

更新検知ツール

[システム仕様書]

2010/10/21

独立行政法人 産業技術総合研究所
バイオメディシナル情報研究センター

目次

本システム仕様書の構成は、以下のとおりです。

0. システム構成

P. 2

1. 更新検知ツール

P. 3

1.1 処理仕様

P. 3

1.2 処理詳細

P. 4

1.3 エラー処理

P. 6

1.4 XMLの構造

P. 7

2. 管理GUI

P. 8

2.1 ディレクトリ構成

P. 8

2.2 Basic認証

P. 9

2.3 画面仕様

P. 10

2.3.1 検索メイン画面

P. 10

2.3.2 アイテム編集画面

P. 12

2.3.3 アイテム削除画面

P. 14

2.3.4 アイテム追加画面

P. 16

2.3.5 アイテムグループ編集画面

P. 18

2.3.6 更新履歴編集画面

P. 21

3. DB設計

P. 23

3.1 .アイテムDB

P. 23

3.1.1 テーブル一覧表

P. 23

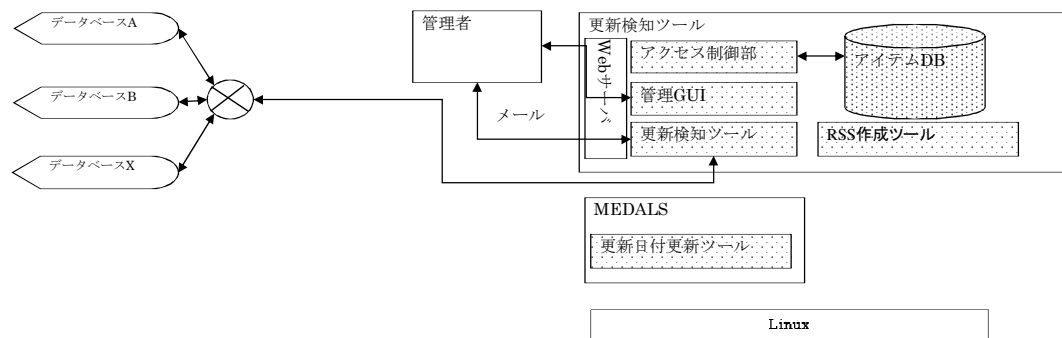
3.1.2 テーブル設計

P. 23

ドキュメント名	システム仕様書
システム名称	更新検知ツール

システム構成

システム構成は下図のとおり。



ドキュメント名	システム仕様書
システム名称	更新検知ツール

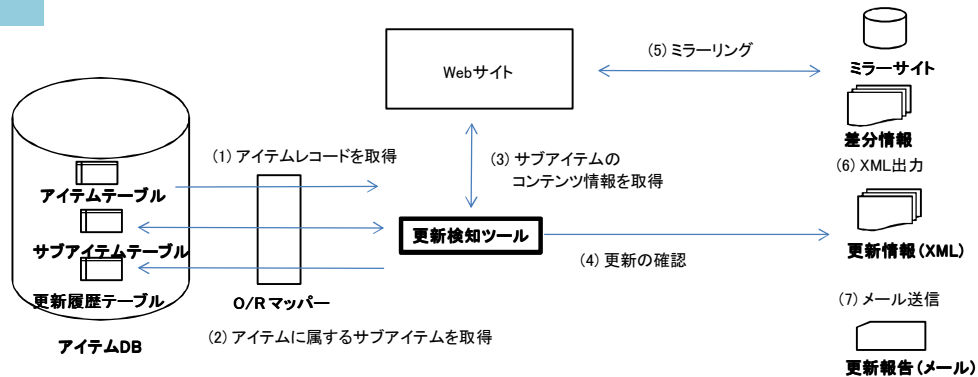
1. 更新検知ツール

- 指定されたWebページにアクセスし、更新を確認する。
- Webページは、それぞれ1つのアイテムとして、アイテムDBのアイテムテーブルで管理されている。
- 更新確認の範囲は、アイテムだけでなく、アイテムを構成する画像やCSS、同ドメイン内のリンク先にまで及ぶ。これらアイテムを構成するコンテンツは、それぞれサブアイテムとして、アイテムDBのサブアイテムテーブルで管理されている。1つのアイテムは、複数のサブアイテムで構成されることになる。
- 更新の確認は、MD5ハッシュ値の値について、新規にアクセスした時の値と、前回アクセスした時の値を比較することで確認する。
前回アクセスした時のMD5ハッシュ値の値は、サブアイテムテーブルで管理されている。
- 更新が検知された場合は、更新情報をXML形式で出力する。また、更新された場合かアイテムにエラーが発生した場合に、MEDALS管理者にメールで送信する。
- アイテムに初めてアクセスした時、または更新が検知された時、アイテムのサイトをミラーリングする。
ミラーサイトは管理GUIからブラウジングできる。
- アイテムが更新された場合は、更新履歴テーブルの更新を行う。
- アイテムがHTML形式の場合に、変更箇所がわかるようにタグ付けされたHTMLを出力する。
- アイテムが更新された場合に、diffiに実行し、差分情報を出力する。また、RSS出力用の差分も出力する。

1.1 処理仕様

プログラム名		detect_update.pl	
項番	項目	内容	備考
1	出力	更新情報	XML形式で出力する。
2	出力	差分情報	HTML形式(HTMLの場合のみ)、Text形式、RSS用のText形式の3種類を出力
3	出力	更新報告	MEDALS管理者宛てに、メールで送信する。

概念図



ドキュメント名	システム仕様書
システム名称	更新検知ツール

1.2 処理詳細

項番	処理ステップ	処理詳細	備考
1	主処理	<p>1 アイテムテーブルから、全レコードを取得する。</p> <p>2 アイテムレコードの件数だけ、2-1～3を行う。</p> <p>2-1 アイテムレコードの項目(*1)を引数とし、項番2を行う。</p> <p>2-2 項番2で、返却値が返された場合 項番2で返却された現在時刻とサブアイテムレベルの更新情報をもとに、 アイテムレベルの更新情報をXMLファイルに出力する。</p> <p>2-3 項番2で、何も返却されなかった場合 何もしない。</p> <p>3 2で1件以上の更新が確認された場合 更新された旨をメールで報告する。</p> <p>4 2で1件も更新が確認されなかった場合 何もしない。</p>	<p>*1 アイテムレコードの項目</p> <ul style="list-style-type: none"> ・item_id ・item_URL ・follow_link ・user_id ・password ・parameter
2	更新検知	<p>1 アイテムレコードの項目を引数として受け取る。</p> <p>2 サブアイテムテーブルから、item_idに対応するレコードを全て取得する。</p> <p>2-1 サブアイテムレコード(*2)が存在しない場合</p> <p>2-1-1 アイテムレコードの項目を引数とし、項番4を行う。</p> <p>2-1-2 何も返却しない。</p> <p>2-2 サブアイテムレコードが存在する場合</p> <p>2-2-1 サブアイテムレコードの件数だけ、2-2-1-1～3を行う。</p> <p>2-2-1-1 sub_item_URLを引数とし、項番3を行う この時、サブアイテムのMD5ハッシュ値を得る</p> <p>2-2-1-2 前回アクセスした時のMD5ハッシュ値（サブアイテムレコードのMD5）と、 2-2-1-1で得たMD5ハッシュ値を比較する。変化が認められた場合、 更新されたというフラグを立てる。</p> <p>2-2-1-3 更新を示すフラグが立っていた場合、サブアイテムレベルの更新情報を 保持しておく。フラグが立っていなかった場合、何もしない。</p> <p>2-2-2 2-2-1で、サブアイテムが1個以上更新された場合</p> <p>2-2-2-1 サブアイテムテーブルから、item_id（外部キー）が同じレコードを 全て削除する。</p> <p>2-2-2-2 アイテムレコードの項目を引数とし、項番4を行う。</p> <p>2-2-2-3 項番4の返却値を受け取る。これをミラーリング時刻とする。</p> <p>2-2-2-4 ミラーリング時刻と、サブアイテムレベルの更新情報を返却する。</p> <p>2-2-3 2-2-1で、サブアイテムが1個も更新されなかった場合 何も返却しない。</p>	<p>*2 サブアイテムレコード 項目は以下のとおり</p> <ul style="list-style-type: none"> ・sub_item_id ・sub_item_URL ・last_modified ・content_length ・e_tag ・MD5 ・item_id（外部キー）
3	コンテンツ情報取得	<p>1 URLを引数として受け取る。</p> <p>2 1のURLにアクセスし、コンテンツを取得する</p> <p>3 コンテンツのMD5ハッシュ値を計算する</p> <p>4 MD5ハッシュ値を返却する</p>	

ドキュメント名	システム仕様書
システム名称	更新検知ツール

1.2 処理詳細 (続き)

項番	処理ステップ	処理詳細	備考
4	ミラーリング	<ol style="list-style-type: none"> 1 アイテムレコードの項目を受け取る。 2 item_URLにアクセスする。 3 リダイレクトがあった場合 <ol style="list-style-type: none"> 3-1 item_URLをリダイレクト先のURLに設定し直す。 3-2 アイテムテーブルのitem_URLを更新する。 4 リダイレクトがなかった場合 何もしない。 5 項番5を行い、現在時刻を取得する。 これをミラーリング時刻(*3)とする。 6 item_URL のWebサイトをミラーリングする。 7 ミラーリングの際アクセスした各コンテンツについて、MD5ハッシュ値を計算し、 それぞれを1サブアイテムとしてサブアイテムテーブルに登録する。 それぞれのサブアイテムは、外部キーとして同じitem_idを持つように登録する。 8 更新履歴テーブルを更新する。 9 Wgetのログから抽出した、アイテムのURLとファイル名をlink.txtとして出力 10 HTMLの場合、tidyを実行し整形を行う。 整形したHTMLと前回の結果との差分をdiffを実行することにタグ付けし出力する。 再度tidyを実行し、タグの位置を修正する。結果をdiff.htmlとして出力する。 11 diffを実行し、差分情報を取得し、diff.txtとして出力する。 12 diffを実行し、RSS用にrss.frgとして出力する。 13 ミラーリング時刻を返却する 	<p>*3 ミラーリング時刻 ・年月日時分秒</p>
5	現在時刻取得	<ol style="list-style-type: none"> 1 現在時刻を取得する。 2 現在時刻(*4)を返却する。 	<p>(*4)現在時刻 ・年月日時分秒</p>

