national center for ontological research
ncor

University at Buffalo
The State University of New York

Ontologies in an Automated Workflow Part 1

John Beverley

Assistant Professor, *University at Buffalo*
Co-Director, National Center for Ontological Research
Affiliate Faculty, *Institute of Artificial Intelligence and Data Science*

Outline

- Warmup
- The Common Core Ontologies Repository
- SPARQL

Outline

- Warmup
- The Common Core Ontologies Repository
- SPARQL

A Murder Mystery

A Murder Mystery

A Psychologist has been Murdered…

1. Suspects: Werner, Mark, Neil, and Barry
2. Mark plays violin
3. Neil, Mark, and Barry daydream
4. Barry cannot play music, but reads and solves problems
5. Werner never daydreams
6. Mark doesn't problem solve
7. Werner plays trumpet and problem solves
8. Neil cannot play music and cannot read
9. The murderer daydreams, is either a musician or literate, and solves problems

A Psychologist has been Murdered…

1. Suspects: Werner, Mark, Neil, and Barry
2. Mark plays violin
3. Neil, Mark, and Barry daydream
4. Barry cannot play music, but reads and solves problems
5. Werner never daydreams
6. Mark doesn't problem solve
7. Werner plays trumpet and problem solves
8. Neil cannot play music and cannot read
9. The murderer daydreams, is either a musician or literate, and solves problems

Represent these facts in Protégé.

A Psychologist has been Murdered…

1. Suspects: Werner, Mark, Neil, and Barry
2. Mark plays violin
3. Neil, Mark, and Barry daydream
4. Barry cannot play music, but reads and solves problems
5. Werner never daydreams
6. Mark doesn't problem solve
7. Werner plays trumpet and problem solves
8. Neil cannot play music and cannot read
9. The murderer daydreams, is either a musician or literate, and solves problems

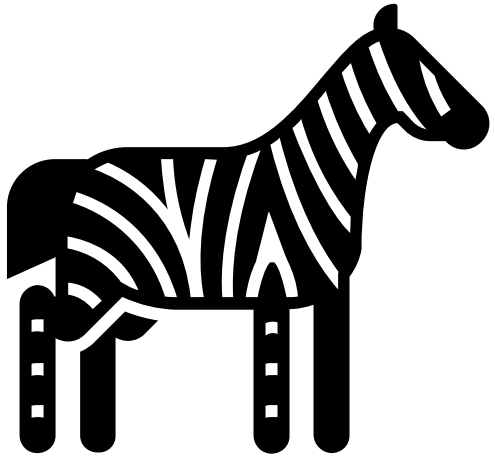
Represent these facts in Protégé.

Create a 'Murderer' class that is equivalent to line 9.

If done correctly, the Protege reasoner will return exactly one member of this class.

Zebra Puzzle

1. There are five houses.
2. The Englishman lives in the red house.
3. The Spaniard owns the dog.
4. Coffee is drunk in the green house.
5. The Ukrainian drinks tea.
6. The green house is immediately to the right of the ivory house.
7. The Old Gold smoker owns snails.
8. Kools are smoked in the yellow house.
9. Milk is drunk in the middle house.
10. The Norwegian lives in the first house.
11. The man who smokes Chesterfields lives in the house next to the man with the fox.
12. Kools are smoked in a house next to the house where the horse is kept.
13. The Lucky Strike smoker drinks orange juice.
14. The Japanese man smokes Parliaments.
15. The Norwegian lives next to the blue house.



WHO OWNS THE ZEBRA?

Outline

- Warmup
- The Common Core Ontologies Repository
- SPARQL

The Common Core Ontologies (CCO)

What is CCO?

The Common Core Ontologies (CCO) is a widely-used suite of eleven ontologies that consist of logically well-defined generic terms and relations among them reflecting entities across all domains of interest.

These eleven ontologies constitute a mid-level ontology that extends from the Basic Formal Ontology (BFO), an ISO-standard top-level ontology. Whereas BFO represents only the most generic entities and relations, CCO contains classes that users will find common across data sets in many domains. Such classes include, for example, person, facility, date, employment, nickname, and measurement.

Both BFO and CCO have been directed for use as "baseline standards" for formal ontology development across the United States Department of Defense and Intelligence Community.

CCO is currently being evaluated as a mid-level ontology standard by the IEEE Standards Association under PAR3195.1.

CCO itself is not intended to grow indefinitely by including content that is proper to particular domains. Users are encouraged to create their own domain extensions with content particular to those domains and publish these ontologies for re-use by others.

https://github.com/CommonCoreOntology/CommonCoreOntologies

The Common Core Ontologies

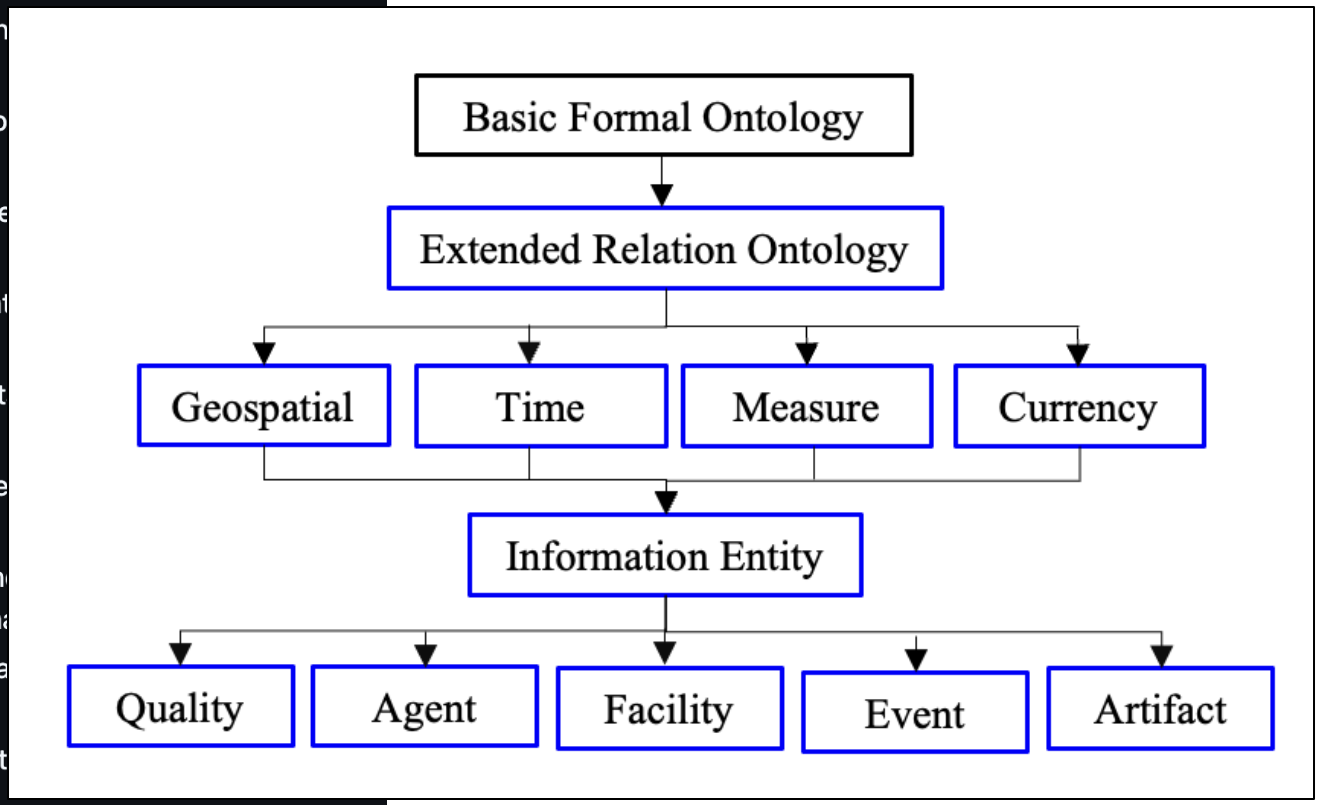
- **Geospatial Ontology** - An ontology whosse scope is the representation of sites, spatial regions, and other entities, especially those that are located near the surface of Earth, as well as the relations that hold between them.
- **Information Entity Ontology** - An ontology whose scope is the representation of generic types of information as well as the relationships between information and other entities.
- **Event Ontology** - An ontology whose scope is the representation of processual entities, especially those performed by agents, that occur within multiple domains.
- **Time Ontology** - An ontology whose scope is the representation of temporal regions and the relations that hold between them.
- **Agent Ontology** - An ontology whose scope is the representation of represent agents, especially persons and organizations, and their roles.
- **Quality Ontology** - An ontology whose scope is the representation of a range of attributes of entities especially qualities, realizable entities, and process profiles.
- **Units of Measure Ontology** - An ontology whose scope is the representation of standard measurement units that are used when measuring various attributes of entities.
- **Currency Unit Ontology** - An ontology whose scope is the representation of currencies that are issued and used by countries.
- **Facility Ontology** - An ontology whose scope is the representation of buildings and campuses that are designed to serve some specific purpose, and which are common to multiple domains.
- **Artifact Ontology** - An ontology whose scope is the representation of artifacts that are common to multiple domains along with their models, specifications, and functions.
- **Extended Relation Ontology** - An ontology whose scope is the representation of the relations that hold between entities at the level of the mid-level Common Core Ontologies.

https://github.com/CommonCoreOntology/CommonCoreOntologies

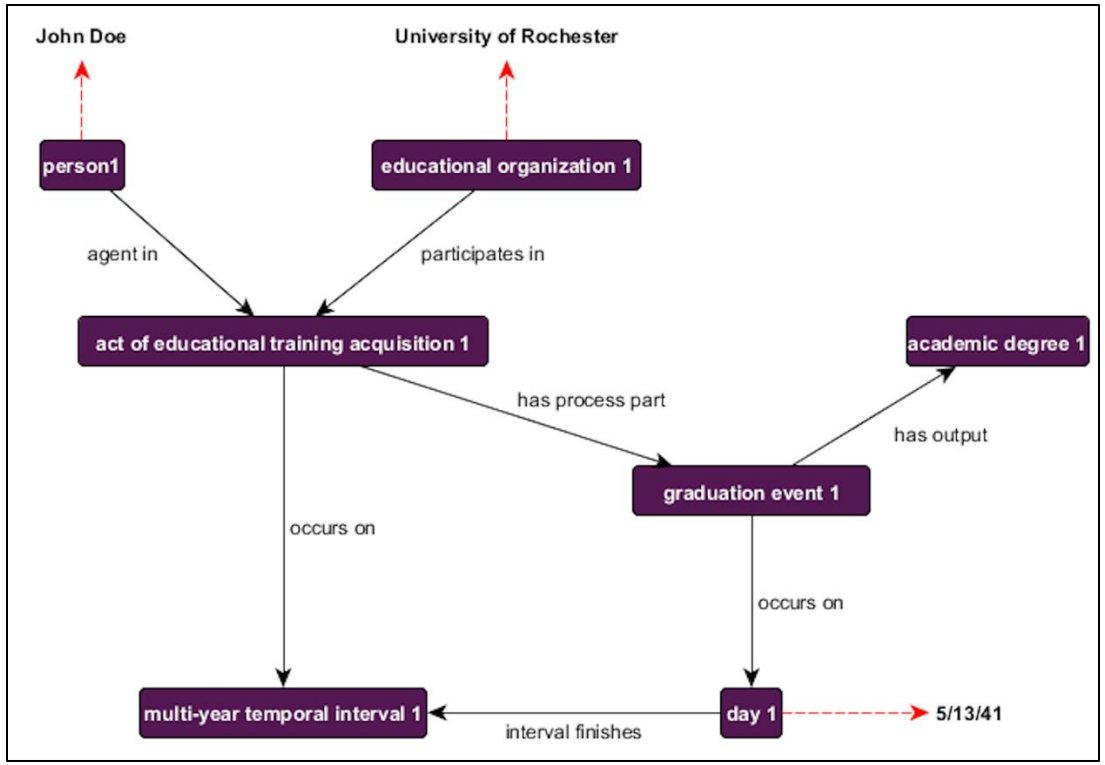
The Common Core Ontologies

- **Geospatial Ontology** - An ontology whosse scope is the representation of sites, spatial regions, and other entities, especially those that are located near the surface of Earth, as well as the relations that hold between them.
- **Information Entity Ontology** - An ontology whose scope is the representation of generic types of information as well as the relationships between information and other entities.
- **Event Ontology** - An ontology whose scope is the representation of processual en performed by agents, that occur within multiple domains.
- **Time Ontology** - An ontology whose scope is the representation of temporal regio hold between them.
- **Agent Ontology** - An ontology whose scope is the representation of represent age organizations, and their roles.
- **Quality Ontology** - An ontology whose scope is the representation of a range of at especially qualities, realizable entities, and process profiles.
- **Units of Measure Ontology** - An ontology whose scope is the representation of st that are used when measuring various attributes of entities.
- **Currency Unit Ontology** - An ontology whose scope is the representation of curre used by countries.
- **Facility Ontology** - An ontology whose scope is the representation of buildings an designed to serve some specific purpose, and which are common to multiple dom
- **Artifact Ontology** - An ontology whose scope is the representation of artifacts tha domains along with their models, specifications, and functions.
- **Extended Relation Ontology** - An ontology whose scope is the representation of t between entities at the level of the mid-level Common Core Ontologies.

