

ステータス:	却下	開始日:	2017-03-04
優先度:	通常	期日:	
担当者:	toshi_a 初音	進捗率:	0%
カテゴリ:			
対象バージョン:	3.6		
プラグイン名:			

説明

mikutterが扱うアカウント情報をホストしているSNSなどのサービスをmikutter側から取り扱うための仕組みをWorldと名付け、実装する。

Worldとは

Worldというクラスを実際に用意するわけではなく、Worldとして期待される特定の振る舞いを全て実装したModelのことをWorldと呼ぶ。したがってWorldのスーパークラスは全て `Diva::Model` となる。

目的

従来であればアカウントは全てTwitterだったため、特定のMessage

ModelがTwitterに属しているかどうか、ふぁぼれるか、リプライできるかといったことの判断基準となっていた。

マルチワールドが実現された暁には、特定のMessage

Modelが「Twitterか否か」ではなく「どういったサービスで提供されたものか」判断する必要がある。Worldが存在することで、アカウントや全てのModelは、それが所属するWorldを取り扱うことが出来るようになるため、例えばRSSのトピックに対してリツイートをしようとした時に、それが不可能であることがわかる。

アカウントを登録する時、最初にどのWorldクラスを利用するかをユーザに選ばせる。具体的にはこれが、TwitterやFacebookといった、アカウントが所属するサービスを選ぶことになる。

要求する振る舞い

一般的なプラグインがWorldのメソッドを直接扱うことは恐らく無く、二つのWorldインスタンスまたはクラスを比較することに意味があると思われる。

一方で、WorldはTwitterのCK/CSなど、大域的な認証情報を保持することができる。TwitterAccountは、TwitterWorldのインスタンスを親として持つことで、そのトークンがどのCK/CSの組み合わせで利用できるかを記録することができる。

例

Twitterの場合

Twitterの場合は、一つのTwitterクライアントがこれに当たる。実際のUIとして提供するかどうかは別として、TwitterWorldをCK/CS毎に作れば、それぞれにぶら下がるアカウントを別々に持てるため、アカウントごとに投稿元Twitterクライアントを変更することができる。また、別々のTwitterWorldに同じユーザでログインすることで、同じユーザで複数の投稿元Twitterクライアントを切り替えることが出来る。実用性はないが、BOTとして利用する時にmikutterを書き換えずに独自のクライアントキーに変更することができるようになる可能性がある。

現実的には、mikutterのTwitterWorldインスタンスをひとつだけ提供することになるし、異なるTwitterWorldであっても同じTwitterであるため、出来ることは変わらない。Twitterに関して言えばWorldのインスタンスを作成することが出来るという性質の恩恵はあまりないと思われる。

実用的な例: Slackの場合

Slackの場合、Worldはチームにあたる。新たなSlackチームに参加する場合は、対応するSlackWorldを作成し、アカウントをぶら下げることになる。Slackでは、チーム内にユーザが作成されるので、Twitterと違ってユーザは特定のSlackWorldのインスタンスに属する。

関連するチケット:

関連している 機能 #981: カスタムアカウント

終了

2017-01-30

履歴

#1 - 2017-04-07 20:27 - toshi_a 初音

- 関連している 機能 #981: カスタムアカウント を追加

#2 - 2017-12-20 06:55 - toshi_a 初音

- ステータスを 新規 から 却下 に変更

[#981](#) に対応する