ドキュメント名	システム仕様書
システム名称	更新検知ツール

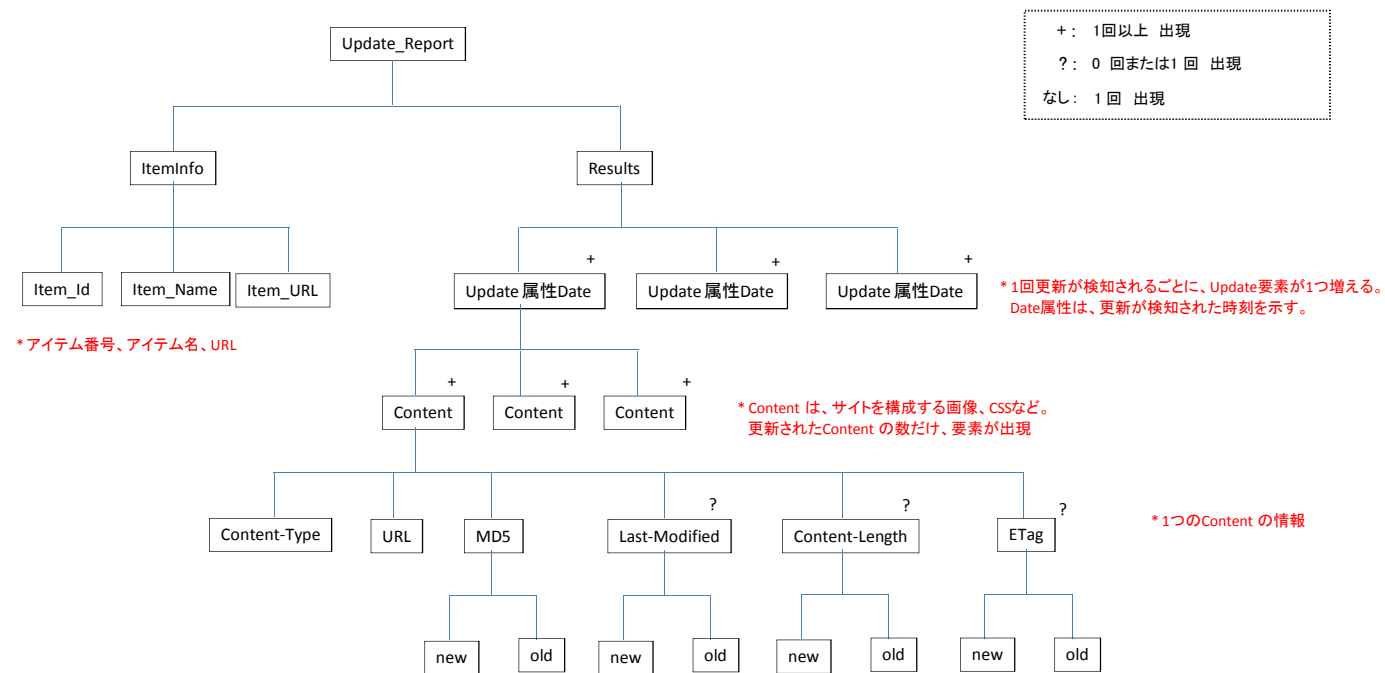
1.3 エラー処理

以下の場合、エラー処理を行う。

					各エラーメッセージを出力
項番	処理ステップ	パターン	処理	戻り値	logへの出力
1	主処理	アイテムレコード取得に失敗した場合	ログにエラーメッセージ出力、終了	空	[Error1] Cannot Get Item Record
2	更新検知	MD5ハッシュ値を取得できなかった場合	ログにエラーメッセージ出力	空	[Error2] Cannot Get MD5 at Detecting Update -> [URL]
3	コンテンツ情報取得	コンテンツ情報取得の際、HTTPアクセスに失敗した場合	ログにエラーメッセージ出力	空	[Error3] HTTP Request Error at Getting Content Information -> [レスポンスコード] [URL]
4	ミラーリング	リダイレクト調査の際、HTTPアクセスに失敗した場合	ログにエラーメッセージ出力	空	[Error4] HTTP Request Error at Checking Redirect -> [レスポンスコード] [アイテム番号] [URL]
5	ミラーリング	wget コマンド実行に失敗した場合	ログにエラーメッセージ出力	空	[Error5] Wget Command Error -> [アイテム番号] [URL]
6	ミラーリング	wget のログファイルのオープンに失敗した場合	ログにエラーメッセージ出力	空	[Error6] Wget Log Open Error -> [アイテム番号] [URL]
7	ミラーリング	wget のログファイルのクローズに失敗した場合	ログにエラーメッセージ出力	空	[Error7] Wget Log Close Error -> [アイテム番号] [URL]
8	ミラーリング	ミラーリングに失敗した場合	ログにエラーメッセージ出力	空	[Error8] Mirroring Error [アイテム番号] [URL] -> [wget のエラーメッセージ]
9	ミラーリング	DBへの登録に失敗した場合	ログにエラーメッセージ出力	空	[Error9] Cannot Insert Into SubItem -> [URL] [Error9] Cannot Insert Into UpdateHistory ->
10	ミラーリング	ミラーリングの際、MD5ハッシュ値を取得できなかった場合	ログにエラーメッセージ出力	空	[Error10] Cannot Get MD5 at Mirroring -> [URL]
11	主処理	出力するXMLファイルのオープンに失敗した場合	ログにエラーメッセージ出力	空	[Error11] XML Open Error -> [ファイル名]
12	主処理	出力するXMLファイルのクローズに失敗した場合	ログにエラーメッセージ出力	空	[Error12] XML Close Error -> [ファイル名]
13	主処理	メール送信に失敗した場合	ログにエラーメッセージ出力	空	[Error13] Mail Send Error

1.4 XMLの構造

XMLの構造は下図のとおり



2. 管理GUI

2.1 ディレクトリ構成

MEDALS管理GUIのディレクトリ構成は以下の通り。

項番	ディレクトリ名	ディレクトリ名	詳細
1	System_Root		CGIを格納するディレクトリ Basic認証用ファイルを格納するディレクトリ (.htaccecc、.htpasswd)
2		pm	Perlモジュールを格納するディレクトリ
3		template	HTMLテンプレートファイルを格納するディレクトリ
4		js	JavaScriptを格納するディレクトリ
5		css	スタイルシートを格納するディレクトリ
6		img	画像ファイルを格納するディレクトリ

サーバへは、このディレクトリ構成でインストールを行う。
インストールが完了したら、設定ファイルを開きサーバ環境に合わせ設定を行う。設定ファイルは【System_Root】/pm/system_config.pm。

```
#-----# DB関連#-----
# DB名
$:dbname = "item_matsumura:10.228.136.121";
# ユーザ名
$:dbuser = "admin";
# パスワード
$:dbpass = "admin";
# データベースポート
$:dbport = 3306;
#-----# HTML関連#-----
# ログインユーザ名
$:login_usr = $ENV{'REMOTE_USER'};
# CSS格納ディレクトリ
$:dir_css = "/css";
# javascript格納ディレクトリ
$:dir_js = "/js";
# イメージファイルディレクトリ
$:dir_img = "/img";
# テンプレート格納ディレクトリ
$:dir_template = "/template";
# バックナンバー格納ディレクトリ
$:dir_histry = "/medals_tool/dummy";
# バックナンバーのリンク先
$:histry_url = "http://cl3253/motomura/medals/medals_tool/dummy";
----- 以下メッセージ関連 -----
※以下は自由にメッセージの変更を行う。
リファラー
$:referer = "http://cl3253/motomura/medals/medals_tool/";
$:referer_full = "http://cl3253/motomura/medals/medals_tool/index.cgi";
```

項番	変数名	詳細
1	\$:dbname	データベース名
2	\$:dbuser	データベースに接続する際のユーザ名
3	\$:dbpass	データベースに接続する際のパスワード
4	\$:login_usr	Basic認証でログインした際のユーザ名を格納(変更不可)
5	\$:dir_css	スタイルシートファイルを格納したディレクトリへのパス
6	\$:dir_js	JavaScriptファイルを格納したディレクトリへのパス
7	\$:dir_img	イメージファイルを格納したディレクトリへのパス
8	\$:dir_template	HTMLのテンプレートファイルを格納したディレクトリへのパス
9	\$:dir_histry	ミラーサイトを格納したディレクトリへのパス
10	\$:histry_url	ミラーサイトへのURL(項番9までのパスを指定)
11	\$:referer	DBの更新を実行できるURL
12	\$:referer_full	DBの更新を実行できるURL

2.2 Basic認証

管理GUIではBasic認証を利用している。

Basic認証では「.htaccess」「.htpasswd」の2つのファイルが必要。このうち「.htaccess」を置いたディレクトリ配下のコンテンツに対してBasic認証を行う。

本システムでは、管理GUIのSystem_Root配下に「.htaccess」「.htpasswd」を配置している。但し、「.htpasswd」は任意のディレクトリに配置することが可能。

2.2.1 .htaccess

【.htaccess】※パーミッションは604に設定

```
<Files ~ "^.(htaccess|htpasswd)$">
deny from all
</Files>
<Files ~ "%.(pm|tmpl)$">
deny from all
</Files>

Options -Indexes
AuthUserFile /WWW/gpmanager/.htpasswd
AuthGroupFile /dev/null
AuthName "Please enter your ID and password"
AuthType Basic
require valid-user
```

項番	属性	説明
1	Files	「.htaccess」「.htpasswd」「.pm」「.tmpl」をWeb上から閲覧不可に設定。
2	Options	オプションの設定。「-Indexes」に設定するとファイルの一覧を表示させない設定となる。
3	AuthUserFile	パスワードファイル(.htpasswd)への絶対パスを指定。
4	AuthGroupFile	ユーザー認証のための、グループファイルを設定。/dev/nullは特に指定がない場合に用いる。
5	AuthName	Basic 認証をかける領域名の指定。
6	AuthType	ユーザ認証機構の種別を指定。
7	require	ユーザ認証の際に、認証させるユーザを指定。Valid-userはすべてのユーザに、認証の機会を与える。
8	order	アクセス制限の設定。

2.2.2 .htpasswd

【.htpasswd】※パーミッションは644に設定

```
admin:c9AaV3lxHQu6.
guest:23yBB/CGR/sg2
```

.htpasswdファイルの作成方法

htpasswd -c .htpasswd admin	←	ファイル名、ユーザ名を指定
New password:	←	パスワードを入力
Re-type new password:	←	もう一度パスワードを入力

ユーザの追加方法

htpasswd .htpasswd medals	←	ファイル名、ユーザ名を指定
New password:	←	パスワードを入力
Re-type new password:	←	もう一度パスワードを入力

※htpasswdコマンドの詳細については、htpasswd -hで出力されるヘルプを参照のこと

ドキュメント名	2. 管理GUI
システム名称	更新検知ツール

画面ID	1	画面名	検索メイン画面
------	---	-----	---------

2.3 画面仕様

2.3.1 検索メイン画面

2.3.1.1 レイアウト定義

画面レイアウト

概要

【A】検索バー

【B】検索結果エリア

【C】特権モードエリア

【D】変更履歴エリア

【A】検索バー

- ・検索キーワードの入力、検索オプションの設定、検索実行の機能。

【B】検索結果エリア

- ・検索結果の表示。

【C】特権モードエリア

- ・特権ユーザのみ利用可能。
- ・アイテムの編集、削除、追加機能。
- ・アイテムグループの編集、削除、追加機能。

【D】変更履歴エリア

- ・選択したアイテムの変更履歴を表示。
- ・リンクよりミラーサイトを表示(別画面表示)。
- ・diff.txtよりテキストのdiffの結果を表示(別画面表示)。
- ・diff.htmlより更新箇所をハイライトしたHTMLを表示(別画面表示)。
- ・更新履歴の種別を設定機能。