Contributors 11

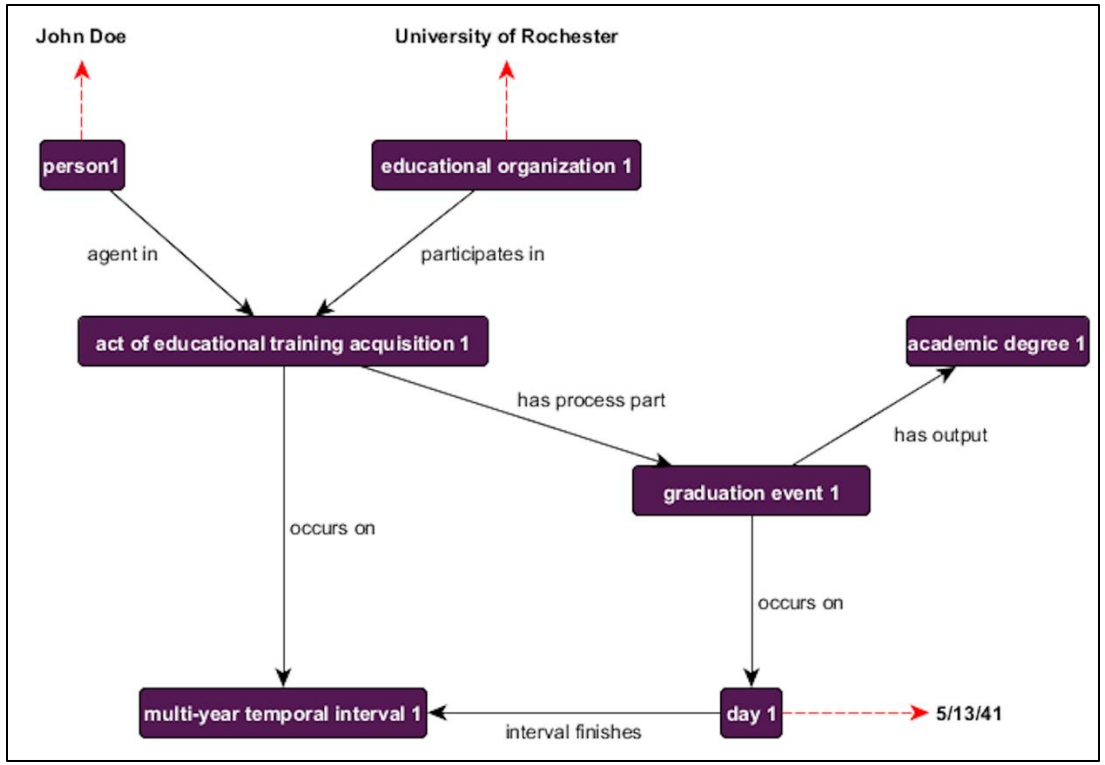


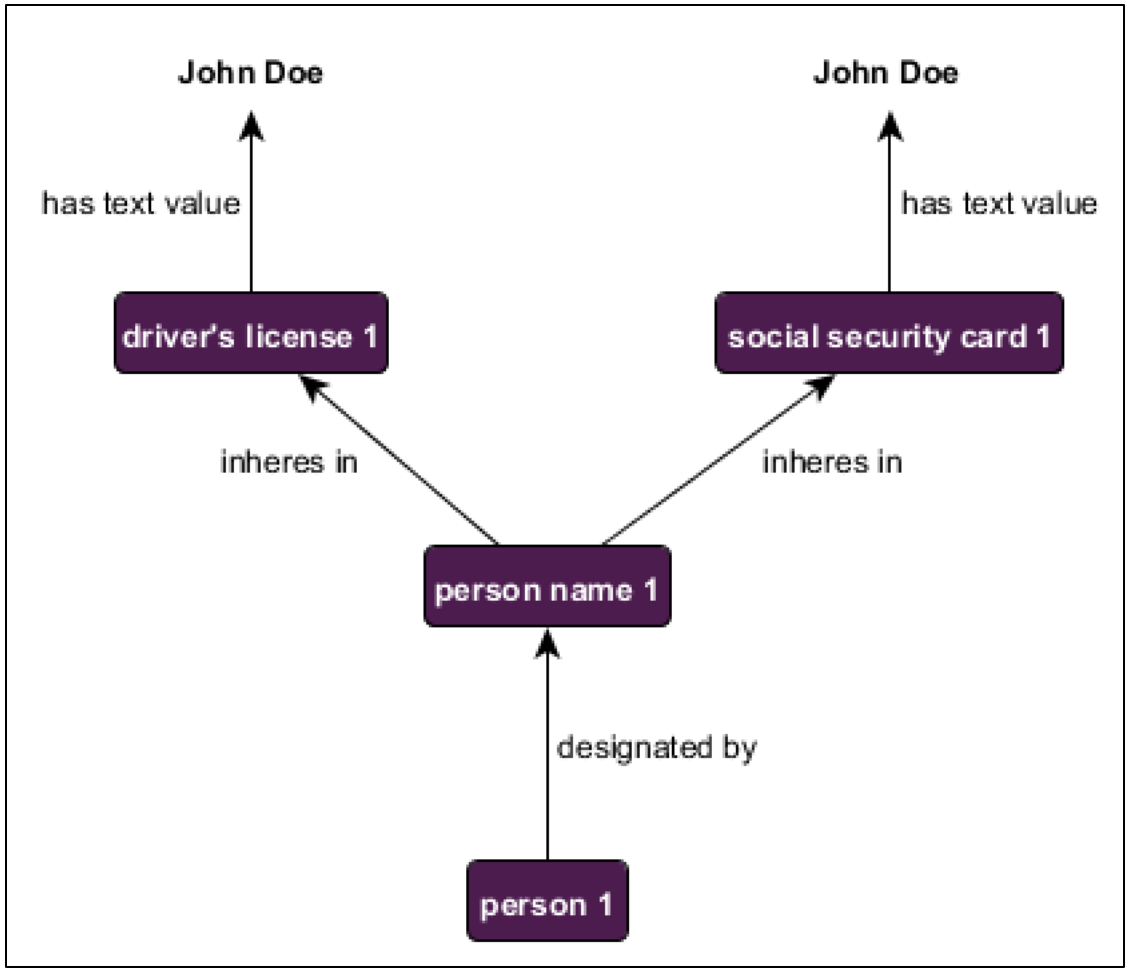
https://github.com/CommonCoreOntology/CommonCoreOntologies



High-level design pattern representing a graduation and the acquisition of an academic degree

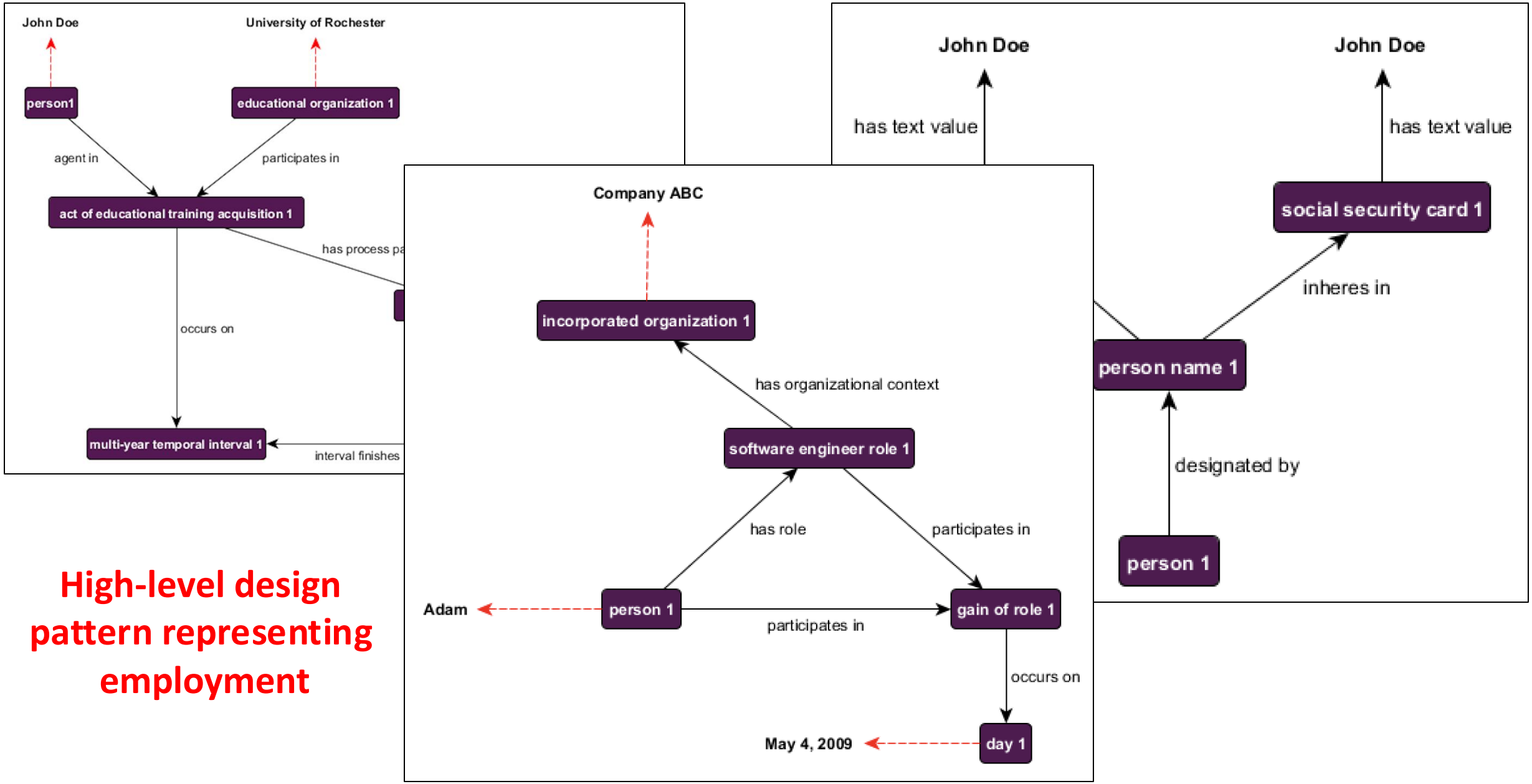
https://github.com/CommonCoreOntology/CommonCoreOntologies





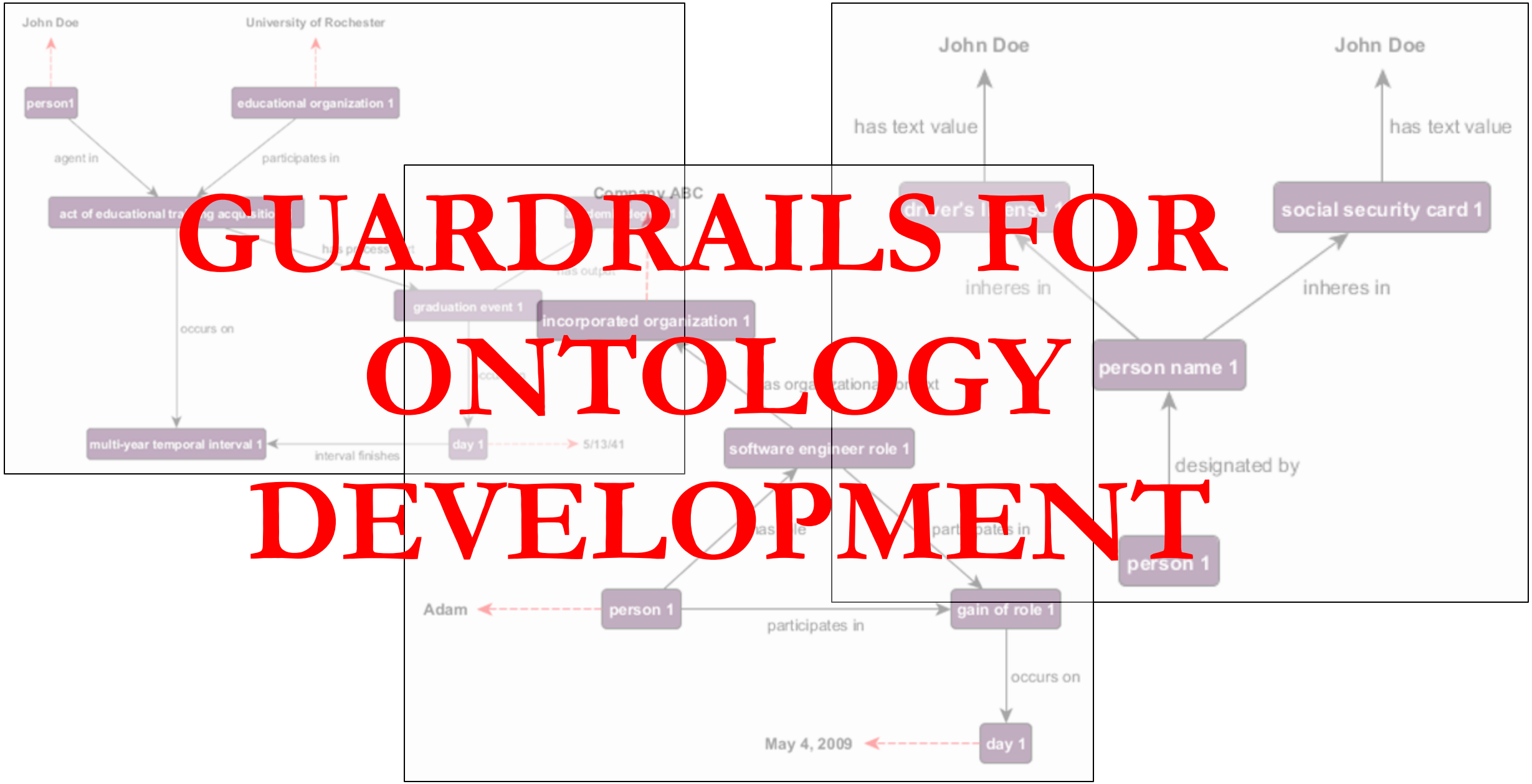
High-level design pattern representing credentials

