラス

～キーワード検索とランダム検索

- キーワード検索とは、アートアーカイブ検索エンジンが解釈できる検索クエリを送信して、検索を行う方法

```
search(callback, query, begin = 0, refresh = false);
```

- ランダム検索とは、特定の検索クエリを送信せずに検索を実行し、アートアーカイブ検索エンジンからランダムに結果を取得する方法

```
random(callback);
```

TAUEDA2Engine クラス

～検索結果の取得

- キーワード検索は、指定した検索クエリに対して、結果の開始番号と個数を指定して行うことができる
- 検索結果はクラス内に自動的にキャッシュされ、前回と同一の検索クエリを使用している期間内では、一度取得した結果は再度通信を行わずに迅速に取得することができる
- キャッシュを無効にし、強制的に通信を行い、最新の結果を取得することもできる
- ランダム検索では、個数を指定して結果を取得することができ、検索結果はキャッシュされない

TAUEDA2Engine クラス

～含まれる関数

- constructor(url = "", length = 0, type = {}, tag = {})
- get url()
- get length()
- get type()
- get tag()
- get query()
- get total()
- get found()
- get isBusy()
- get isRandom()
- getDBID(index)
- getThumbnail(index)
- getImage(index, key = "", extract = false, alternate = false)
- getIdentifier(index)
- getName(index)
- getClassification(index)
- getLifecycle(index)
- getStructureLinkO(index)
- getStructureLinkI(index)
- getRelationLinkO(index)
- getRelationLinkI(index)
- getArtist(index, key = "", extract = false)
- getTitle(index, key = "", extract = false, alternate = false)
- getDate(index, key = "", extract = false)
- getCredit(index, key = "", extract = false)
- getPublisher(index, extract = false)
- getPublish(index)
- getSign(index, key = "", extract = false)
- getTag(index, key = "", extract = false)
- getPlace(index, key = "", extract = false)
- getSize(index, key = "", extract = false)
- getDuration(index, key = "", extract = false)
- getPage(index, key = "", extract = false)
- getNumber(index, key = "", extract = false)
- getPublic(index, key = "", extract = false)
- getCreation(index)
- getUpdate(index)
- getRevision(index)
- getCard(index)
- random(callback)
- search(callback, query, begin = 0, refresh = false)

TAUEDA2Renderer クラス

～利用方法と概要

- ユーザーインターフェースを、HTML で記述する
- HTML の中に、最小限の JavaScript コードを埋め込む
- 初回の検索方法を指定する
- HTML での指定によるが、検索キーワードのリンク処理や複数項目の展開などは自動的に行われる
- 閲覧者のリンクのクリックなどの操作は自動的に処理され、適切な検索が行われ、再度自動的に結果が描画される

TAUEDA2Renderer クラス

～含まれる機能

- 検索の実行と結果の描画
- HTML 内の変数への結果の反映と、複数項目の自動的な展開
- リンクされた項目の自動的なリンク処理
- 検索対象を大きな枠組みで制限するためのサンドボックス機能
- 検索対象を複数パターンで絞り込むためのフィルター機能
- 検索結果の内容で表示結果を動的に切り替えるための、簡単な条件記述機能

TAUEDA2Renderer クラス

～ユーザインターフェースの定義

```
<body id="artwork" class="tau-eda2">
  <div class="tau-eda2-search">
    <div class="tau-eda2-busy">
      <p>読み込み中...</p>
    </div>
    <ul class="tau-eda2-result">
      <li class="tau-eda2-card">
        <span class="tau-eda2-title">
          <span class="tau-eda2-on tau-eda2-unfold" data-eda2-delimiter="&lt;br /&gt;">#{value}% <i>[#{tag}
%]</i></span>
          <span class="tau-eda2-off tau-eda2-unfold" style="color:#aaaaaa;">#{value}%</span>
        </span>
      </li>
    </ul>
    <div class="tau-eda2-error">
      <p class="tau-eda2-message">
        <span class="tau-eda2-on tau-eda2-off">通信エラー: #{value}%</span>
      </p>
    </div>
  </div>
</body>
```

TAUEDA2Renderer クラス

～基本的な操作の流れ

```
// 全ての DOM が読み込まれてから処理を開始
```

```
jQuery(() => {
```

```
    // インスタンスを作成
```

```
    const eda2 = new TAUEDA2Renderer('#artwork');
```

```
    // 初期状態を読み込み
```

```
    eda2.search();
```

```
    return;
```

```
});
```

