

Localization

It is highly recommended that you use this feature as it also allows server admins to customize your strings.

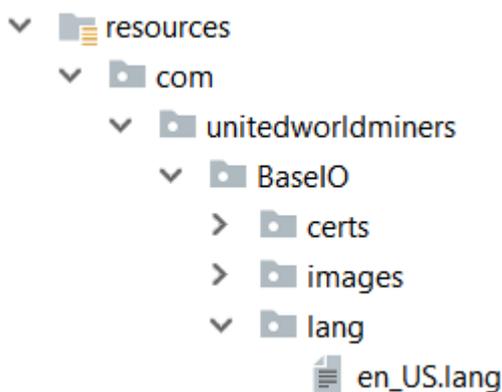
Localization is a feature of MessageUtils.

Setup

To start working with BaseIO's localization feature just call

`MessageUtils.setupTranslations(langResourcePath)`. For the most basic setup use `null` as parameter and you're done. Now server admins can add translations for your strings as they wish.

If you want to add your own translations to your mod fill `langResourcePath` with a string path pointing to a folder in your mod resources.



For example BaseIO has `MessageUtils.setupTranslations("com/unitedworldminers/BaseIO/lang")`. It is recommended to call this method in your init method (`FMLInitializationEvent`).

Creating localization mappings

The default language file is `en_US.lang`. This file will be used if you aren't providing the correct language.

You can use almost any string as a key, but it's recommended to abstract them.

The mapping files itself are the same as the ones from Minecraft: `<key>=<mapping>`, an example file can be found on the right sidebar. This files have to be named by the [language tag](#) given by Minecraft and must have the extension `.lang`.

In the mappings you can use arguments following the same format of the java formatter (`String.format`). These arguments can be injected in the method calls using localizations.

If you are a modder you put the files in your resource folder that you pointed at in `setupTranslations`. If you are a server admin you can put the files into `config/lang/<modid>/`.

You can reload the mappings with `/baseio reloadMessages <modid>`.

Using localizations

Your localizations can be used in

- `MessageUtils.messageToSender()`
- Simplecommand's `commandUsage()`
- Manually: `MessageUtils.getTranslation()`

Revision #9

Created 31 May 2017 11:01:01 by deregges

Updated 19 October 2017 06:56:31 by deregges