https://github.com/CommonCoreOntology/CommonCoreOntologies



High-level design pattern representing employment

https://github.com/CommonCoreOntology/CommonCoreOntologies



GUARDRAILS FOR ONTOLOGY DEVELOPMENT

https://github.com/CommonCoreOntology/CommonCoreOntologies

CommonCoreOntologies

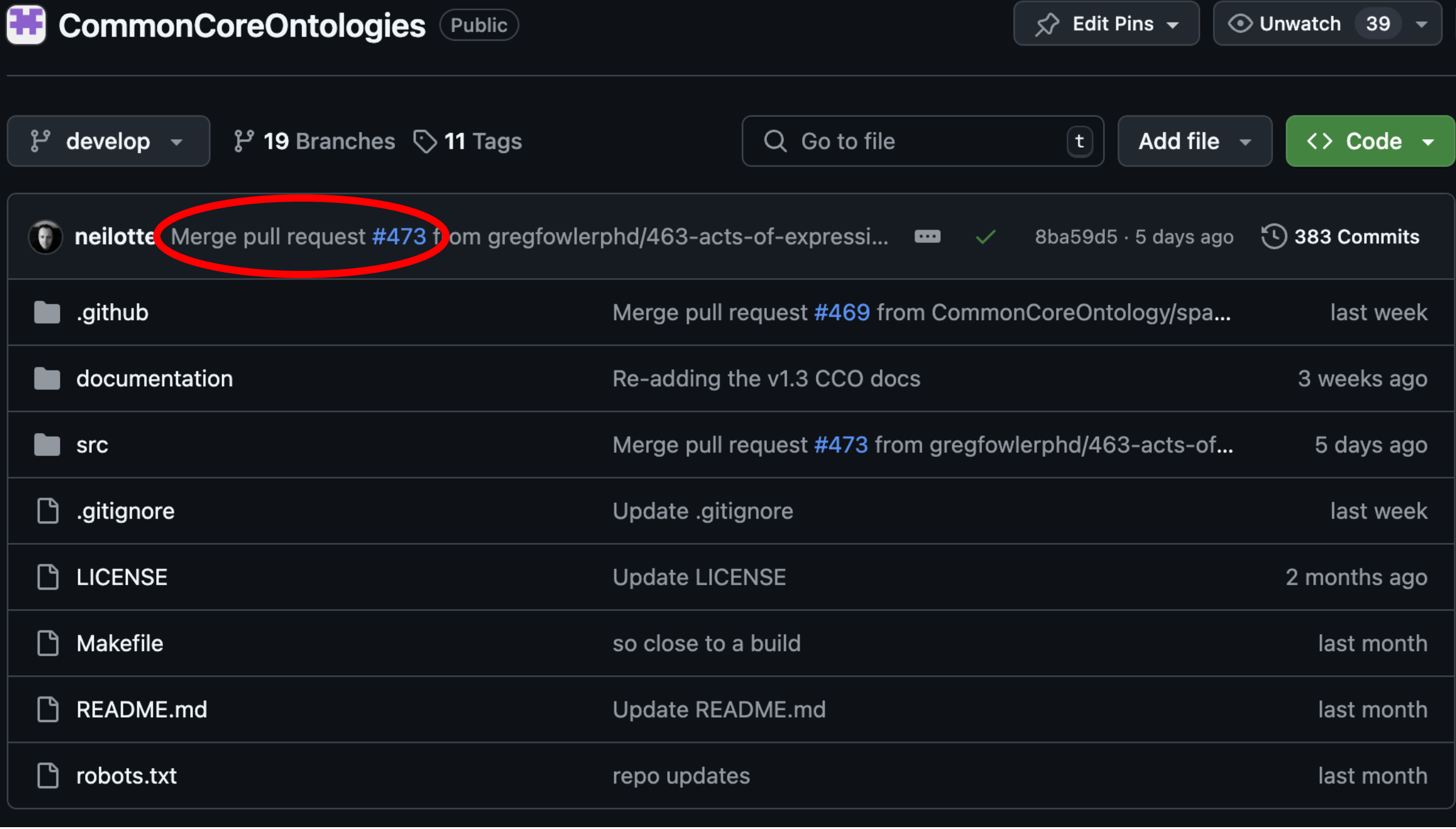
Public

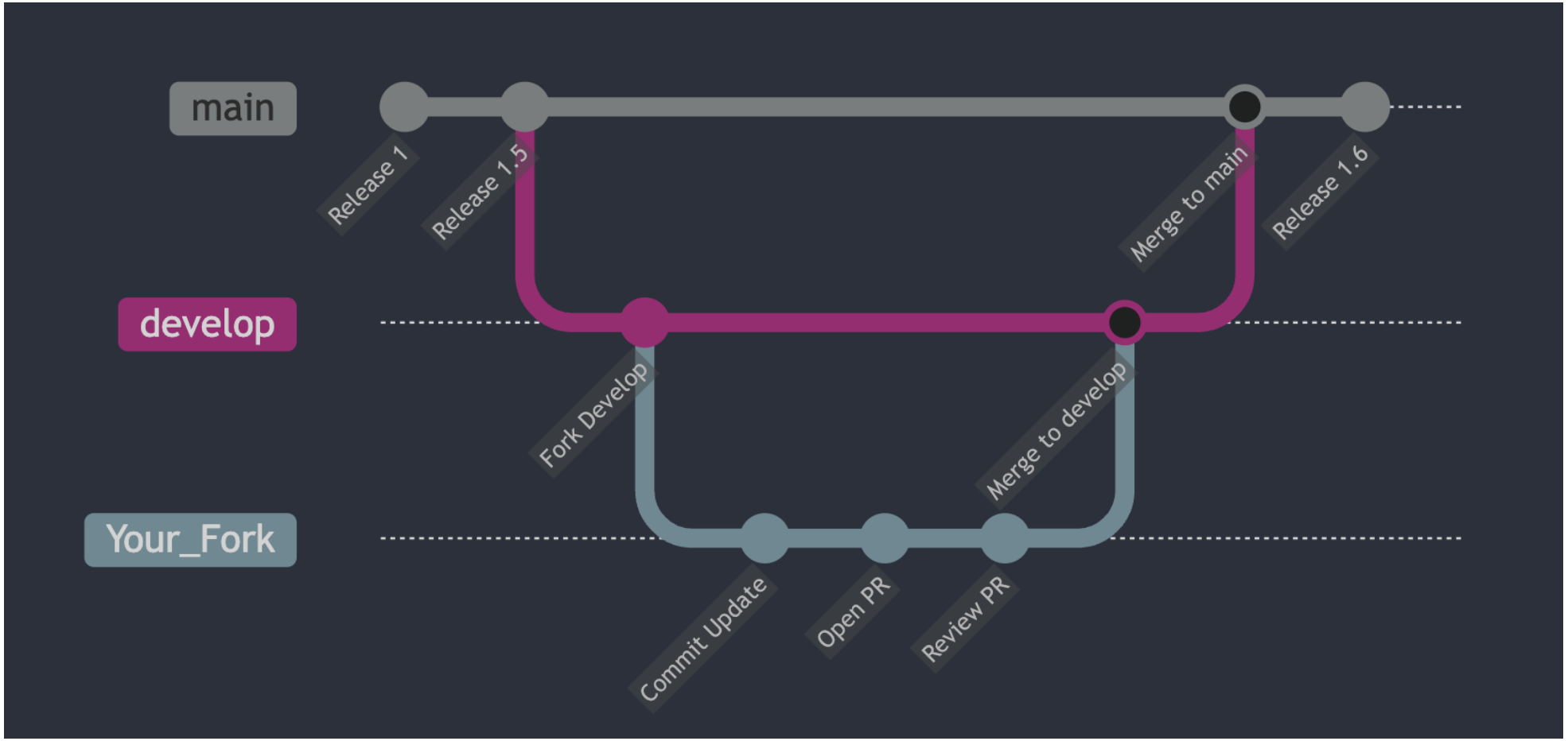
Edit Pins | Unwatch 39

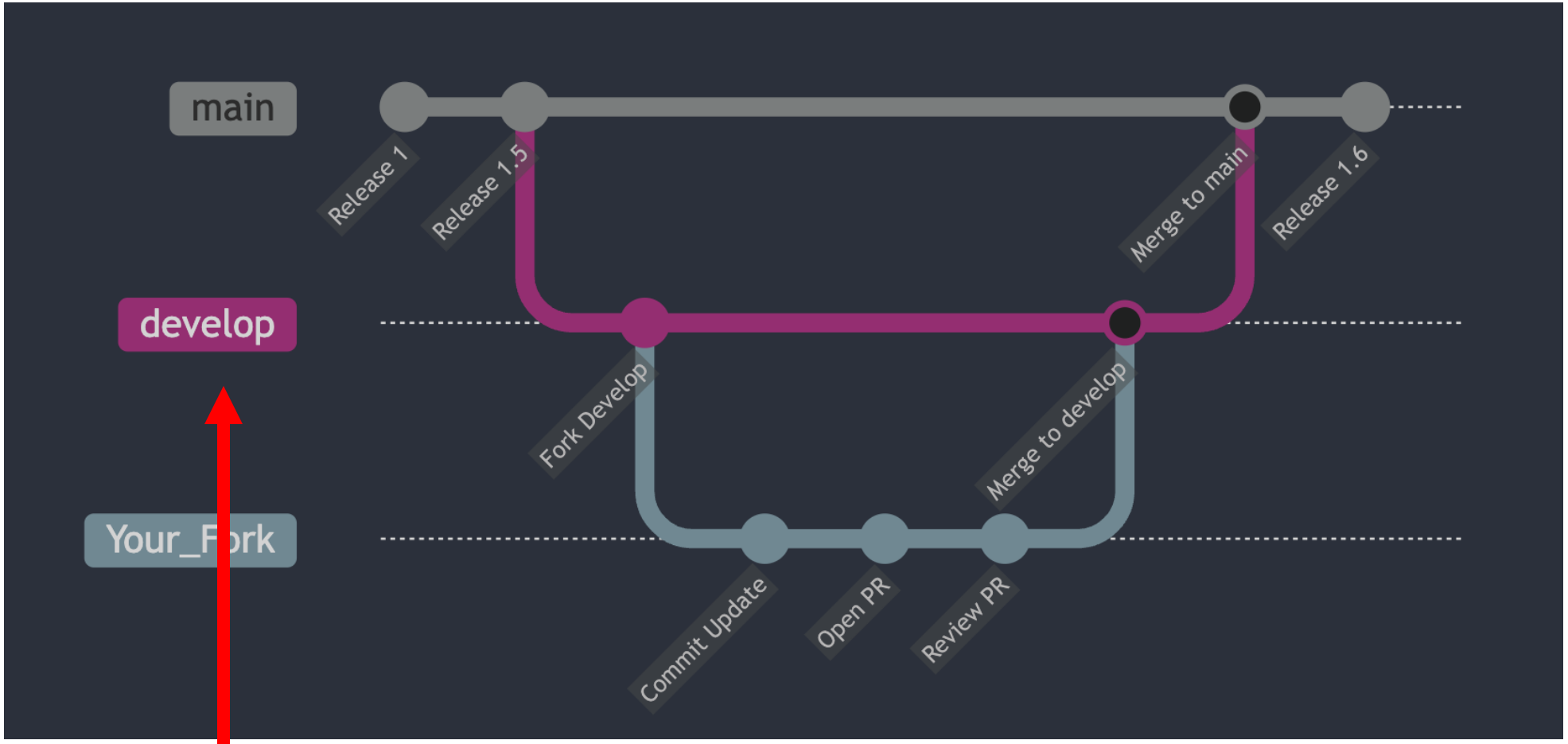
develop | 19 Branches | 11 Tags | Go to file t | Add file | Code