TAUEDA2Renderer クラス

～インターフェース記述用の HTML キーワード

クラス指定子

tau-eda2
tau-eda2-search
tau-eda2-inspect
tau-eda2-busy
tau-eda2-result
tau-eda2-total
tau-eda2-found
tau-eda2-query
tau-eda2-random
tau-eda2-card
tau-eda2-dbid
tau-eda2-thumbnail
tau-eda2-image
tau-eda2-identifier
tau-eda2-name
tau-eda2-classification
tau-eda2-lifecycle
tau-eda2-structurelink-o
tau-eda2-structurelink-i
tau-eda2-relationlink-o
tau-eda2-relationlink-i
tau-eda2-artist
tau-eda2-title
tau-eda2-date

tau-eda2-credit
tau-eda2-publisher
tau-eda2-publish
tau-eda2-sign
tau-eda2-tag
tau-eda2-place
tau-eda2-size
tau-eda2-duration
tau-eda2-page
tau-eda2-number
tau-eda2-public
tau-eda2-creation
tau-eda2-update
tau-eda2-revision
tau-eda2-qrcode
tau-eda2-graph
tau-eda2-front
tau-eda2-prev
tau-eda2-index
tau-eda2-next
tau-eda2-back
tau-eda2-retry
tau-eda2-return
tau-eda2-error
tau-eda2-message
tau-eda2-on

tau-eda2-off
tau-eda2-alt
tau-eda2-unfold
tau-eda2-value
tau-eda2-key
tau-eda2-reftxt
tau-eda2-refimg
tau-eda2-graph-structure
tau-eda2-graph-relation
tau-eda2-graph-self
tau-eda2-graph-parent
tau-eda2-graph-child
tau-eda2-graph-sibling

※

第1階層
第2階層
第3階層
第4階層
第5階層
第6階層
第7階層
第8階層
特殊用途

パラメータ指定子

data-eda2-strict
data-eda2-key
data-eda2-extract
data-eda2-alternate
data-eda2-on
data-eda2-off
data-eda2-alt
data-eda2-scale
data-eda2-delimiter
data-eda2-query
data-eda2-value
data-eda2-key
data-eda2-reftxt
data-eda2-refimg
data-eda2-width
data-eda2-height
data-eda2-box-width
data-eda2-box-height
data-eda2-value
data-eda2-key
data-eda2-reftxt
data-eda2-refimg
data-eda2-src

TAUEDA2Renderer クラス

～第1階層

- HTML の任意のタグのクラスとして 'tau-eda2' を指定することによって TAUEDA2Renderer クラスに描画対象であることを指示する役目
- その他の全てのキーワードは、'tau-eda2' を指定したタグの子孫タグに設定する
- 例：

```
<div class="tau-eda2">  
</div>
```

TAUEDA2Renderer クラス

～第2階層

- 検索モードと詳細モードによりコンテンツを切り替える役目
- 任意のタグのクラスとして、検索モード 'tau-eda2-search', 詳細モード 'tau-eda2-inspect' を指定する
- それぞれのキーワードは必要に応じて省略できる
- 例：

```
<div class="tau-eda2-search">  
</div>  
<div class="tau-eda2-inspect">  
</div>
```

TAUEDA2Renderer クラス

～第3階層

- 通信の状態によりコンテンツを切り替える役目
- 任意のタグのクラスとして、通信中 'tau-eda2-busy', 通信成功 'tau-eda2-result', 通信失敗 'tau-eda2-error' を指定する
- それぞれのキーワードは必要に応じて省略できる
- 例：

```
<div class="tau-eda2-result">  
</div>  
<div class="tau-eda2-error">  
</div>
```

TAUEDA2Renderer クラス

～第4階層

- 第3階層の内容について詳細を記述する役目
- 第3階層の状態に応じて、利用できるキーワードが変わる
- それぞれのキーワードは必要に応じて省略できる

- 例：

```
<div class="tau-eda2-result">  
  <span class="tau-eda2-query">  
    <span class="tau-eda2-on">{%value}%</span>  
  </span>  
</div>
```

TAUEDA2Renderer クラス

～第5階層

- 第4階層が 'tau-eda2-card' の場合は、取得するデータベースの項目を指定する役目
- それ以外の場合は、第5階層はスキップ
- 'data-eda2-key', 'data-eda2-extract', 'data-eda2-alternate' を使用して、項目の取得の方法を変更可能
- 'data-eda2-strict' を使用して、描画結果のタグの内容が空の場合でも、タグを自動消去しないように指定可能
- それぞれのキーワードは必要に応じて省略できる

TAUEDA2Renderer クラス

～‘tau-eda2-busy’

- このキーワードが指定されているタグは、通信中に有効
- 利用できる第4階層以下のキーワードは無し
- 利用できる変数は無し

TAUEDA2Renderer クラス

～‘tau-eda2-result’