入力項目		
項番	項目名	概要
1	検索モード	Item or URL or Comment or ItemID
2	キーワード	検索する単語を入力 or Item IDを入力
3	アイテムグループリスト	All or none or 登録されているグループ名
4	サイト更新日	サイト更新日を入力(指定日以降を検索)
5	サイト更新日	サイト更新日を入力(指定日以前を検索)
6	URL	アイテムのサイトを表示
7	更新日時	ミラーサイトを表示
8	diff.txt	diffの結果のテキスト表示
9	diff.html	変更箇所をハイライトしたHTMLを表示
10	更新履歴Edit	更新履歴の種別設定する画面を表示
11	アイテムEdit	アイテム編集画面を表示
12	アイテムDelete	アイテム削除画面を表示
13	アイテムAdd	アイテム追加画面を表示
14	アイテムグループEdit	アイテムグループ編集画面を表示
15	再取得	選択したアイテムの現地点の変更を確認し、変更があれば取得

※項番2は必須事項。

2.3.1.2 検索処理詳細

検索モード、検索文字列よりtbl_itemテーブルを検索する(部分一致)。検索モードと検索対象カラムの対応は以下の通り。

項番	検索モード	検索対象カラム(tbl_itemテーブル)
1	Item	item_name
2	URL	item_URL
3	Comment	item_URL
4	ItemID	item_id

項番1～3までは、部分一致しておれば、アイテム情報(item_id, item_name, item_URL, parameter, user_id, password, comment, group_id, item_group_name(tbl_item_group))

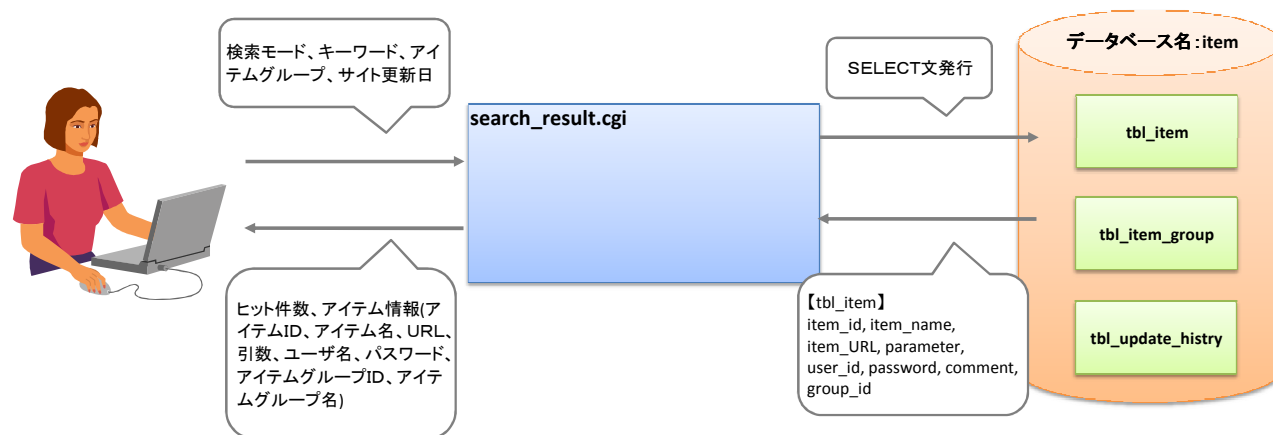
とヒット件数を取得し、表示する。また、アイテムグループの指定があった場合は、item_group_idでの絞り込み検索(AND検索)を行う。さらにサイト更新日を指定した場合は、絞り込み検索(AND検索)を行う。

その際、tbl_update_histryテーブルのdate_updateと比較を行い、指定されたサイト更新日の期間内にデータが存在するアイテム情報のみを取得する。取得したいサイト更新日の開始のみを指定された場合は、指定日以降のデータが存在するアイテム情報のみを、終了のみを指定された場合は、指定日以前にデータが存在するアイテム情報のみを取得する。

項番4については、item_idと完全一致するものを取得する。

表示の際に、データが存在しないものに関しては""(空文字)を表示する。

検索結果はアイテム名の昇順で表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	2
画面名	アイテム編集画面

2.3.2 アイテム編集画面

2.3.2.1 レイアウト定義

画面レイアウト

概要

- ・【B】検索結果エリアより、編集したいアイテムを選択し【C】特権モードエリアに設置されたアイテム「Edit」ボタンを押下することで表示。
- ・アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDALS番号、コメントの8項目が編集可能。
- ・編集終了後「ok」ボタンを押下すると更新情報をDBに反映する(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
- ・「cancel」ボタンを押下すると、編集情報が破棄される。
- ・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。
- ・同じアイテム名かつURL情報が先に登録されている場合、更新情報は反映されない。

入力項目		
項番	項目名	概要
1	アイテム名	アイテム名を入力
2	URL	アイテムのURLを入力
3	引数リスト	0～5
4	ユーザ名	アイテム取得の際に必要となるユーザID
5	パスワード	アイテム取得の際に必要となるパスワード
6	アイテムグループリスト	none or 登録されているグループ名
7	MEDALS番号	アイテムに対応するMEDALSのIDを入力
8	コメント	アイテムに対するコメントを入力
9	ok	アイテムデータを更新(登録内容確認画面を)
10	cancel	アイテムデータ変更を破棄

※項番1～8はデフォルト値として、選択したアイテム情報を表示。

※項番1,2は必須事項。

2.3.2.2 アイテム更新処理詳細

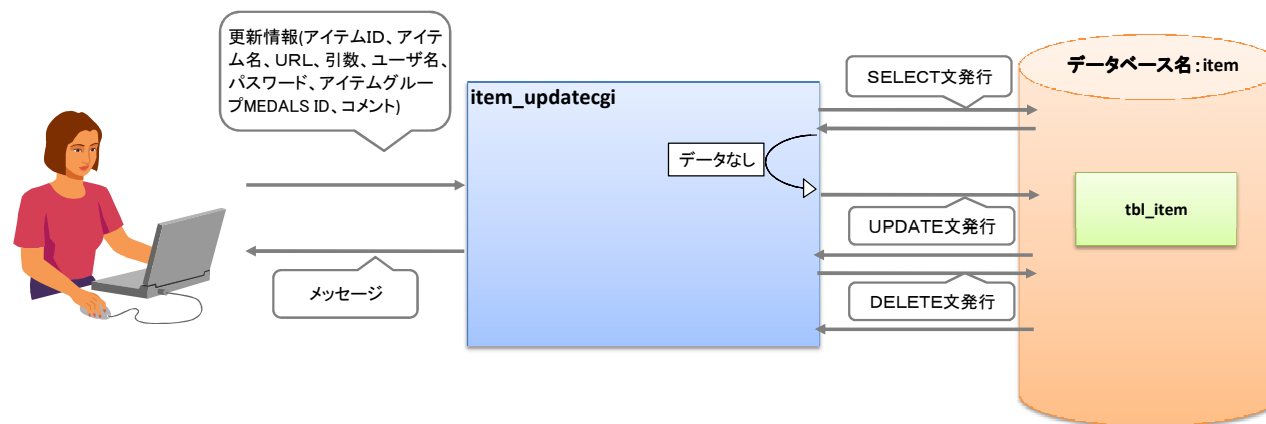
更新情報(アイテムID、アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDLAS ID、コメント)をもとに、tbl_itemテーブルにアクセスし、更新データかどうか確認する。

更新データでない場合(更新画面で変更が行われていない場合)は、「すでに登録済みです」とメッセージを表示する。

上記の結果、更新データであった場合は、tbl_itemテーブルの指定されたアイテムIDの行の更新を行う。

また、tbl_sub_itemテーブルの指定されたアイテムIDの行の削除を行う。

更新・削除が正常に終了した場合は、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	3
画面名	アイテム削除画面

2.3.3 アイテム削除画面

2.3.3.1 レイアウト定義

画面レイアウト

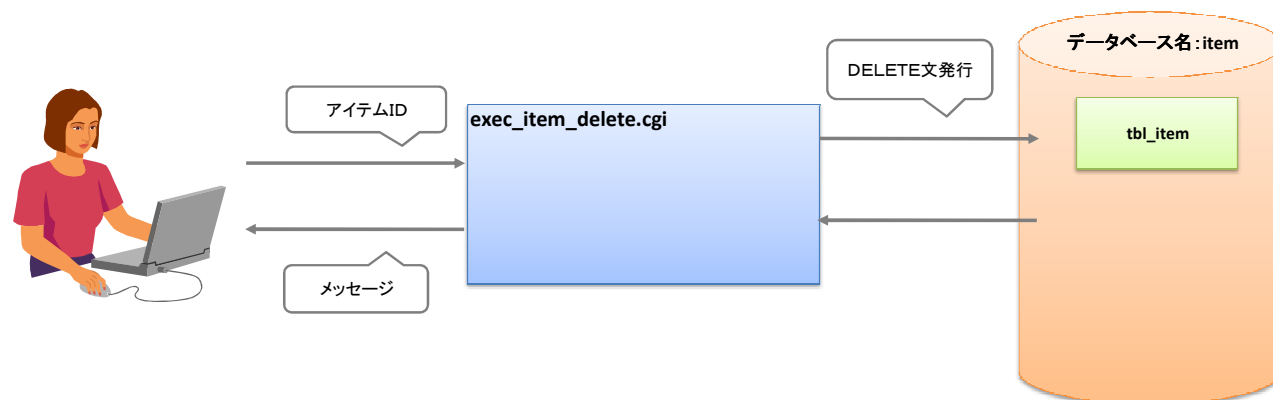
概要

- ・【B】検索結果エリアより、削除したいアイテムを選択し【C】特権モードエリアに設置されたアイテム「Delete」ボタンを押下することで表示。
- ・選択したアイテム情報を削除確認情報として画面に出力。
- ・削除しても良い場合は「ok」ボタンを押下する。押下後削除情報をDBに反映する。
- ・削除を取り消したい場合は「cancel」ボタンを押下する。
- ・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。

入力項目		
項番	項目名	概要
1	ok	アイテムデータを削除
2	cancel	アイテムデータ削除を破棄

2.3.3.2 アイテム削除処理詳細


選択したアイテムのアイテムIDをもとに、tbl_itemテーブルにアクセスし、該当するitem_idの行を削除する。
削除が完了したら、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	4
画面名	アイテム追加画面