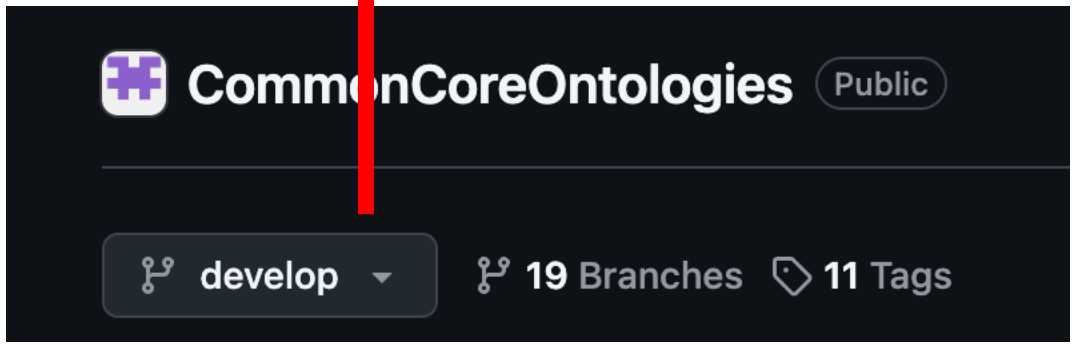
neilotte Merge pull request #473 from gregfowlerphd/463-acts-of-expressi... 8ba59d5 · 5 days ago 383 Commits

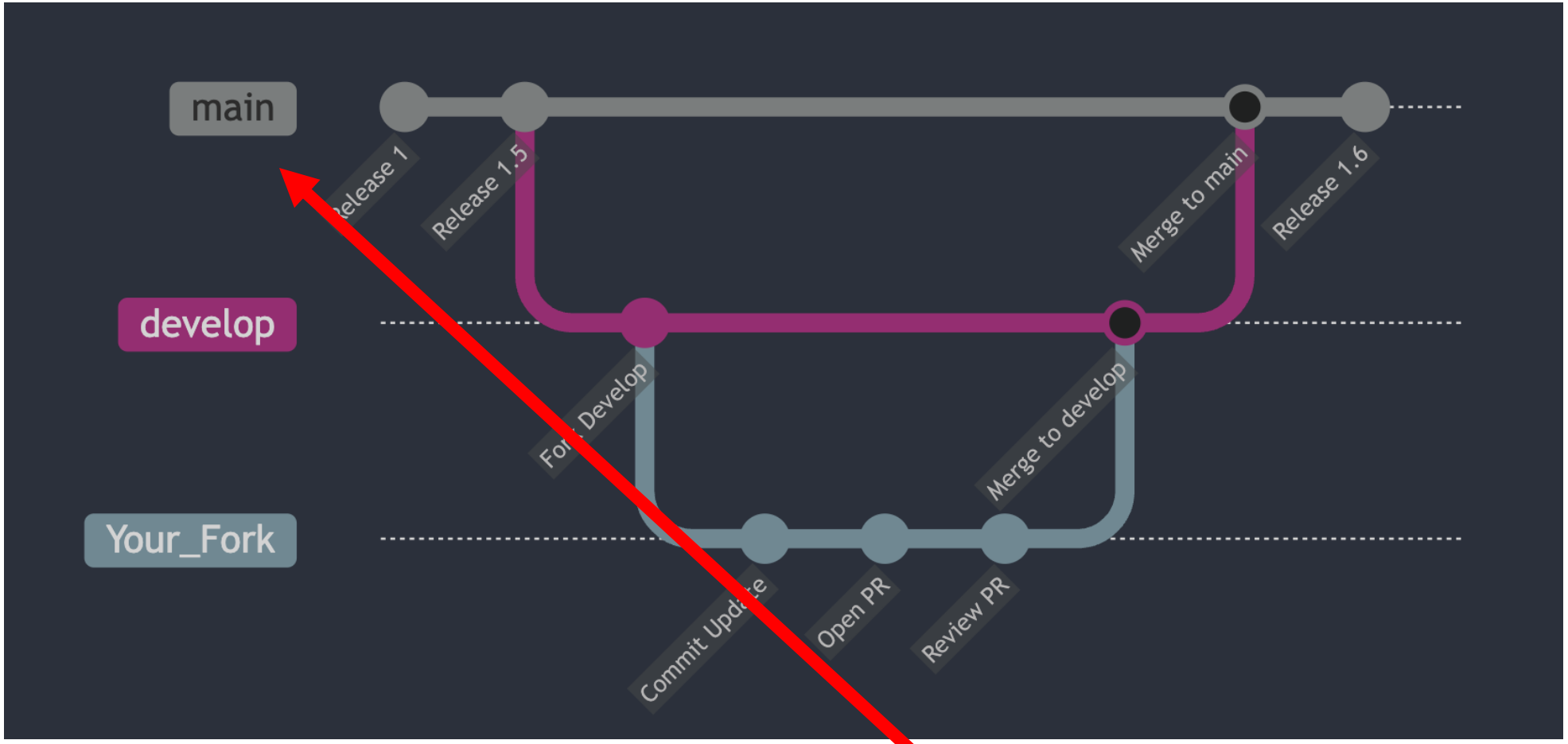
.github	Merge pull request #469 from CommonCoreOntology/spa...	last week
documentation	Re-adding the v1.3 CCO docs	3 weeks ago
src	Merge pull request #473 from gregfowlerphd/463-acts-of...	5 days ago
.gitignore	Update .gitignore	last week
LICENSE	Update LICENSE	2 months ago
Makefile	so close to a build	last month
README.md	Update README.md	last month
robots.txt	repo updates	last month


CommonCoreOntologies
Public
Edit Pins
Unwatch 39
develop
19 Branches
11 Tags
Go to file
t
Add file
Code
neilotte
Merge pull request #473 from gregfowlerphd/463-acts-of-expressi...
8ba59d5 · 5 days ago
383 Commits
.github
Merge pull request #469 from CommonCoreOntology/spa...
last week
documentation
Re-adding the v1.3 CCO docs
3 weeks ago
src
Merge pull request #473 from gregfowlerphd/463-acts-of...
5 days ago
.gitignore
Update .gitignore
last week
LICENSE
Update LICENSE
2 months ago
Makefile
so close to a build
last month
README.md
Update README.md
last month
robots.txt
repo updates
last month


main
develop
Your_Fork
Release 1
Release 1.5
Merge to main
Release 1.6
Fork Develop
Merge to develop
Commit Update
Open PR
Review PR


main
develop
Your_Fork
Release 1
Release 1.5
Merge to main
Release 1.6
Fork Develop
Merge to develop
Commit Update
Open PR
Review PR