- このキーワードが指定されているタグは、通信成功時に有効
- 利用できる第4階層のキーワードは、‘tau-eda2-total’, ‘tau-eda2-found’, ‘tau-eda2-query’, ‘tau-eda2-random’, ‘tau-eda2-card’, ‘tau-eda2-front’, ‘tau-eda2-prev’, ‘tau-eda2-index’, ‘tau-eda2-next’, ‘tau-eda2-back’, ‘tau-eda2-retry’, ‘tau-eda2-return’
- 利用できる変数には、%{value}% があり、キーワードの種類によって内容が異なる

TAUEDA2Renderer クラス

～‘tau-eda2-error’

- このキーワードが指定されているタグは、通信失敗時に有効
- 利用できる第4階層のキーワードは、‘tau-eda2-message’
- 利用できる変数は、%{value}%

TAUEDA2Renderer クラス

～第3階層から第6階層まとめ

第3階層	第4階層	第6階層	変数	条件と変数の内容
tau-eda2-busy	なし	なし	なし	
tau-eda2-result	tau-eda2-total	tau-eda2-on, tau-eda2-off	{value}	登録数が0より大きい時に on 状態、0以下の時に off 状態 {value} は登録数
	tau-eda2-found	tau-eda2-on, tau-eda2-off	{value}	一致数が0より大きい時に on 状態、0以下の時に off 状態 {value} は一致数
	tau-eda2-query	tau-eda2-on, tau-eda2-off	{value}	検索クエリが有効な文字列の時に on 状態、空文字列の時に off 状態 {value} は検索クエリ
	tau-eda2-random	tau-eda2-on, tau-eda2-off	なし	ランダム検索時に on 状態、キーワード検索時に off 状態
	tau-eda2-card	後述を参照	第4階層に依存	第4階層に依存
	tau-eda2-front	tau-eda2-on, tau-eda2-off	なし	ランダム検索時は削除、キーワード検索時に先頭に遷移可能な時に on 状態、 遷移不可能な時に off 状態
	tau-eda2-prev	tau-eda2-on, tau-eda2-off	なし	ランダム検索時は削除、キーワード検索時に前ページに遷移可能な時に on 状態、 遷移不可能な時に off 状態
	tau-eda2-index	tau-eda2-on, tau-eda2-off	{value}	ランダム検索時は削除、キーワード検索時に現在のページの時に on 状態、 現在のページではない時に off 状態、{value} はページ番号
	tau-eda2-next	tau-eda2-on, tau-eda2-off	なし	ランダム検索時は削除、キーワード検索時に次ページに遷移可能な時に on 状態、 遷移不可能な時に off 状態
	tau-eda2-back	tau-eda2-on, tau-eda2-off	なし	ランダム検索時は削除、キーワード検索時に末尾に遷移可能な時に on 状態、 遷移不可能な時に off 状態
	tau-eda2-retry	tau-eda2-on, tau-eda2-off	なし	キーワード検索時は削除、ランダム検索時は常に on 状態
	tau-eda2-return	tau-eda2-on, tau-eda2-off	なし	ブラウザの履歴が存在する時に on 状態、存在しない時に off 状態
tau-eda2-error	tau-eda2-message	tau-eda2-on, tau-eda2-off	{value}	エラーメッセージが有効な時に on 状態、無効な時に off 状態 {value} はエラーメッセージ

TAUEDA2Renderer クラス

～第4階層が‘tau-eda2-card’の場合

- 第4階層が‘tau-eda2-card’の場合には、第5階層で取得したいデータベースの項目を指定する
- データベースの項目毎に、取得されるデータの形式に違いがあるが、大きく分けて、画像情報・単純文字列・任意形式の3種類である
- 第6階層から第8階層で、取得したデータをどのように表示するかを指定する

TAUEDA2Renderer クラス

～作品タイトルの取得例

```
<div class="tau-eda2">  
  <div class="tau-eda2-search tau-eda2-result">  
    <div class="tau-eda2-card">  
      <span class="tau-eda2-title">  
        <span class="tau-eda2-on tau-eda2-alt tau-eda2-  
unfold">%{value}% [%{key}%]</span>  
        <span class="tau-eda2-off tau-eda2-unfold">%  
{value}%</span>  
      </span>  
    </div>  
  </div>  
</div>
```

TAUEDA2Renderer クラス

～第6階層

- 第4階層または、第5階層の内容について、条件に応じて描画結果を切り替える役目
- 有効時 'tau-eda2-on', 無効時 'tau-eda2-off', 代替時 'tau-eda2-alt' を利用できる
- 'data-eda2-on', 'data-eda2-off', 'data-eda2-alt' を使用して詳細な条件を指定することも可能
- それぞれのキーワードは必要に応じて省略できる