2.3.4 アイテム追加画面

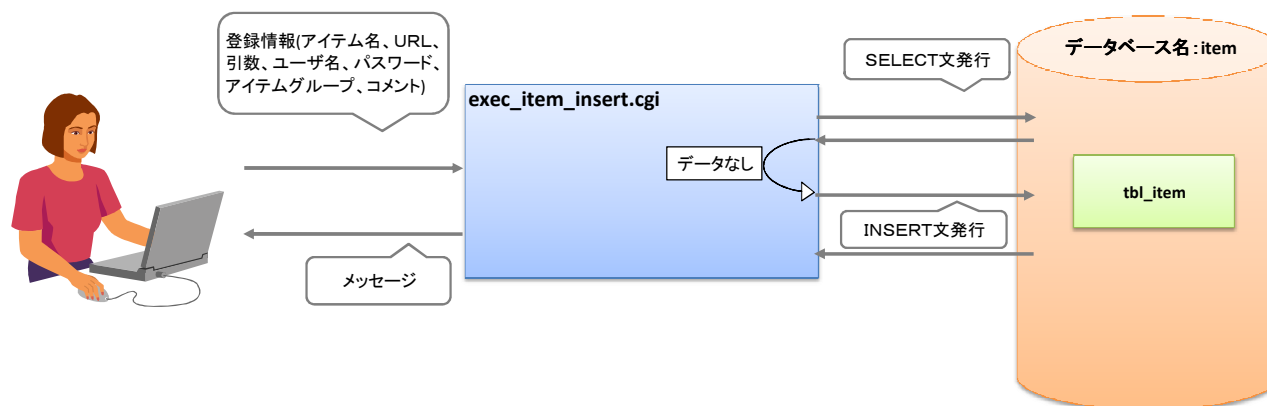
2.3.4.1 レイアウト定義

画面レイアウト	概要																																				
	<ul style="list-style-type: none">・【C】特権モードエリアに設置されたアイテム「Add」ボタンを押下することで表示。・登録したいアイテムの情報を各々のエリアに入力または指定する。・入力終了後「ok」ボタンを押下するとアイテム情報がDBに追加される(「ok」ボタン押下後、登録内容確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。・「ok」ボタンを押下するとDBに反映される。・「cancel」ボタンを押下すると、新規アイテム情報が破棄される。・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。・同じアイテム名かつURL情報が先に登録されている場合、アイテム情報は追加されない。 <table><tr><th colspan="3">入力項目</th></tr><tr><th>項番</th><th>項目名</th><th>概要</th></tr><tr><td>1</td><td>アイテム名</td><td>アイテム名を入力</td></tr><tr><td>2</td><td>URL</td><td>アイテムのURLを入力</td></tr><tr><td>3</td><td>引数リスト</td><td>0～5</td></tr><tr><td>4</td><td>ユーザ名</td><td>アイテム取得時に必要となるユーザID</td></tr><tr><td>5</td><td>パスワード</td><td>アイテム取得時に必要となるパスワード</td></tr><tr><td>6</td><td>アイテムグループリスト</td><td>none or 登録されているグループ名</td></tr><tr><td>7</td><td>MEDALS番号</td><td>アイテムに対応するMEDALSのIDを入力</td></tr><tr><td>8</td><td>コメント</td><td>アイテムに対するコメントを入力</td></tr><tr><td>9</td><td>ok</td><td>アイテムデータを追加 (登録内容確認画面を表示)</td></tr><tr><td>10</td><td>cancel</td><td>アイテムデータ追加を破棄</td></tr></table> <p>※項番1,2は必須事項。</p>	入力項目			項番	項目名	概要	1	アイテム名	アイテム名を入力	2	URL	アイテムのURLを入力	3	引数リスト	0～5	4	ユーザ名	アイテム取得時に必要となるユーザID	5	パスワード	アイテム取得時に必要となるパスワード	6	アイテムグループリスト	none or 登録されているグループ名	7	MEDALS番号	アイテムに対応するMEDALSのIDを入力	8	コメント	アイテムに対するコメントを入力	9	ok	アイテムデータを追加 (登録内容確認画面を表示)	10	cancel	アイテムデータ追加を破棄
入力項目																																					
項番	項目名	概要																																			
1	アイテム名	アイテム名を入力																																			
2	URL	アイテムのURLを入力																																			
3	引数リスト	0～5																																			
4	ユーザ名	アイテム取得時に必要となるユーザID																																			
5	パスワード	アイテム取得時に必要となるパスワード																																			
6	アイテムグループリスト	none or 登録されているグループ名																																			
7	MEDALS番号	アイテムに対応するMEDALSのIDを入力																																			
8	コメント	アイテムに対するコメントを入力																																			
9	ok	アイテムデータを追加 (登録内容確認画面を表示)																																			
10	cancel	アイテムデータ追加を破棄																																			

2.3.4.2 アイテム登録処理詳細

新規データ(アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDALS ID、コメント)をtbl_itemテーブルに登録する。

登録が完了したら、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	5
画面名	アイテムグループ編集画面

2.3.5 アイテムグループ編集画面

2.3.5.1 レイアウト定義

画面レイアウト

概要

・【C】特権モードエリアに設置されたアイテムグループ「Edit」ボタンを押下することで表示。
・アイテムグループの編集、削除、追加が可能。
・同じアイテムグループ名が先に登録されている場合、アイテムグループ情報は更新・追加されない。

【アイテムグループ編集】

・アイテムグループリストより編集したいアイテムグループ名を選択する。
・グループ名、コメントが編集可能。編集後、「ok」ボタンを押下すると、DBに反映される(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
・「cancel」ボタンを押下すると、編集情報が破棄される。

【アイテムグループ削除】

・アイテムグループリストより削除したいアイテムグループ名を選択し、「delete」ボタンを押下する。
・「delete」ボタン押下後、削除確認画面が表示される。削除しても良い場合は「ok」ボタンを押下する。押下後削除情報をDBに反映する。
・削除するアイテムグループに所属するアイテムが存在した場合、これらのアイテムグループ情報を「0(どのグループにも所属していない)」に更新する。

【アイテムグループ追加】

・アイテムグループリストより「新規登録」を選択。
・登録したいグループ名、コメントと入力し、「ok」ボタンを押下すると、DBに反映される(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
・「cancel」ボタンを押下すると、追加情報が破棄される。

入力項目

項番	項目名	概要
1	アイテムグループリスト	「新規登録」or 登録されているグループ名
2	グループ名	グループ名を入力
3	アイテムグループ	アイテムグループに対するコメントを入力
4	ok	アイテムデータを更新または追加 (登録内容確認画面を表示)
5	cancel	アイテムグループ変更、削除、追加を破棄

※項番2は必須事項。

2.3.5.2 アイテムグループ編集処理詳細

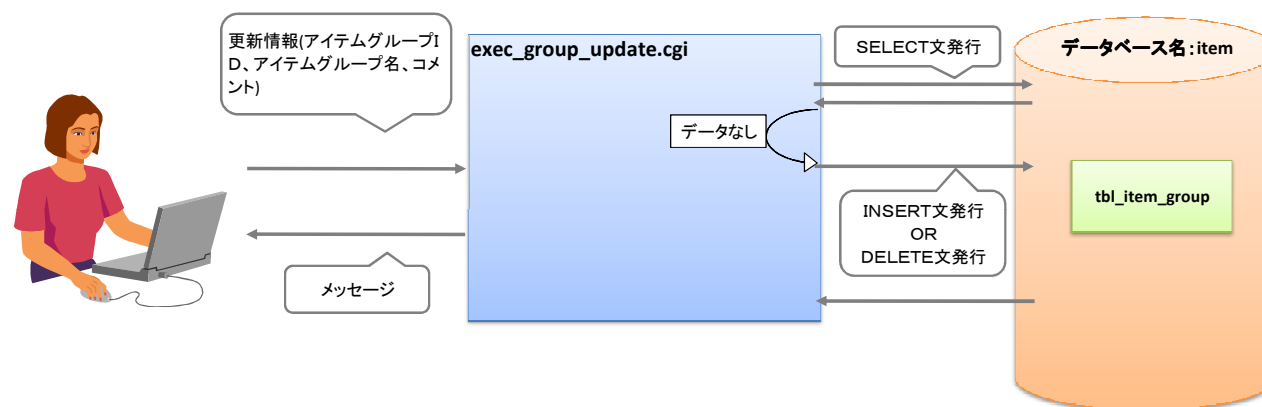
(1) アイテムグループの更新および新規登録

更新、新規登録情報(アイテムグループ名、コメント)をもとに、tbl_item_groupテーブルにアクセスし、更新データかどうか確認する。

更新データでない場合(編集画面で更新が行われていない場合)は、「すでに登録済みです」とメッセージを表示する。

上記の結果、更新データであった場合、アイテムグループIDが「new」の場合は、tbl_item_groupテーブルへの新規登録を行い、それ以外の場合は更新を行う。

更新、登録が完了したら、完了メッセージを表示する。



(2)アイテムグループの削除

選択したアイテムグループのアイテムグループIDをもとに、tbl_itemテーブルにアクセスし、選択したアイテムグループに属するアイテムが存在するかどうかを確認する。
属するアイテムが存在した場合、「アイテムテーブルにデータが存在します。XXX(選択したアイテムグループ名)を削除して良いですか?」とメッセージを表示する。

属するアイテムが存在しない場合「XXX(選択したアイテムグループ名)を削除して良いですか?」とメッセージを表示する。

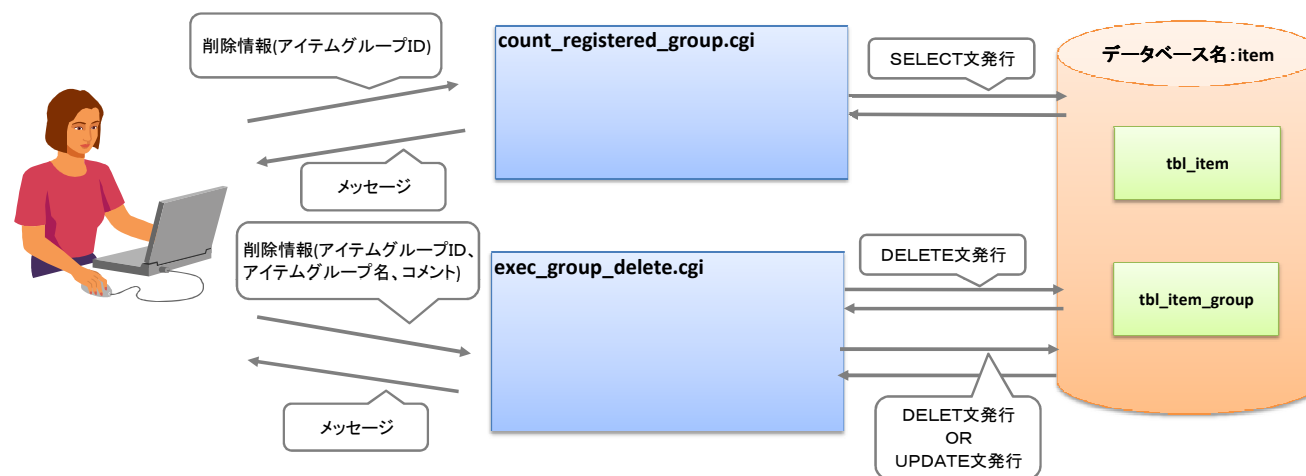
「アイテムグループに属するアイテムデータも同時に削除する」というチェックボックスを表示する。