CommonCoreOntologies
Public
develop
19 Branches
11 Tags



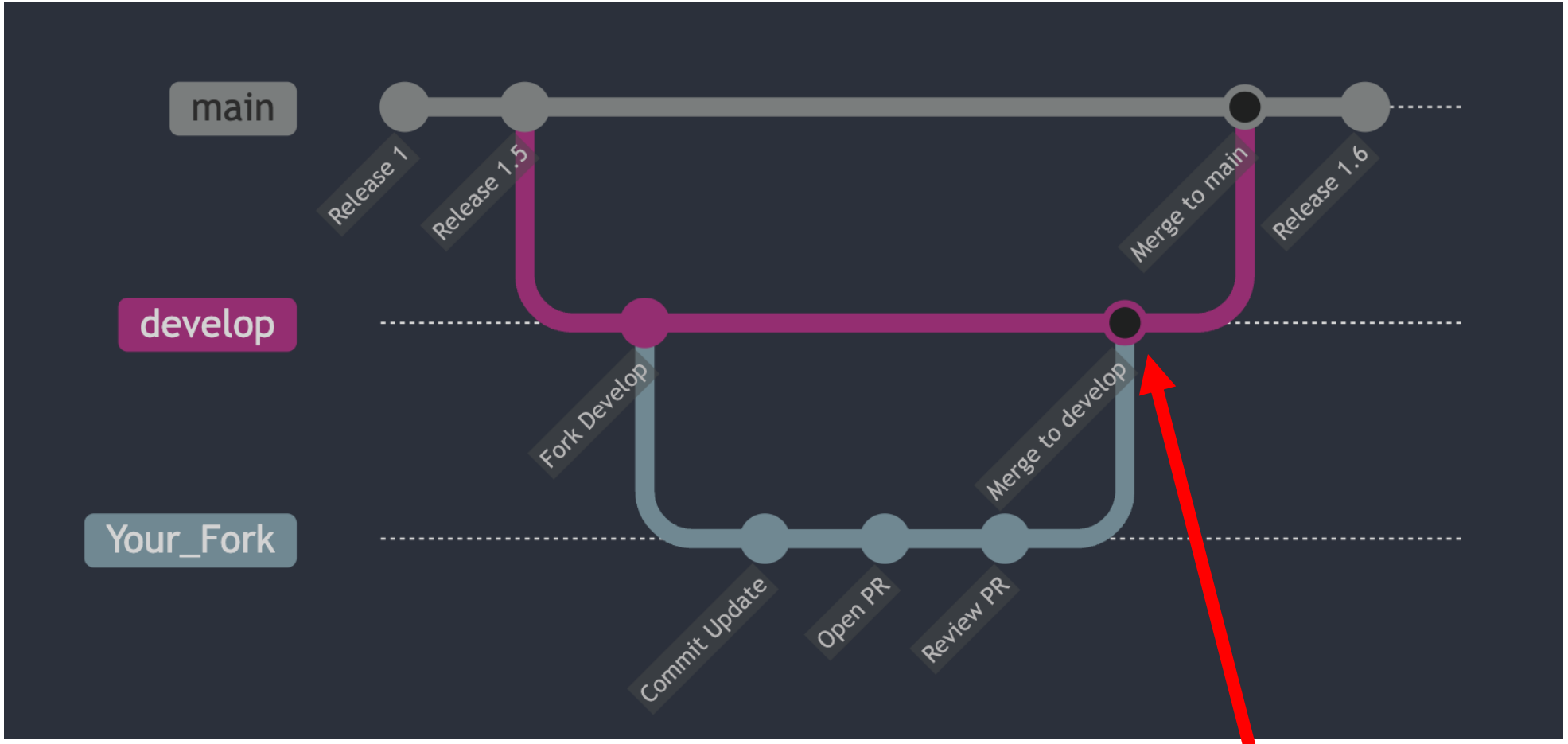
CommonCoreOntologies Public

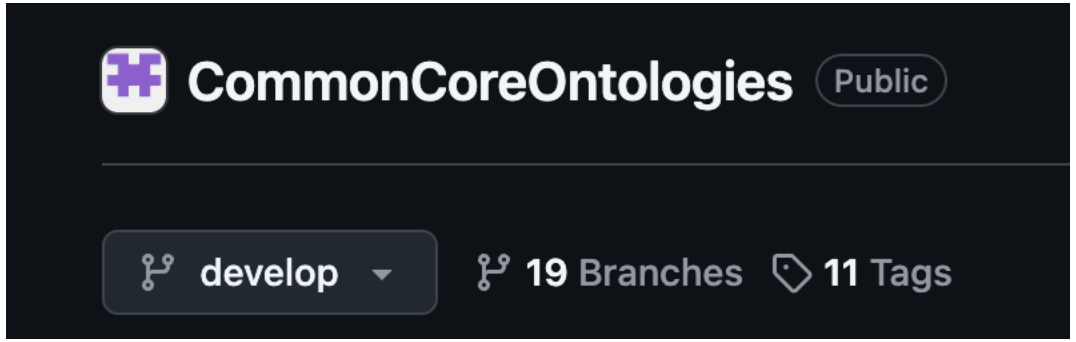
develop 19 Branches 11 Tags

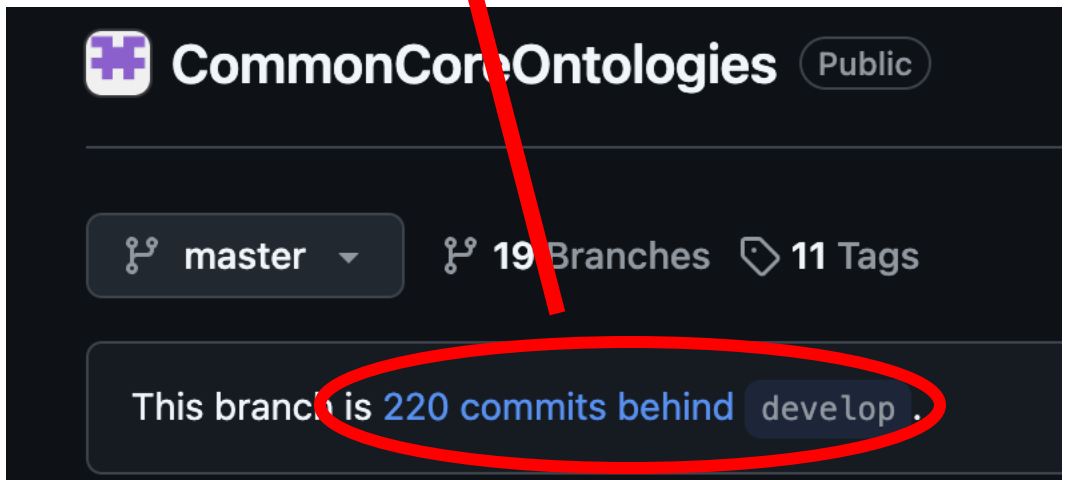
CommonCoreOntologies Public

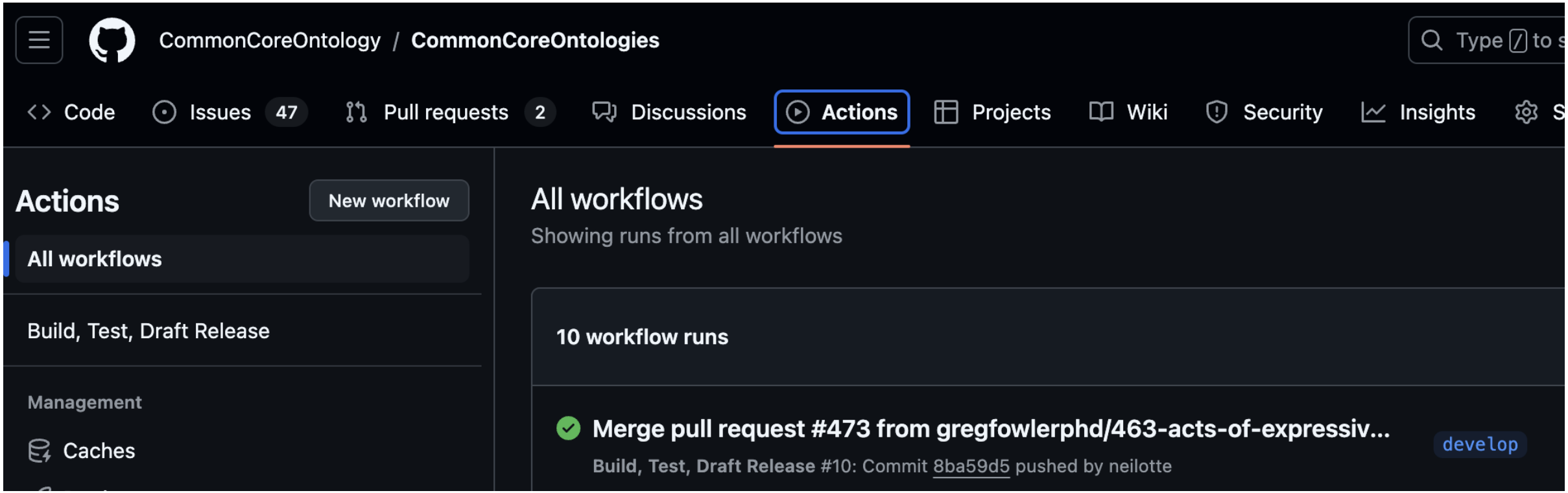
master 19 Branches 11 Tags

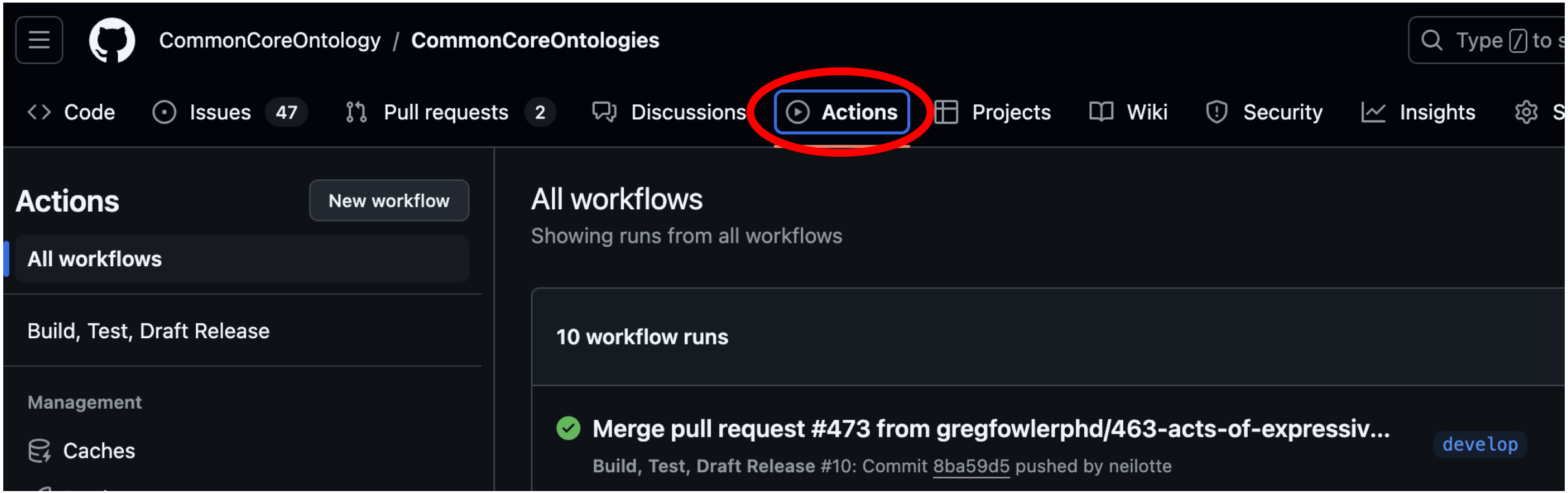
This branch is 220 commits behind develop.

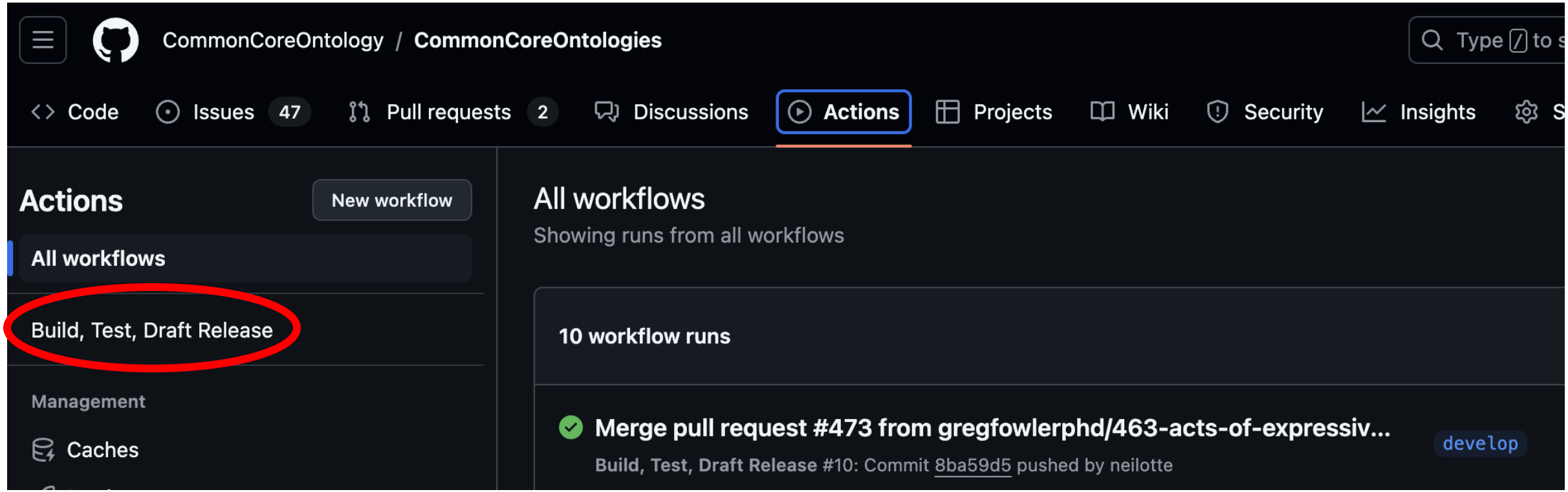

main
develop
Your_Fork
Release 1
Release 1.5
Fork Develop
Commit Update
Open PR
Review PR
Merge to develop
Merge to main
Release 1.6

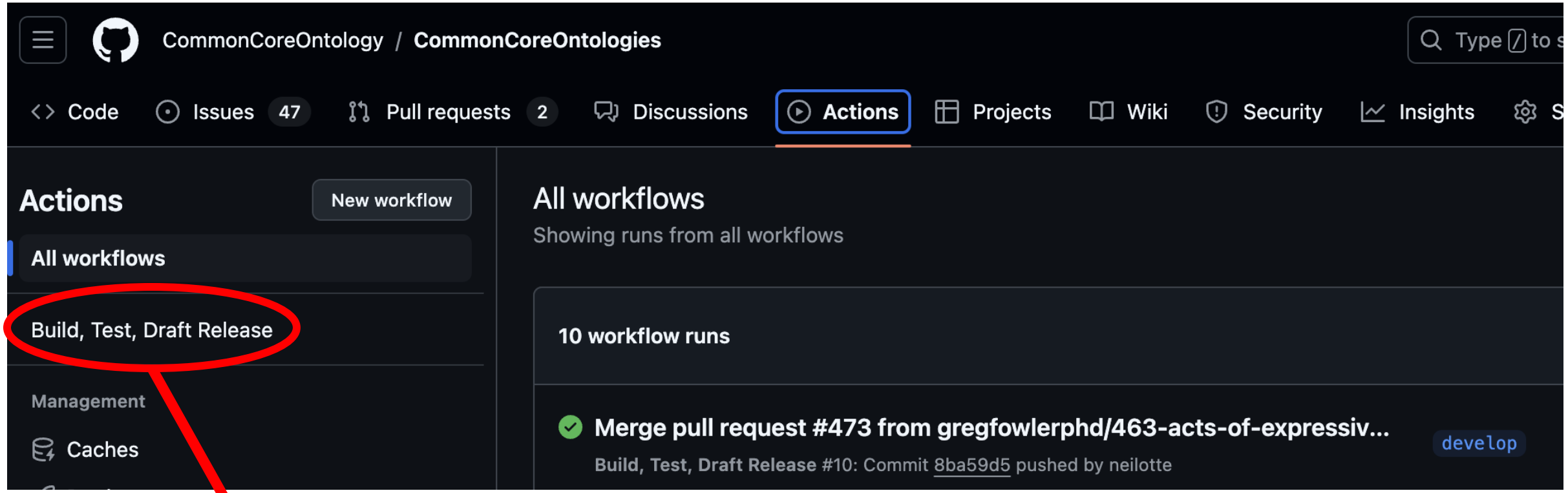

CommonCoreOntologies
Public
develop
19 Branches
11 Tags

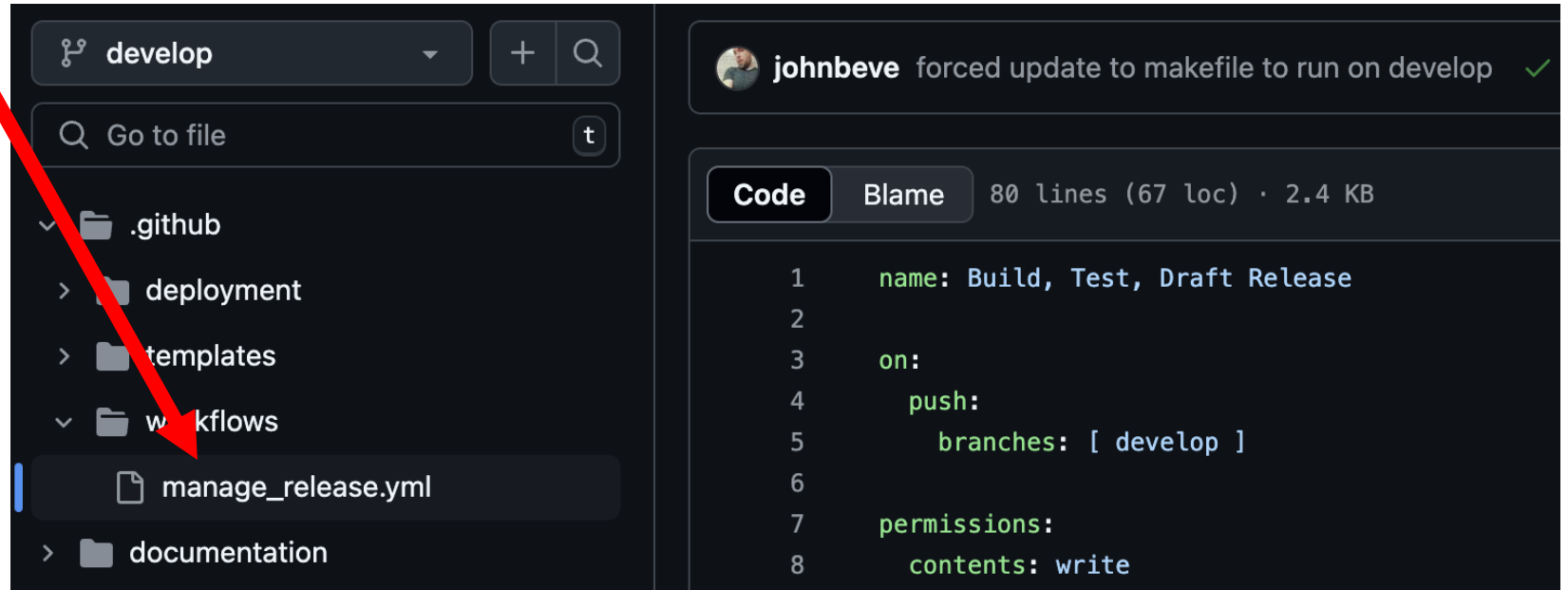

CommonCoreOntologies
Public
master
19 Branches
11 Tags
This branch is 220 commits behind develop .