TAUEDA2Renderer クラス

～第7階層

- 取得した項目の内容を展開する場所を指定する役目
- 取得した項目の内容は、‘tau-eda2-unfold’を指定したタグの子孫タグの中で記述される `%{name}%` 形式の変数に展開される
- ‘data-eda2-scale’, ‘data-eda2-delimiter’, ‘data-eda2-query’を使用して変数への展開時の挙動を変更可能
- ‘data-eda2-value’, ‘data-eda2-key’, ‘data-eda2-reftxt’, ‘data-eda2-refimg’を使用して変数への展開時の条件を記述可能
- それぞれのキーワードは必要に応じて省略できる

TAUEDA2Renderer クラス

～第8階層

- 変数の展開箇所の指定と展開時の条件をタグ形式で記述する役目
- 値変数用 ‘tau-eda2-value’, キー変数用 ‘tau-eda2-key’, 引用テキスト用 ‘tau-eda2-reftxt’, 引用画像用 ‘tau-eda2-refimg’ を使用できる
- タグが展開される条件の詳細を記述するために、‘data-eda2-value’, ‘data-eda2-key’, ‘data-eda2-reftxt’, ‘data-eda2-refimg’ を使用することができる（これらは、第7階層で指定できる同名の指定子とは別のものである）
- それぞれのキーワードは必要に応じて省略できる

TAUEDA2Renderer クラス

～第7階層と第8階層の関係

- 第7階層と第8階層は似たような機能を提供するが、タグを使用するかタグを使用したくないかによって使い分けることができる
- タイトルを取得して表示する時の2種類の記述方法を示す

```
<div class="tau-eda2-title">  
  <!-- タグを使用しない -->  
  <span class="tau-eda2-on tau-eda2-unfold">#{value}%</span>  
span>  
  <!-- タグを使用 -->  
  <span class="tau-eda2-on tau-eda2-unfold"><span class="tau-eda2-value">#{self}%</span></span>  
</div>
```

TAUEDA2Renderer クラス

～階層の統合

- 第1階層～第8階層を表現するために、階層毎に div タグなどで内容を囲うことは冗長である
- 冗長なタグが css での指定を複雑にしたり、通信量を増加させる
- 第2階層と第3階層、第6階層～第8階層は統合可能である

TAUEDA2Renderer クラス

～階層の統合

- 例 1 :

```
<div class="tau-eda2-search tau-eda2-result">  
  <span class="tau-eda2-found">  
    <span class="tau-eda2-on tau-eda2-off">{%value}%</span>  
  </span>  
</div>
```

- 例 2 :

```
<div class="tau-eda2-title">  
  <span class="tau-eda2-on tau-eda2-alt tau-eda2-unfold tau-  
eda2-value">{%self}%</span>  
  <span class="tau-eda2-off tau-eda2-unfold">{%value}%</span>  
</div>
```

TAUEDA2Renderer クラス

～含まれる関数

- constructor(id, preparator = null, completor = null, searcher = "", inspector = "", url = "", length = 0, type = {}, tag = {})
- get url()
- get length()
- get type()
- get tag()
- get mode()
- get total()
- get found()
- get isBusy()
- get isRandom()
- get engine()
- setSandbox(mode, sentence)
- getSandbox()
- clearSandbox()
- setFilter(group, name, mode, sentence)
- getFilter(group, name)
- clearFilter(group, name)
- useFilter(group, name, state)
- checkFilter(group, name)
- selectGroupFilter(group, name)
- resetGroupFilter(group)
- findGroupFilter(group, wildcard = undefined)
- getQuery(filter = false)
- search(query = null, begin = 0, force = false)
- inspect(dbid, force = false)

TAUEDA2Utility クラス

- ソフトウェア開発で頻繁に利用する基本的な機能を集約
- すべて static 関数
- クラスのインスタンス化不要

TAUEDA2Utility クラス

～含まれる関数

- static isUndefined(any)
- static isNull(any)
- static isValue(any)
- static isBoolean(any)
- static isNumber(any)
- static isString(any)
- static isArray(any)
- static isObject(any)
- static verifyBoolean(any, boolean)
- static verifyNumber(any, number, finite = true)
- static verifyString(any, string)
- static verifyArray(any, array)
- static verifyObject(any, object)
- static getFixed(number)
- static getPrecision(number)
- static formatDate(date, format)
- static formatURLParam(prefix, object)
- static parseURLParam(prefix, string)
- static postJSON(url, json)