メッセージ表示画面で削除「ok」ボタンが押下された場合、選択したアイテムグループIDをもとに、tbl_item_groupテーブルにアクセスし、該当するitem_group_idの行を削除する。

「アイテムグループに属するアイテムデータも同時に削除する」というチェックボックスにチェックが入っていた場合は、アイテムグループに属するアイテムの行を削除する。

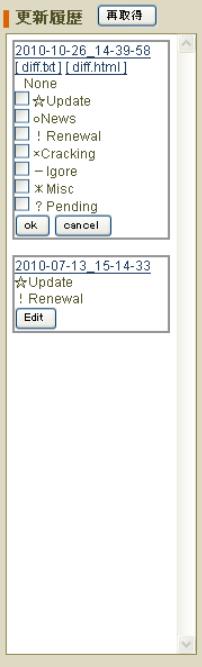
上記以外の場合は、tbl_itemテーブルにアクセスし、選択したアイテムグループに属するアイテムのitem_group_idを「0(どのグループにも属さない)」に更新する。

削除、更新が完了したら、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール

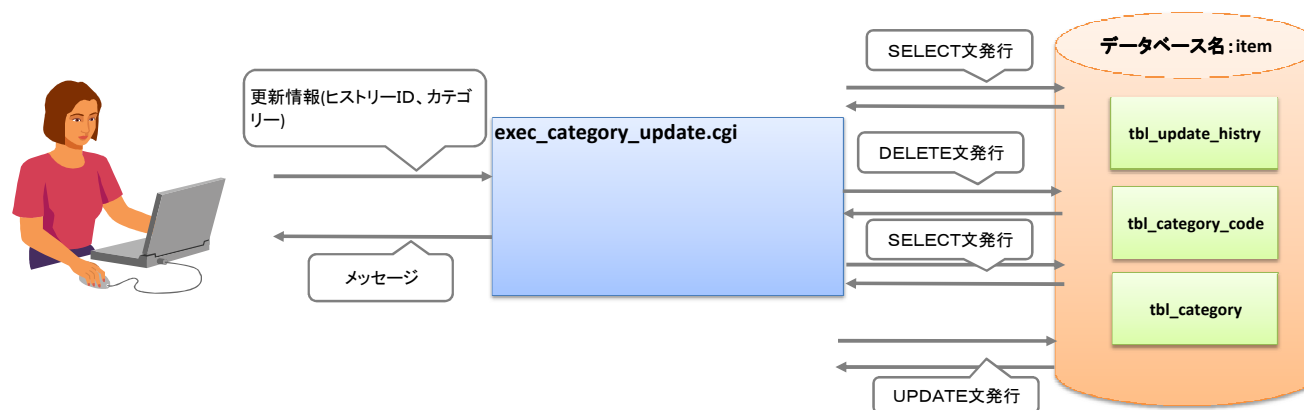
画面ID	6	画面名	更新履歴編集画面
------	---	-----	----------

2.3.6 更新履歴編集画面	
2.3.6.1 レイアウト定義	
画面レイアウト	概要
	<ul style="list-style-type: none">・【D】変更履歴エリアのEditボタンを押下することにより、更新履歴の種別を設定するためのチェックボックス群を表示する。・更新履歴にセットしたい種別を選択後、「ok」ボタンを押下することによりDBへ反映される。・全てのチェックボックスにチェックを入れなかった場合は、Noneに設定される。・「cancel」ボタンを押下することにより、編集情報を破棄できる。・チェックボックス群は、tbl_category_codeテーブルに設定されているものが表示される。

(1)更新履歴の種別の更新

選択された更新履歴の履歴IDをもとに、tbl_update_histryにデータが存在するか確認する。
データが存在した場合、履歴IDをもとに、tbl_categoryにアクセスし、該当する行を削除する。

選択された種別が1つ以上あれば、その種別のコードがtbl_category_codeに存在することを確認する。
データが存在した場合は、更新情報(履歴ID、カテゴリー)をもとに、tbl_categoryにデータを登録する。
選択された種別が1つもなかった場合は、tbl_categoryに'00'を登録する。



ドキュメント名	システム仕様書
システム名称	更新検知ツール

3. DB設計

DB一覧表

項番	DB名	説明
1	アイテムDB	更新検知ツールで使用するDB

3.1 アイテムDB

3.1.1 テーブル一覧表

DB名	item	
項番	テーブル名	説明
1	tbl_item_group	アイテムグループを管理するテーブル
2	tbl_item	アイテムを管理するテーブル
3	tbl_sub_item	アイテムを構成するコンテンツ(画像、CSSなど)を管理するテーブル
4	tbl_update_histroy	アイテムの更新履歴を格納したテーブル
5	tbl_category	更新履歴の種別を格納したテーブル
6	tbl_category_code	更新履歴の種別のコードを管理するテーブル

3.1.2 テーブル設計

テーブル名		tbl_item_group					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	item_gruop_id	int unsigned		○	○		アイテムグループ番号
2	item_gruop_name	varchar	255				アイテムグループ名
3	comment	text					説明

テーブル名		tbl_item					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	item_id	int unsigned		○	○		アイテム番号
2	item_name	varchar	255	○			アイテム名
3	item_URL	varchar	255	○			URL
4	follow_link	int					POSTで渡す引数
5	user_id	text		○			Basic認証のユーザID
6	password	text		○			Basic認証のパスワード
7	comment	text		○			説明
8	item_group_id	int					外部キー
9	medals_id	int		○			MEDALS番号
10	parameter	text					未使用

ドキュメント名	システム仕様書
システム名称	更新検知ツール

テーブル名		tbl sub item					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	sub_item_id	bigint unsigned		○	○		サブアイテム番号
2	sub_item_name	text					サブアイテム名
3	sub_item_URL	text					URL
4	last_modified	text					HTTPヘッダの最終更新日
5	content_length	text					HTTPヘッダのContent-Length
6	e_tag	text					HTTPヘッダのE-tag
7	item_id	int unsigned					外部キー

テーブル名		tbl update histry					
項番	カラム名		桁数	NOT NULL	主キー	インデックス	説明
1	id	int unsigned		○	○		連番
2	item_id	int unsigned					アイテム番号
3	date_update	int					アイテムが更新された日付(2009/10/01→20091001)
4	time_update	int					アイテムが更新された時間(22:10:01→221001)

テーブル名		tbl category					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	id	int unsigned		○	○		カテゴリー番号
2	histry_id	int unsigned					ヒストリーID
3	category_code	varchar	2				カテゴリーコード

テーブル名		tbl category code					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	code	varchar	2	○	○		カテゴリーコード
2	name	varchar	255				カテゴリー名

2. 管理GUI

2.1 ディレクトリ構成

MEDALS管理GUIのディレクトリ構成は以下の通り。

項番	ディレクトリ名	ディレクトリ名	詳細
1	System_Root		CGIを格納するディレクトリ Basic認証用ファイルを格納するディレクトリ (.htaccecc、.htpasswd)
2		pm	Perlモジュールを格納するディレクトリ
3		template	HTMLテンプレートファイルを格納するディレクトリ
4		js	JavaScriptを格納するディレクトリ
5		css	スタイルシートを格納するディレクトリ
6		img	画像ファイルを格納するディレクトリ

サーバへは、このディレクトリ構成でインストールを行う。
インストールが完了したら、設定ファイルを開きサーバ環境に合わせ設定を行う。設定ファイルは【System_Root】/pm/system_config.pm。

```
#-----# DB関連#-----
# DB名
$:dbname = "item_matsumura:10.228.136.121";
# ユーザ名
$:dbuser = "admin";
# パスワード
$:dbpass = "admin";
# データベースポート
$:dbport = 3306;
#-----# HTML関連#-----
# ログインユーザ名
$:login_usr = $ENV{'REMOTE_USER'};
# CSS格納ディレクトリ
$:dir_css = "/css";
# javascript格納ディレクトリ
$:dir_js = "/js";
# イメージファイルディレクトリ
$:dir_img = "/img";
# テンプレート格納ディレクトリ
$:dir_template = "/template";
# バックナンバー格納ディレクトリ
$:dir_histry = "/medals_tool/dummy";
# バックナンバーのリンク先
$:histry_url = "http://cl3253/motomura/medals/medals_tool/dummy";
----- 以下メッセージ関連 -----
※以下は自由にメッセージの変更を行う。
リファラー
$:referer = "http://cl3253/motomura/medals/medals_tool/";
$:referer_full = "http://cl3253/motomura/medals/medals_tool/index.cgi";
```

項番	変数名	詳細
1	\$:dbname	データベース名
2	\$:dbuser	データベースに接続する際のユーザ名
3	\$:dbpass	データベースに接続する際のパスワード
4	\$:login_usr	Basic認証でログインした際のユーザ名を格納(変更不可)
5	\$:dir_css	スタイルシートファイルを格納したディレクトリへのパス
6	\$:dir_js	JavaScriptファイルを格納したディレクトリへのパス
7	\$:dir_img	イメージファイルを格納したディレクトリへのパス
8	\$:dir_template	HTMLのテンプレートファイルを格納したディレクトリへのパス
9	\$:dir_histry	ミラーサイトを格納したディレクトリへのパス
10	\$:histry_url	ミラーサイトへのURL(項番9までのパスを指定)
11	\$:referer	DBの更新を実行できるURL
12	\$:referer_full	DBの更新を実行できるURL

2.2 Basic認証

管理GUIではBasic認証を利用している。

Basic認証では「.htaccess」「.htpasswd」の2つのファイルが必要。このうち「.htaccess」を置いたディレクトリ配下のコンテンツに対してBasic認証を行う。

本システムでは、管理GUIのSystem_Root配下に「.htaccess」、「.htpasswd」を配置している。但し、「.htpasswd」は任意のディレクトリに配置することが可能。

2.2.1 .htaccess

【.htaccess】※パーミッションは604に設定

```
<Files ~ "^.(htaccess|htpasswd)$">
deny from all
</Files>
<Files ~ "%.(pm|tmpl)$">
deny from all
</Files>

Options -Indexes
AuthUserFile /WWW/gpmanager/.htpasswd
AuthGroupFile /dev/null
AuthName "Please enter your ID and password"
AuthType Basic
require valid-user
```

項番	属性	説明
1	Files	「.htaccess」「.htpasswd」「.pm」「.tmpl」をWeb上から閲覧不可に設定。
2	Options	オプションの設定。「-Indexes」に設定するとファイルの一覧を表示させない設定となる。
3	AuthUserFile	パスワードファイル(.htpasswd)への絶対パスを指定。
4	AuthGroupFile	ユーザー認証のための、グループファイルを設定。/dev/nullは特に指定がない場合に用いる。
5	AuthName	Basic 認証をかける領域名の指定。
6	AuthType	ユーザ認証機構の種別を指定。
7	require	ユーザ認証の際に、認証させるユーザを指定。Valid-userはすべてのユーザに、認証の機会を与える。
8	order	アクセス制限の設定。