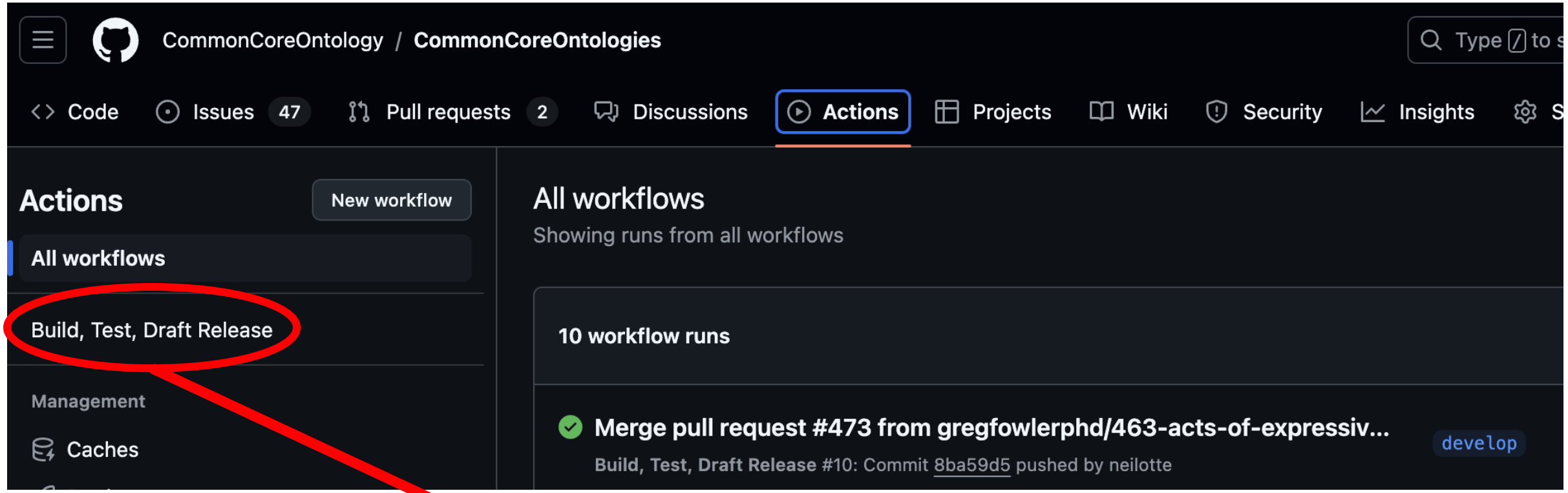

CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
S
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte
develop

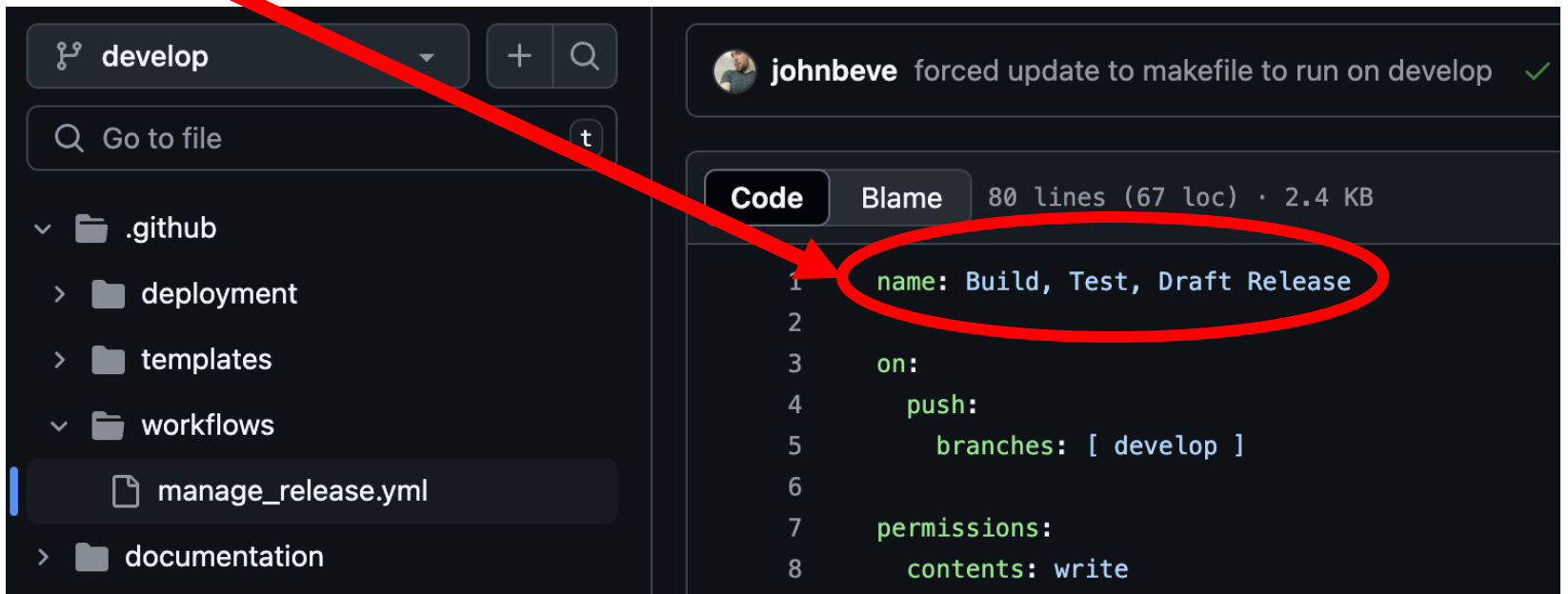

CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte
develop

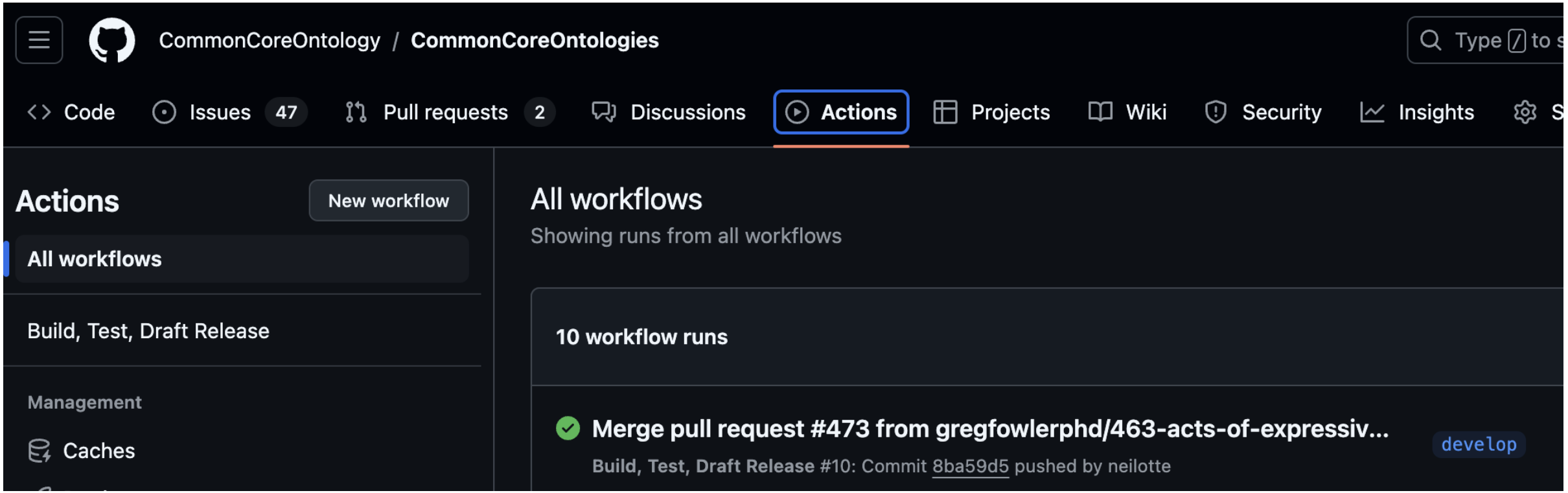

CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte
develop

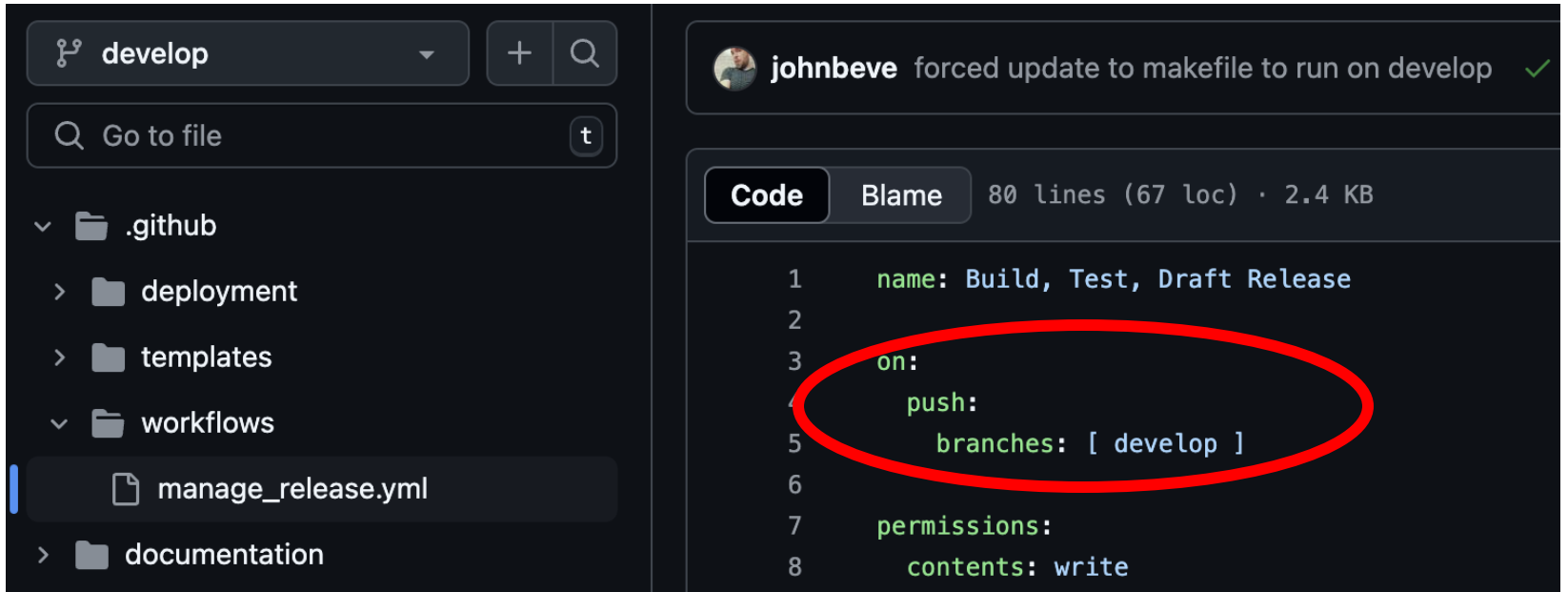

CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
develop
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte

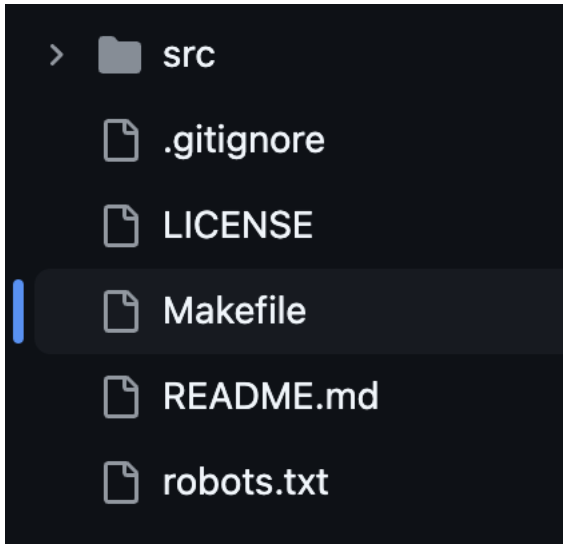

develop
Go to file
t
.github
deployment
templates
workflows
manage_release.yml
documentation
johnbeve forced update to makefile to run on develop
Code
Blame
80 lines (67 loc) · 2.4 KB
1 name: Build, Test, Draft Release
2
3 on:
4 push:
5 branches: [develop]
6
7 permissions:
8 contents: write

