



ISO Developer's Toolkit - mdTranslator in Ruby

2015 CDI Workshop

May 11, 2015

Stan Smith, USGS

mdTranslator in Ruby 1



1. Complete the installation of Ruby and the adiwg-mdtranslator RubyGem if this is not already done.
2. Using your favorite code editor create a new file named 'translate.rb'.
3. Start by including the RubyGem 'adiwg-mdtranslator'.

```
# simple mdTranslator Ruby application  
  
# include the adiwg-mdtranslator RubyGem in the application  
require 'adiwg-mdtranslator'
```



mdTranslator in Ruby 2



4. Read in the file with your metadata content in mdJson format.

```
# read file in the mdJson file
my_file = File.open('C:\Users\...\full_example.json', 'r')
jsonObj = my_file.read
my_file.close
```

5. Call the mdTranslator.

```
# call mdtranslator with desired parameters
# mdtranslator uses a 'named' parameter list
metadata = ADIWG::Mdtranslator.translate(
  file: jsonObj, reader: 'mdJson', validate: 'normal',
  writer: 'iso19115_2', showAllTags: true)
```

mdTranslator in Ruby 3



6. Extract the output from the return.

```
# extract the metadata output from the returned metadata hash
if !metadata[:writerOutput].nil?
  writerOut = metadata[:writerOutput]
  metadata[:writerOutput] = 'Extracted'
end
```

7. Save the formatted metadata record.

```
# send the metadata output to a file
File.open('C:\Users\...\mdOutput.xml', 'w') { |file|
  file.write(writerOut) }
```

8. All done!



mdTranslator in Ruby 4

9. Other things to do ...

- Examine other elements of the 'return hash'

```
# show metadata returned hash  
require 'pp'  
pp metadata
```

- Check return values for errors
- Handle error reporting
- Handle different writers
- Support mdTranslator.translate options
- Blah, blah, blah ...

Questions?