2.2.2 .htpasswd

【.htpasswd】※パーミッションは644に設定

```
admin:c9AaV3lxHQu6.
guest:23yBB/CGR/sg2
```

.htpasswdファイルの作成方法

htpasswd -c .htpasswd admin	←	ファイル名、ユーザ名を指定
New password:	←	パスワードを入力
Re-type new password:	←	もう一度パスワードを入力

ユーザの追加方法

htpasswd .htpasswd medals	←	ファイル名、ユーザ名を指定
New password:	←	パスワードを入力
Re-type new password:	←	もう一度パスワードを入力

※htpasswdコマンドの詳細については、htpasswd -hで出力されるヘルプを参照のこと

ドキュメント名	2. 管理GUI
システム名称	更新検知ツール

画面ID	1	画面名	検索メイン画面
------	---	-----	---------

2.3 画面仕様

2.3.1 検索メイン画面

2.3.1.1 レイアウト定義

画面レイアウト

概要

【A】検索バー

- ・検索キーワードの入力、検索オプションの設定、検索実行の機能。
- 【B】検索結果エリア
- ・検索結果の表示。
- 【C】特権モードエリア
- ・特権ユーザのみ利用可能。
- ・アイテムの編集、削除、追加機能。
- ・アイテムグループの編集、削除、追加機能。

【D】変更履歴エリア

- ・選択したアイテムの変更履歴を表示。
- ・リンクよりミラーサイトを表示(別画面表示)。
- ・diff.txtよりテキストのdiffの結果を表示(別画面表示)。
- ・diff.htmlより更新箇所をハイライトしたHTMLを表示(別画面表示)。
- ・更新履歴の種別を設定機能。

【D】変更履歴エリア

入力項目

項番	項目名	概要
1	検索モード	Item or URL or Comment or ItemID
2	キーワード	検索する単語を入力 or Item IDを入力
3	アイテムグループリスト	All or none or 登録されているグループ名
4	サイト更新日	サイト更新日を入力(指定日以降を検索)
5	サイト更新日	サイト更新日を入力(指定日以前を検索)
6	URL	アイテムのサイトを表示
7	更新日時	ミラーサイトを表示
8	diff.txt	diffの結果のテキスト表示
9	diff.html	変更箇所をハイライトしたHTMLを表示
10	更新履歴Edit	更新履歴の種別設定する画面を表示
11	アイテムEdit	アイテム編集画面を表示
12	アイテムDelete	アイテム削除画面を表示
13	アイテムAdd	アイテム追加画面を表示
14	アイテムグループEdit	アイテムグループ編集画面を表示
15	再取得	選択したアイテムの現地点の変更を確認し、変更があれば取得

※項番2は必須事項。

2.3.1.2 検索処理詳細

検索モード、検索文字列よりtbl_itemテーブルを検索する(部分一致)。検索モードと検索対象カラムの対応は以下の通り。

項番	検索モード	検索対象カラム(tbl_itemテーブル)
1	Item	item_name
2	URL	item_URL
3	Comment	item_URL
4	ItemID	item_id

項番1～3までは、部分一致しておれば、アイテム情報(item_id, item_name, item_URL, parameter, user_id, password, comment, group_id, item_group_name(tbl_item_group))

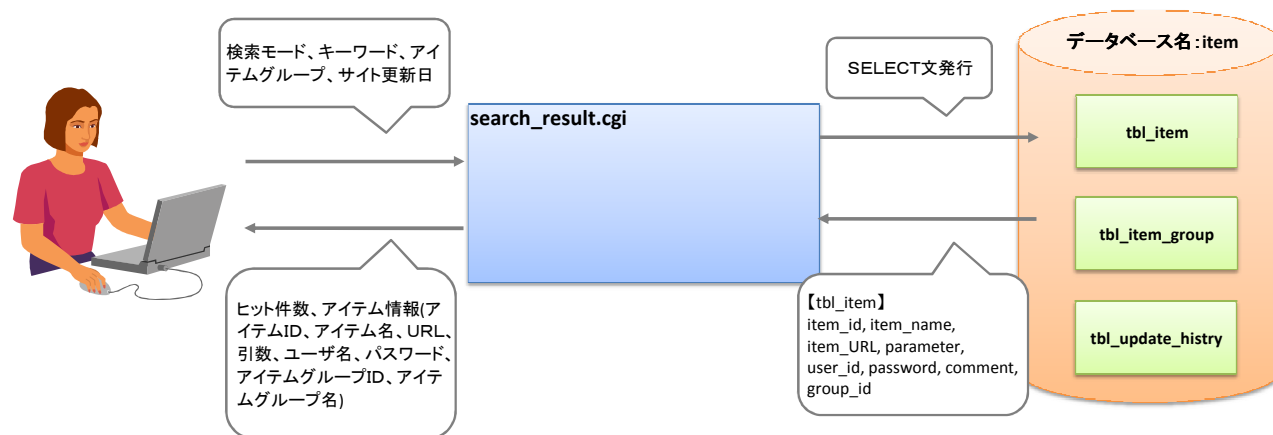
とヒット件数を取得し、表示する。また、アイテムグループの指定があった場合は、item_group_idでの絞り込み検索(AND検索)を行う。さらにサイト更新日を指定した場合は、絞り込み検索(AND検索)を行う。

その際、tbl_update_histryテーブルのdate_updateと比較を行い、指定されたサイト更新日の期間内にデータが存在するアイテム情報のみを取得する。取得したいサイト更新日の開始のみを指定された場合は、指定日以降のデータが存在するアイテム情報のみを、終了のみを指定された場合は、指定日以前にデータが存在するアイテム情報のみを取得する。

項番4については、item_idと完全一致するものを取得する。

表示の際に、データが存在しないものに関しては""(空文字)を表示する。

検索結果はアイテム名の昇順で表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	2
画面名	アイテム編集画面

2.3.2 アイテム編集画面

2.3.2.1 レイアウト定義

画面レイアウト

The screenshot shows the MEDALS database portal interface. At the top, there's a search bar with 'Item' selected. Below it, a list of items is displayed, including 'Evola', 'ExCatDB', 'FORTE', 'G-compas', 'GENIUS II', and 'GGDR'. A modal window is open for editing 'Evola'. The form contains fields for 'アイテム番号' (1), 'アイテム名' (Evola), 'URL' (http://wwwh-invitational.jp/evola/search.html), '引数' (0), 'ユーザ名' (empty), 'パスワード' (empty), 'アイテムグループ' (none), 'MEDALS 番号' (1), and 'コメント' (empty). There are 'ok' and 'cancel' buttons at the bottom of the modal. The background shows the '更新履歴' (Update History) section with a table of updates.

概要

- ・【B】検索結果エリアより、編集したいアイテムを選択し【C】特権モードエリアに設置されたアイテム「Edit」ボタンを押下することで表示。
- ・アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDALS番号、コメントの8項目が編集可能。
- ・編集終了後「ok」ボタンを押下すると更新情報をDBに反映する(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
- ・「cancel」ボタンを押下すると、編集情報が破棄される。
- ・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。
- ・同じアイテム名かつURL情報が先に登録されている場合、更新情報は反映されない。

入力項目		
項番	項目名	概要
1	アイテム名	アイテム名を入力
2	URL	アイテムのURLを入力
3	引数リスト	0～5
4	ユーザ名	アイテム取得の際に必要なユーザID
5	パスワード	アイテム取得の際に必要なパスワード
6	アイテムグループリスト	none or 登録されているグループ名
7	MEDALS番号	アイテムに対応するMEDALSのIDを入力
8	コメント	アイテムに対するコメントを入力
9	ok	アイテムデータを更新(登録内容確認画面を)
10	cancel	アイテムデータ変更を破棄

※項番1～8はデフォルト値として、選択したアイテム情報を表示。

※項番1,2は必須事項。

2.3.2.2 アイテム更新処理詳細

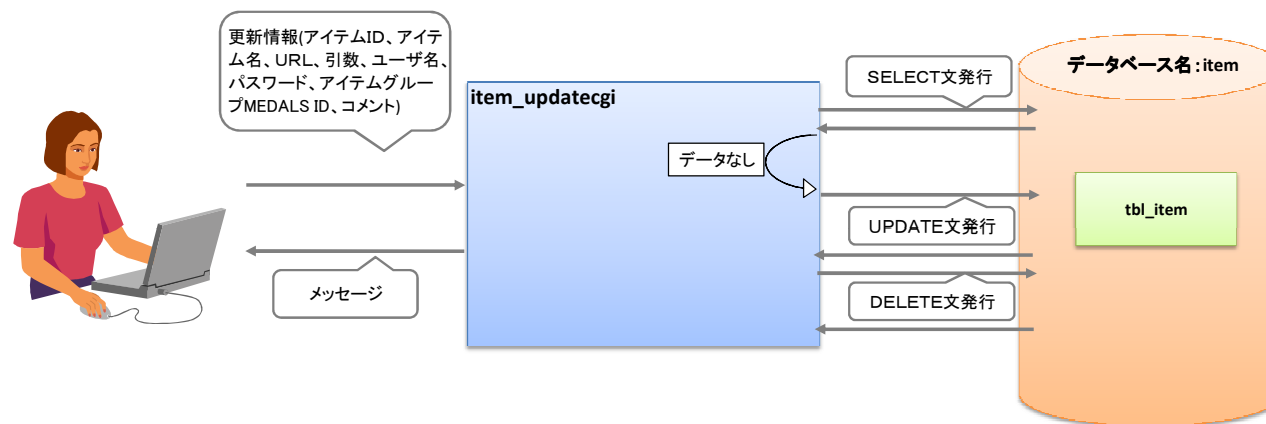
更新情報(アイテムID、アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDLAS ID、コメント)をもとに、tbl_itemテーブルにアクセスし、更新データかどうか確認する。

更新データでない場合(更新画面で変更が行われていない場合)は、「すでに登録済みです」とメッセージを表示する。

上記の結果、更新データであった場合は、tbl_itemテーブルの指定されたアイテムIDの行の更新を行う。

また、tbl_sub_itemテーブルの指定されたアイテムIDの行の削除を行う。

更新・削除が正常に終了した場合は、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	3
画面名	アイテム削除画面