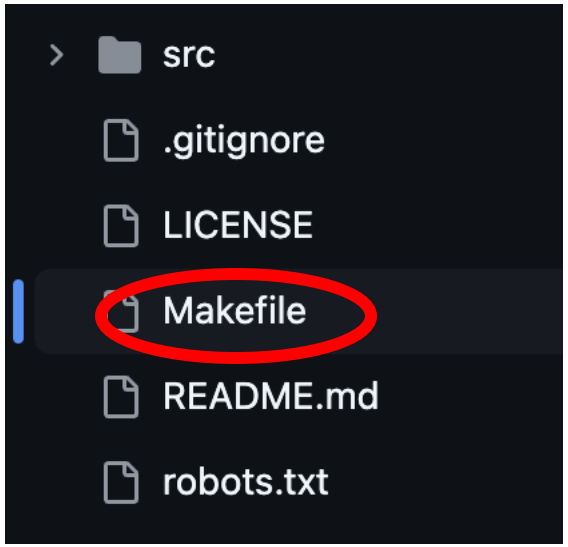

CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
develop
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte

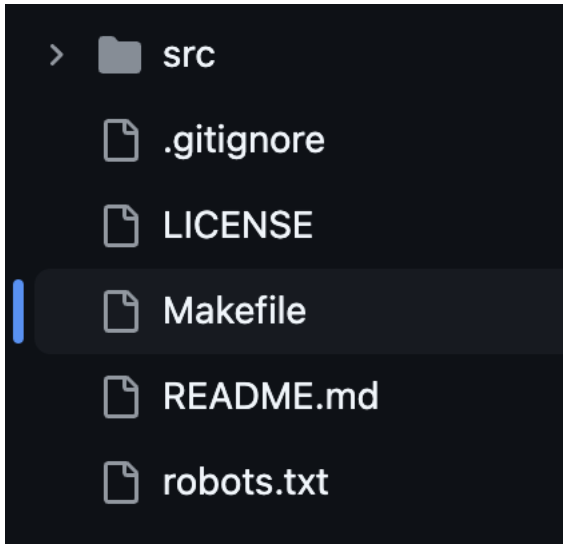

develop
Go to file
t
.github
deployment
templates
workflows
manage_release.yml
documentation
johnbeve forced update to makefile to run on develop
Code
Blame
80 lines (67 loc) · 2.4 KB
name: Build, Test, Draft Release
on:
push:
branches: [develop]
permissions:
contents: write


CommonCoreOntology / CommonCoreOntologies
Type / to s
Code
Issues 47
Pull requests 2
Discussions
Actions
Projects
Wiki
Security
Insights
S
Actions
New workflow
All workflows
Build, Test, Draft Release
Management
Caches
All workflows
Showing runs from all workflows
10 workflow runs
Merge pull request #473 from gregfowlerphd/463-acts-of-expressiv...
develop
Build, Test, Draft Release #10: Commit 8ba59d5 pushed by neilotte

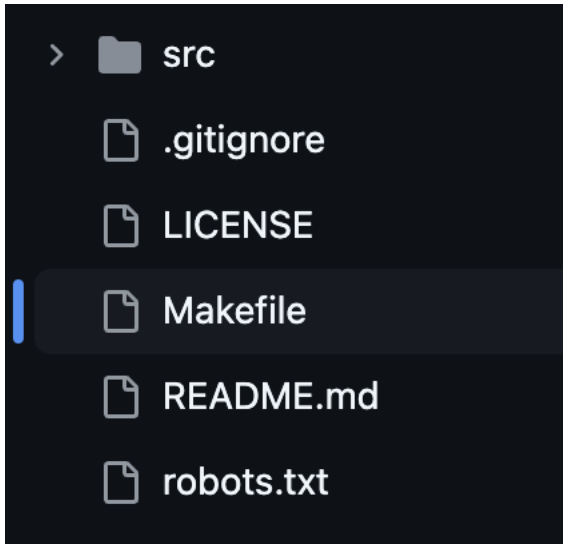

develop
Go to file
t
.github
deployment
templates
workflows
manage_release.yml
documentation
johnbeve forced update to makefile to run on develop
Code
Blame
80 lines (67 loc) · 2.4 KB
1 name: Build, Test, Draft Release
2
3 on:
4 push:
5 branches: [develop]
6
7 permissions:
8 contents: write


src
.gitignore
LICENSE
Makefile
README.md
robots.txt


src
.gitignore
LICENSE
Makefile
README.md
robots.txt



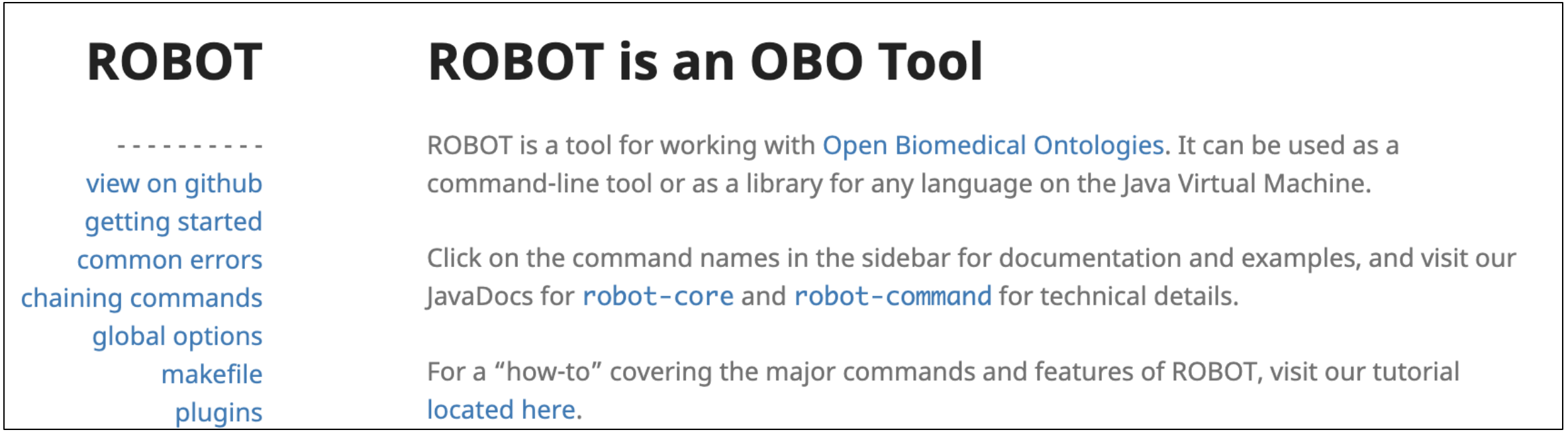

```
# Common Core Ontology Pipeline
# Adapted from previous works; see header comments for full attribution.
# Contact - John Beverley <johnbeve@buffalo.edu>

### Explanation ###
# The workflow involves two major steps: first, individual ontology files are checked and tested.
# After passing, they are merged into a single file, which is then checked and tested again.

# ----------------------------------------
# Project essentials
config.ONTOLOGY_PREFIX := CCO
config.BASE_IRI := http://www.ontologyrepository.com/CommonCoreOntologies/Mid/
config.DEV_IRI := $(config.BASE_IRI)/dev
config.MODULES_IRI := $(config.DEV_IRI)/modules

# Local project directories
config.SOURCE_DIR := src/
config.TEMP_DIR := build/artifacts
config.RELEASE_DIR := /
config.REPORTS_DIR := $(config.TEMP_DIR)
config.QUERIES_DIR := .github/deployment/sparql
config.LIBRARY_DIR := build/lib

# Settings
config.FAIL_ON_TEST_FAILURES := false
config.REPORT_FAIL_ON := none

# Branch-specific configurations
BRANCH := $(shell git rev-parse --abbrev-ref HEAD)

# File names for dev branch
DEV_FILES = \
    src/cco-modules/AgentOntology.ttl \
    src/cco-modules/ArtifactOntology.ttl \
    src/cco-modules/CurrencyUnitOntology.ttl \
    src/cco-modules/EventOntology.ttl \
    src/cco-modules/ExtendedRelationOntology.ttl \
    src/cco-modules/FacilityOntology.ttl \
    src/cco-modules/GeospatialOntology.ttl \
    src/cco-modules/QualityOntology.ttl \
    src/cco-modules/UnitsOfMeasureOntology.ttl \
    src/cco-modules/TimeOntology.ttl \
    src/cco-modules/InformationEntityOntology.ttl
```




```
# Common Core Ontology Pipeline
# Adapted from previous works; see header comments for full attribution.
# Contact - John Beverley <johnbeve@buffalo.edu>

### Explanation ###
# The workflow involves two major steps: first, individual ontology files are checked and tested.
# After passing, they are merged into a single file, which is then checked and tested again.

# ----------------------------------------
# Project essentials
config.ONTOLOGY_PREFIX := CCO
config.BASE_IRI := http://www.ontologyrepository.com/CommonCoreOntologies/Mid/
config.DEV_IRI := $(config.BASE_IRI)/dev
config.MODULES_IRI := $(config.DEV_IRI)/modules

# Local project directories
config.SOURCE_DIR := src/
config.TEMP_DIR := build/artifacts
config.RELEASE_DIR := /
config.REPORTS_DIR := $(config.TEMP_DIR)
config.QUERIES_DIR := .github/deployment/sparql
config.LIBRARY_DIR := build/lib

# Settings
config.FAIL_ON_TEST_FAILURES := false
config.REPORT_FAIL_ON := none

# Branch-specific configurations
BRANCH := $(shell git rev-parse --abbrev-ref HEAD)

# File name[illegible] for dev branch
DEV_[illegible] = \
    src/cco-modules/AgentOntology.ttl \
    src/cco-modules/ArtifactOntology.ttl \
    src/cco-modules/CurrencyUnitOntology.ttl \
    src/cco-modules/EventOntology.ttl \
    src/cco-modules/ExtendedRelationOntology.ttl \
    src/cco-modules/FacilityOntology.ttl \
    src/cco-modules/GeospatialOntology.ttl \
    src/cco-modules/QualityOntology.ttl \
    src/cco-modules/UnitsOfMeasureOntology.ttl \
    src/cco-modules/TimeOntology.ttl \
    src/cco-modules/InformationEntityOntology.ttl
```

```
# Download ROBOT JAR
ROBOT_FILE := $(config.LIBRARY_DIR)/robot.jar
$(ROBOT_FILE): setup
	curl -L -o $@ https://github.com/ontodev/robot/releases/download/v1.8.4/robot.jar
	chmod +x $@
```

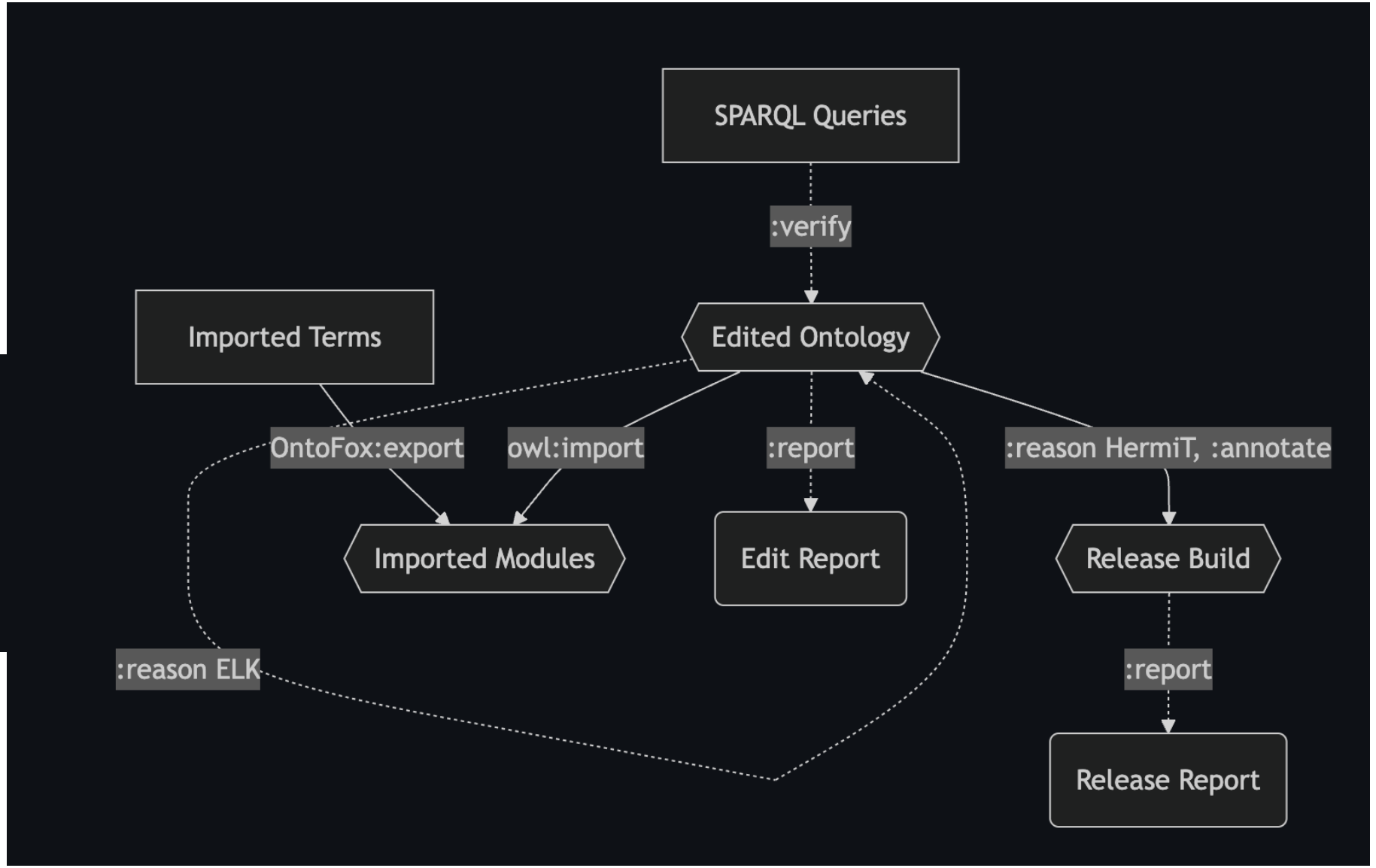
```
# Download ROBOT JAR
ROBOT_FILE := $(config.LIBRARY_DIR)/robot.jar
$(ROBOT_FILE): setup
	curl -L -o $@ https://github.com/ontodev/robot/releases/download/v1.8.4/robot.jar
	chmod +x $@
```



