

my_help の Thor によるインターフェース改善

情報科学科 西谷研究室 27014534 大八木利治

1 動機

my_help とは西谷研究室で用いられているユーザメモソフトである。このソースコードは ruby の標準ライブラリである optparse を用いて記述されており、サブコマンドはマイナスを付した省略記法が取られている [1]。しかし現在では、フルワードを用いた自然言語に近いサブコマンド体系が主流となってきた。そこで、my_help のソースコードを、自然言語に近いサブコマンド体系を実現しやすいライブラリである Thor によって書き換え、コマンドをより直感的に扱えるように変更する [2]。

2 my_help とは

my_help とは、ユーザー独自のマニュアルを作成することができるユーザメモソフトである。これは、terminal だけを用いて簡単に起動、編集、削除などを行うことができるため、非常に便利である。さらに、そのマニュアルは自分ですぐに編集、参照することができるので、メモとしての機能も果たしている。これにより、プログラミング初心者が、頻繁に使うコマンドやキーバインドなどをいちいち web browser を立ち上げて調べるのではなく、terminal 上で即座に取得できるため、プログラム開発を集中することが期待される。

3 optparse と thor の比較

書き換えを行う前に、非常に単純な CLI を作成し、optparse と Thor の比較を行った。今回作成した CLI には、キーボードから受け取った引数を name とし、Hello name と表示するコマンドを登録した。

optparse 版ではコマンドの一覧を自動的に作成できないが、Thor 版では自動でコマンド一覧を作成してくれる。また、description の中でコマンドの作成と定義を同時に行えるため、Thor 版の方がソースコードが短く簡潔になっている。表 1 は 2 つのライブラリの簡単な比較である。

表 1 optparse と Thor の比較

	コマンドの記法	特徴	ヘルプコマンド
optparse	マイナスを付した省略記法	ruby の標準ライブラリであり古くから使われている	手動生成
Thor	任意の文字列	サブコマンドを含むコマンドラインツールの作成が容易	自動生成

4 Thor での書き換え

4.1 手順

optparse から Thor への書き換えの主な手順は以下の通りである。

- .gemspec ファイル内に Thor がインストールされるように変更を加える。
- exe ディレクトリの中にあるコマンドファイルを書き換える。
- ソースファイル内の self.run を消去する。
- ソースファイル内の initialize メソッドを書き換え、Thor の initialize メソッドを継承するように変更する。
- execute 関数を消去する
- 関数宣言を書き換え、自然言語に近いオプションを設定する。
- コマンド設定しない関数をまとめて、コマンドでないことを宣言する。

4.2 コマンドの自動生成について

my_help で作成した自作のヘルプコマンドはデフォルトで存在するオプションと個別に設定されるオプションの 2 種類を持つ。個別に設定するオプションについては optparse を用いて自動設定されている。これを Thor で書き換えようとすると冗長で複雑なソースコードになってしまう。そのため、その機能は optparse で実装し、それ以外の機能を Thor で書き換えた。これにより、自作のヘルプコマンドについては optparse と Thor を連携させて実行できるようにした。

5 考察

今回、オプションの記法を省略記法からフルワードを用いた記法に変更したことで、コマンドを今までよりも直感的に使用できるようになった。そのため、初心者でもコマンドの意味が理解しやすくなり、学習コストの削減が期待される。また、ソースコードを簡略化でき、可読性が上がったことにより、機能拡張が容易になり今後の my_help の継続的な開発が見込まれる。

参考文献

- erikhuda, library optparse, <https://docs.ruby-lang.org/ja/latest/library/optparse.html>, accessed 2018.2.9.
- S.Koichiro, Thor の使い方まとめ, <http://qiita.com/succi0303/items/32560103190436c9435b>, accessed 2018.2.9.