2.3.3 アイテム削除画面

2.3.3.1 レイアウト定義

画面レイアウト

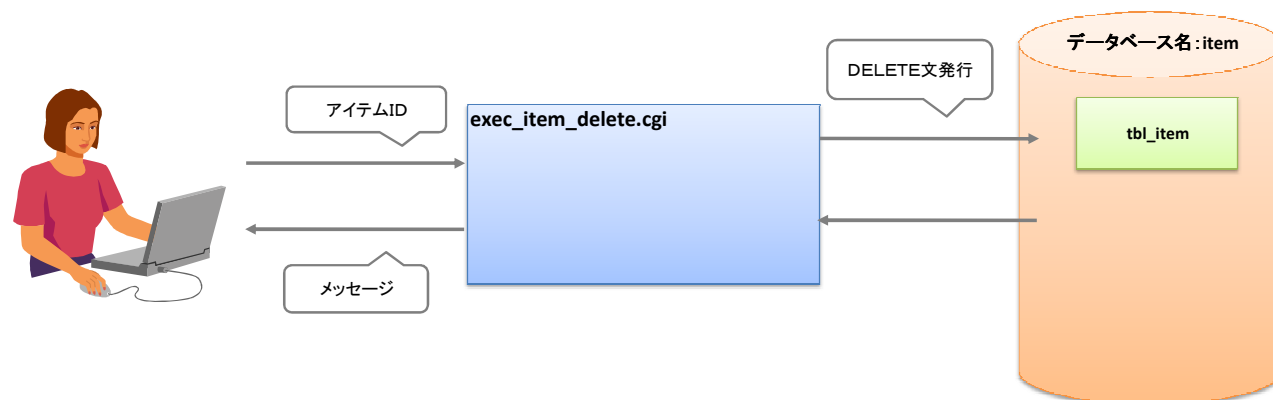
概要

- ・【B】検索結果エリアより、削除したいアイテムを選択し【C】特権モードエリアに設置されたアイテム「Delete」ボタンを押下することで表示。
- ・選択したアイテム情報を削除確認情報として画面に出力。
- ・削除しても良い場合は「ok」ボタンを押下する。押下後削除情報をDBに反映する。
- ・削除を取り消したい場合は「cancel」ボタンを押下する。
- ・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。

入力項目		
項番	項目名	概要
1	ok	アイテムデータを削除
2	cancel	アイテムデータ削除を破棄

2.3.3.2 アイテム削除処理詳細

選択したアイテムのアイテムIDをもとに、tbl_itemテーブルにアクセスし、該当するitem_idの行を削除する。
削除が完了したら、完了メッセージを表示する。

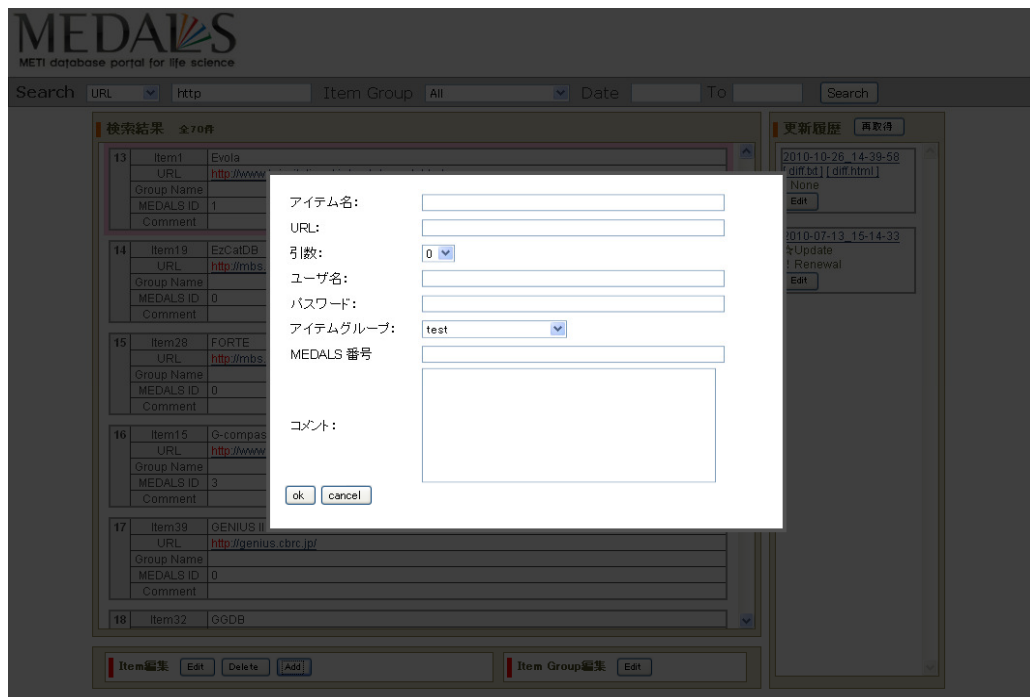


ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	4
画面名	アイテム追加画面

2.3.4 アイテム追加画面

2.3.4.1 レイアウト定義

画面レイアウト



MEDALS
METI database portal for life science

Search URL http Item Group All Date To Search

検索結果 全70件

13 Item1 Evola
URL http://www.
Group Name
MEDALS ID 1
Comment

14 Item19 EzCatDB
URL http://mbs.
Group Name
MEDALS ID 0
Comment

15 Item28 FORTE
URL http://mbs.
Group Name
MEDALS ID 0
Comment

16 Item15 G-compas
URL http://www.
Group Name
MEDALS ID 3
Comment

17 Item39 GENIUS II
URL http://genius.cbrc.jp/
Group Name
MEDALS ID 0
Comment

18 Item32 GGDB

アイテム名:
URL:
引数: 0
ユーザ名:
パスワード:
アイテムグループ: test
MEDALS 番号:
コメント:
ok cancel

更新履歴 再取得
2010-10-26 14:39:58
diff bit | diff html
None
Edit
2010-07-13 15:14:33
Update
Renewal
Edit

Item編集 Edit Delete Add Item Group編集 Edit

概要

- ・【C】特権モードエリアに設置されたアイテム「Add」ボタンを押下することで表示。
- ・登録したいアイテムの情報を各々のエリアに入力または指定する。
- ・入力終了後「ok」ボタンを押下するとアイテム情報がDBに追加される(「ok」ボタン押下後、登録内容確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
- ・「ok」ボタンを押下するとDBに反映される。
- ・「cancel」ボタンを押下すると、新規アイテム情報が破棄される。
- ・DBの更新が終了後、【B】検索結果エリアに結果を反映する(再検索を行う)。
- ・同じアイテム名かつURL情報が先に登録されている場合、アイテム情報は追加されない。

入力項目

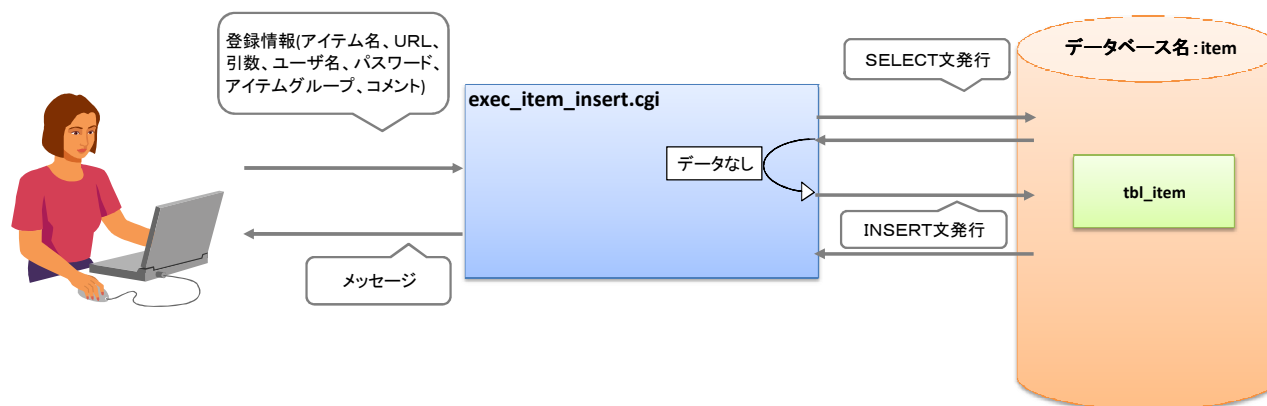
項番	項目名	概要
1	アイテム名	アイテム名を入力
2	URL	アイテムのURLを入力
3	引数リスト	0～5
4	ユーザ名	アイテム取得時に必要となるユーザID
5	パスワード	アイテム取得時に必要となるパスワード
6	アイテムグループリスト	none or 登録されているグループ名
7	MEDALS番号	アイテムに対応するMEDALSのIDを入力
8	コメント	アイテムに対するコメントを入力
9	ok	アイテムデータを追加 (登録内容確認画面を表示)
10	cancel	アイテムデータ追加を破棄

※項番1,2は必須事項。

2.3.4.2 アイテム登録処理詳細

新規データ(アイテム名、URL、引数、ユーザ名、パスワード、アイテムグループ、MEDALS ID、コメント)をtbl_itemテーブルに登録する。

登録が完了したら、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール
画面ID	5
画面名	アイテムグループ編集画面

2.3.5 アイテムグループ編集画面

2.3.5.1 レイアウト定義

画面レイアウト

概要

・【C】特権モードエリアに設置されたアイテムグループ「Edit」ボタンを押下することで表示。
・アイテムグループの編集、削除、追加が可能。
・同じアイテムグループ名が先に登録されている場合、アイテムグループ情報は更新・追加されない。

【アイテムグループ編集】

・アイテムグループリストより編集したいアイテムグループ名を選択する。
・グループ名、コメントが編集可能。編集後、「ok」ボタンを押下すると、DBに反映される(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
・「cancel」ボタンを押下すると、編集情報が破棄される。

【アイテムグループ削除】

・アイテムグループリストより削除したいアイテムグループ名を選択し、「delete」ボタンを押下する。
・「delete」ボタン押下後、削除確認画面が表示される。削除しても良い場合は「ok」ボタンを押下する。押下後削除情報をDBに反映する。
・削除するアイテムグループに所属するアイテムが存在した場合、これらのアイテムグループ情報を「0(どのグループにも所属していない)」に更新する。

【アイテムグループ追加】

・アイテムグループリストより「新規登録」を選択。
・登録したいグループ名、コメントと入力し、「ok」ボタンを押下すると、DBに反映される(「ok」ボタン押下後変更確認画面が表示され、再度「ok」ボタンを押下するとDBに反映される)。
・「cancel」ボタンをうを押下すると、追加情報が破棄される。

入力項目

項番	項目名	概要
1	アイテムグループリスト	「新規登録」 or 登録されているグループ名
2	グループ名	グループ名を入力
3	アイテムグループ	アイテムグループに対するコメントを入力
4	ok	アイテムデータを更新または追加 (登録内容確認画面を表示)
5	cancel	アイテムグループ変更、削除、追加を破棄

※項番2は必須事項。

2.3.5.2 アイテムグループ編集処理詳細

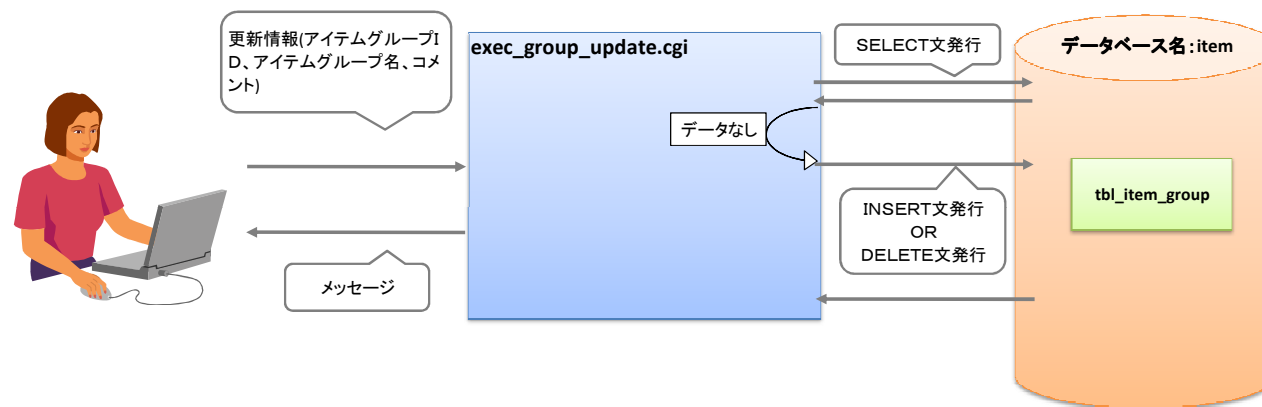
(1) アイテムグループの更新および新規登録

更新、新規登録情報(アイテムグループ名、コメント)をもとに、tbl_item_groupテーブルにアクセスし、更新データかどうか確認する。

更新データでない場合(編集画面で更新が行われていない場合)は、「すでに登録済みです」とメッセージを表示する。

上記の結果、更新データであった場合、アイテムグループIDが「new」の場合は、tbl_item_groupテーブルへの新規登録を行い、それ以外の場合は更新を行う。

更新、登録が完了したら、完了メッセージを表示する。



(2)アイテムグループの削除

選択したアイテムグループのアイテムグループIDをもとに、tbl_itemテーブルにアクセスし、選択したアイテムグループに属するアイテムが存在するかどうかを確認する。
属するアイテムが存在した場合、「アイテムテーブルにデータが存在します。XXX(選択したアイテムグループ名)を削除して良いですか?」とメッセージを表示する。

属するアイテムが存在しない場合「XXX(選択したアイテムグループ名)を削除して良いですか?」とメッセージを表示する。

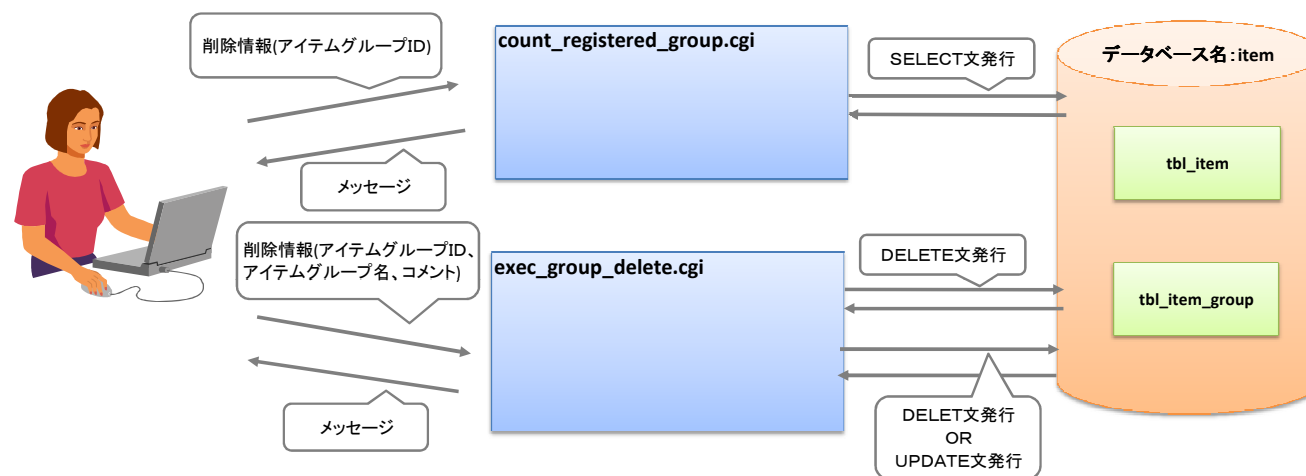
「アイテムグループに属するアイテムデータも同時に削除する」というチェックボックスを表示する。

メッセージ表示画面で削除「ok」ボタンが押下された場合、選択したアイテムグループIDをもとに、tbl_item_groupテーブルにアクセスし、該当するitem_group_idの行を削除する。

「アイテムグループに属するアイテムデータも同時に削除する」というチェックボックスにチェックが入っていた場合は、アイテムグループに属するアイテムの行を削除する。

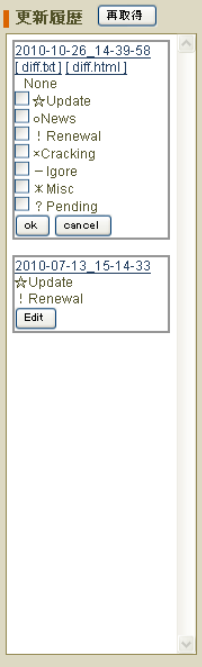
上記以外の場合は、tbl_itemテーブルにアクセスし、選択したアイテムグループに属するアイテムのitem_group_idを「0(どのグループにも属さない)」に更新する。

削除、更新が完了したら、完了メッセージを表示する。



ドキュメント名	2. 管理GUI
システム名称	更新検知ツール

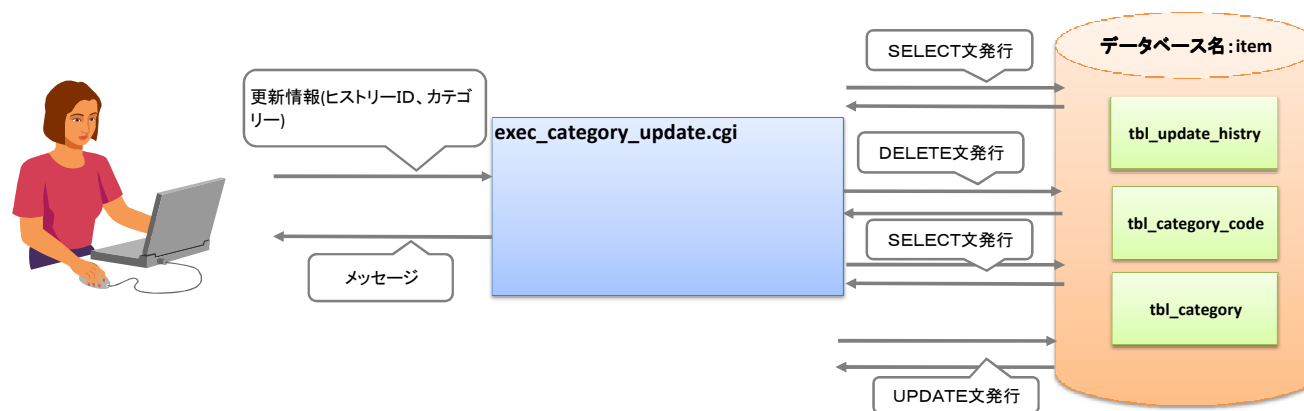
画面ID	6	画面名	更新履歴編集画面
------	---	-----	----------

2.3.6 更新履歴編集画面	
2.3.6.1 レイアウト定義	
画面レイアウト	概要
	<ul style="list-style-type: none">・【D】変更履歴エリアのEditボタンを押下することにより、更新履歴の種別を設定するためのチェックボックス群を表示する。・更新履歴にセットしたい種別を選択後、「ok」ボタンを押下することによりDBへ反映される。・全てのチェックボックスにチェックを入れなかった場合は、Noneに設定される。・「cancel」ボタンを押下することにより、編集情報を破棄できる。・チェックボックス群は、tbl_category_codeテーブルに設定されているものが表示される。

(1)更新履歴の種別の更新

選択された更新履歴の履歴IDをもとに、tbl_update_histryにデータが存在するか確認する。
データが存在した場合、履歴IDをもとに、tbl_categoryにアクセスし、該当する行を削除する。

選択された種別が1つ以上あれば、その種別のコードがtbl_category_codeに存在することを確認する。
データが存在した場合は、更新情報(履歴ID、カテゴリー)をもとに、tbl_categoryにデータを登録する。
選択された種別が1つもなかった場合は、tbl_categoryに'00'を登録する。



ドキュメント名	システム仕様書
システム名称	更新検知ツール

3. DB設計

DB一覧表

項番	DB名	説明
1	アイテムDB	更新検知ツールで使用するDB

3.1 アイテムDB

3.1.1 テーブル一覧表

DB名	item	
項番	テーブル名	説明
1	tbl_item_group	アイテムグループを管理するテーブル
2	tbl_item	アイテムを管理するテーブル
3	tbl_sub_item	アイテムを構成するコンテンツ(画像、CSSなど)を管理するテーブル
4	tbl_update_histroy	アイテムの更新履歴を格納したテーブル
5	tbl_category	更新履歴の種別を格納したテーブル
6	tbl_category_code	更新履歴の種別のコードを管理するテーブル

3.1.2 テーブル設計

テーブル名	tbl_item_group						
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	item_gruop_id	int unsigned		○	○		アイテムグループ番号
2	item_gruop_name	varchar	255				アイテムグループ名
3	comment	text					説明

テーブル名	tbl_item						
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	item_id	int unsigned		○	○		アイテム番号
2	item_name	varchar	255	○			アイテム名
3	item_URL	varchar	255	○			URL
4	follow_link	int					POSTで渡す引数
5	user_id	text		○			Basic認証のユーザID
6	password	text		○			Basic認証のパスワード
7	comment	text		○			説明
8	item_group_id	int					外部キー
9	medals_id	int		○			MEDALS番号
10	parameter	text					未使用

ドキュメント名	システム仕様書
システム名称	更新検知ツール

テーブル名		tbl sub item					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	sub_item_id	bigint unsigned		○	○		サブアイテム番号
2	sub_item_name	text					サブアイテム名
3	sub_item_URL	text					URL
4	last_modified	text					HTTPヘッダの最終更新日
5	content_length	text					HTTPヘッダのContent-Length
6	e_tag	text					HTTPヘッダのE-tag
7	item_id	int unsigned					外部キー

テーブル名		tbl update histry					
項番	カラム名		桁数	NOT NULL	主キー	インデックス	説明
1	id	int unsigned		○	○		連番
2	item_id	int unsigned					アイテム番号
3	date_update	int					アイテムが更新された日付(2009/10/01→20091001)
4	time_update	int					アイテムが更新された時間(22:10:01→221001)

テーブル名		tbl category					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	id	int unsigned		○	○		カテゴリー番号
2	histry_id	int unsigned					ヒストリーID
3	category_code	varchar	2				カテゴリーコード

テーブル名		tbl category code					
項番	カラム名	データ型	桁数	NOT NULL	主キー	インデックス	説明
1	code	varchar	2	○	○		カテゴリーコード
2	name	varchar	255				カテゴリー名