https://robot.obolibrary.org/

Diagram Key

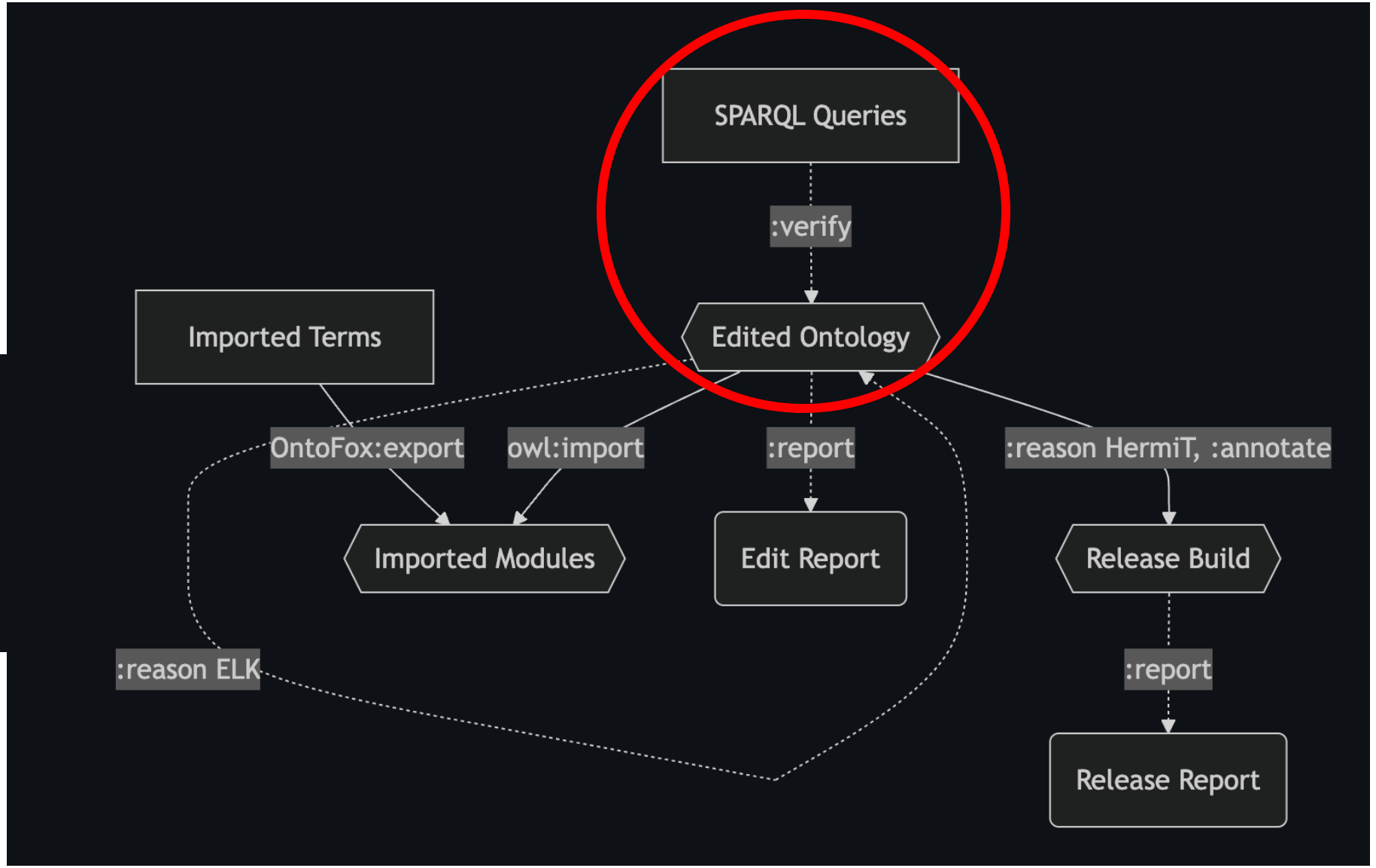
- Hexagons are ontologies
- Rectangles are SPARQL or text files
- Rounded boxes are spreadsheets
- Dotted lines involve automated tests
- ":" prefix means ROBOT command



https://github.com/tmprd/ontology-pipeline/blob/master/docs/Architecture.md

Diagram Key

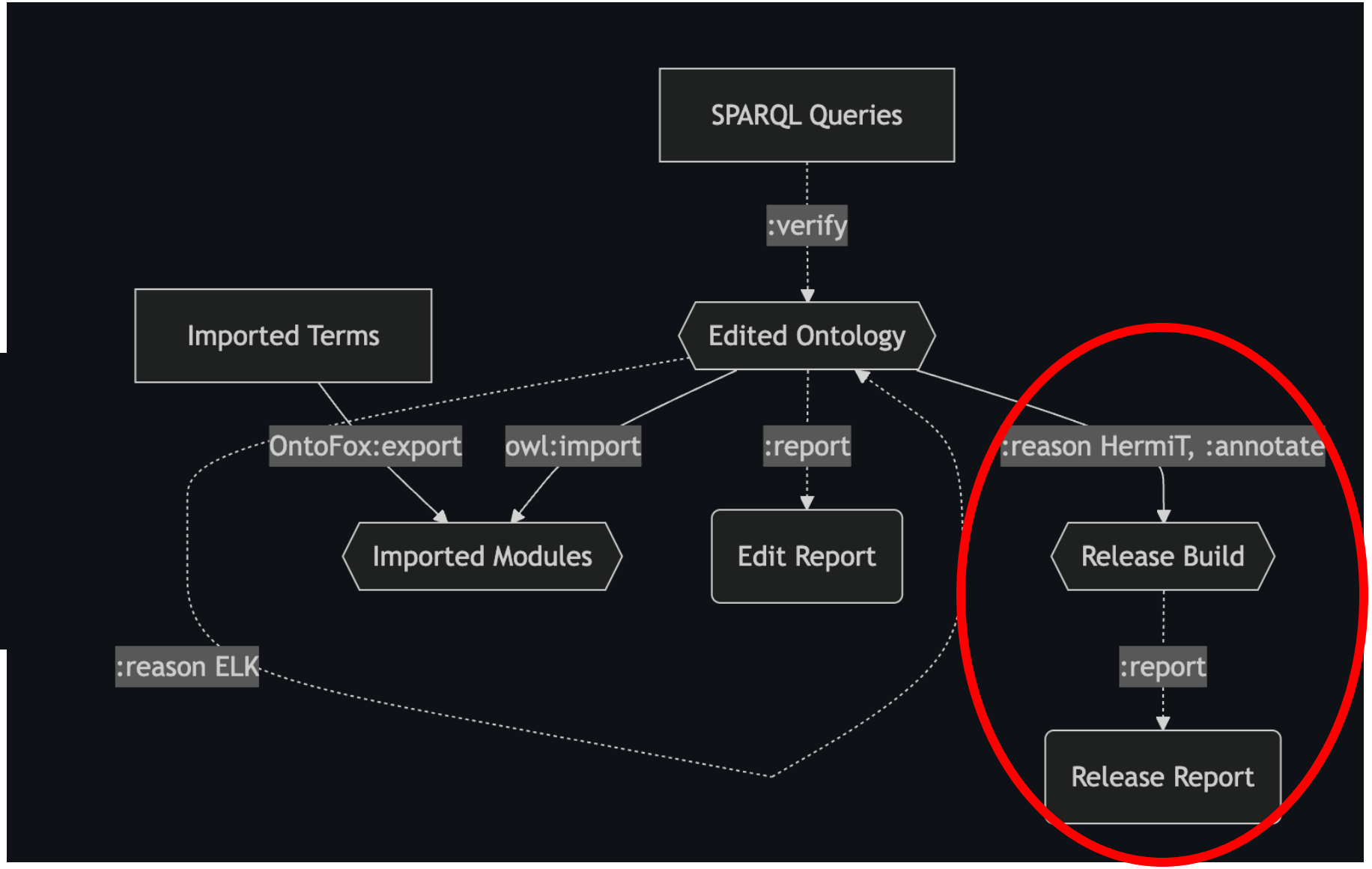
- Hexagons are ontologies
- Rectangles are SPARQL or text files
- Rounded boxes are spreadsheets
- Dotted lines involve automated tests
- ":" prefix means ROBOT command



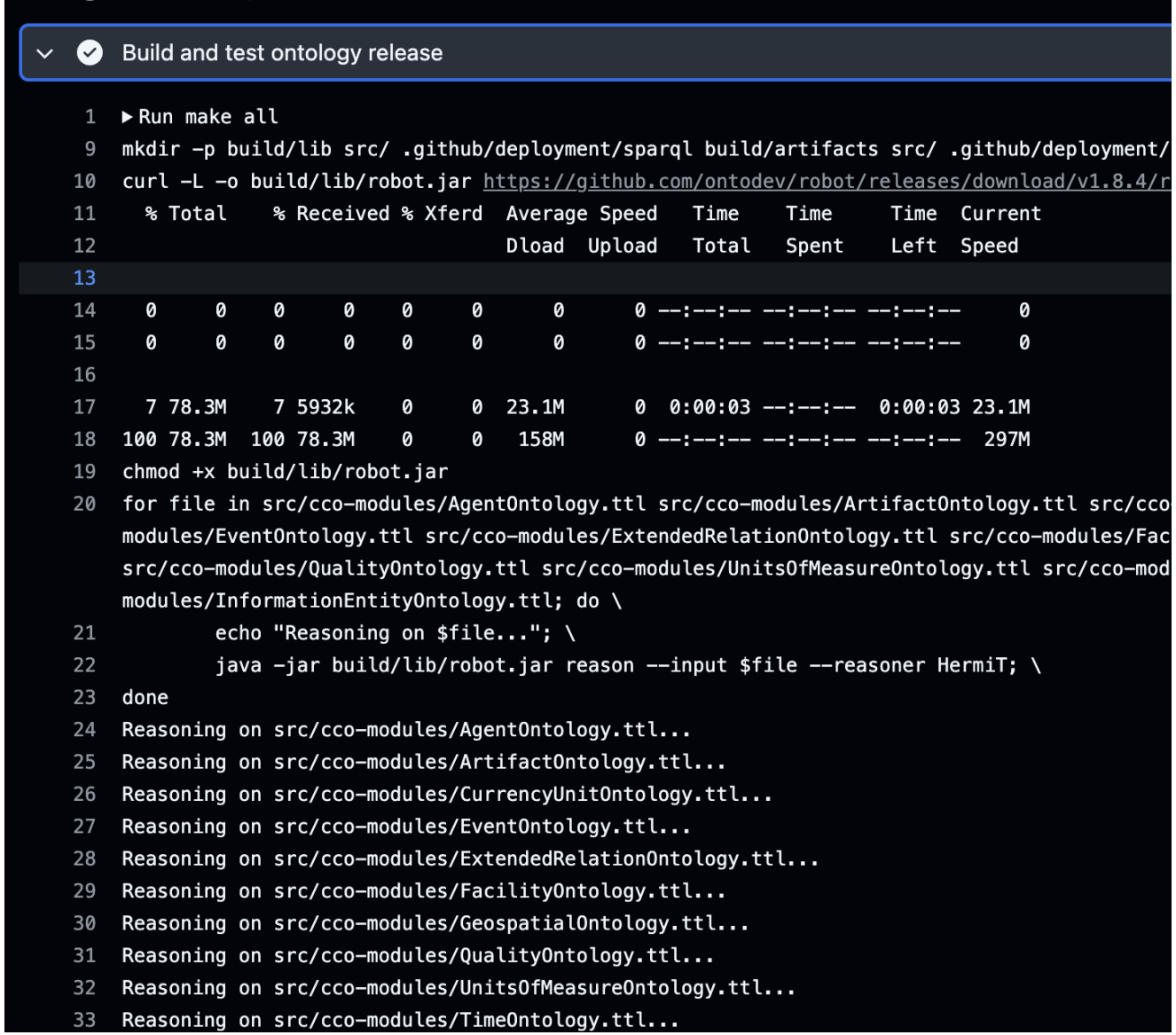
https://github.com/tmprd/ontology-pipeline/blob/master/docs/Architecture.md

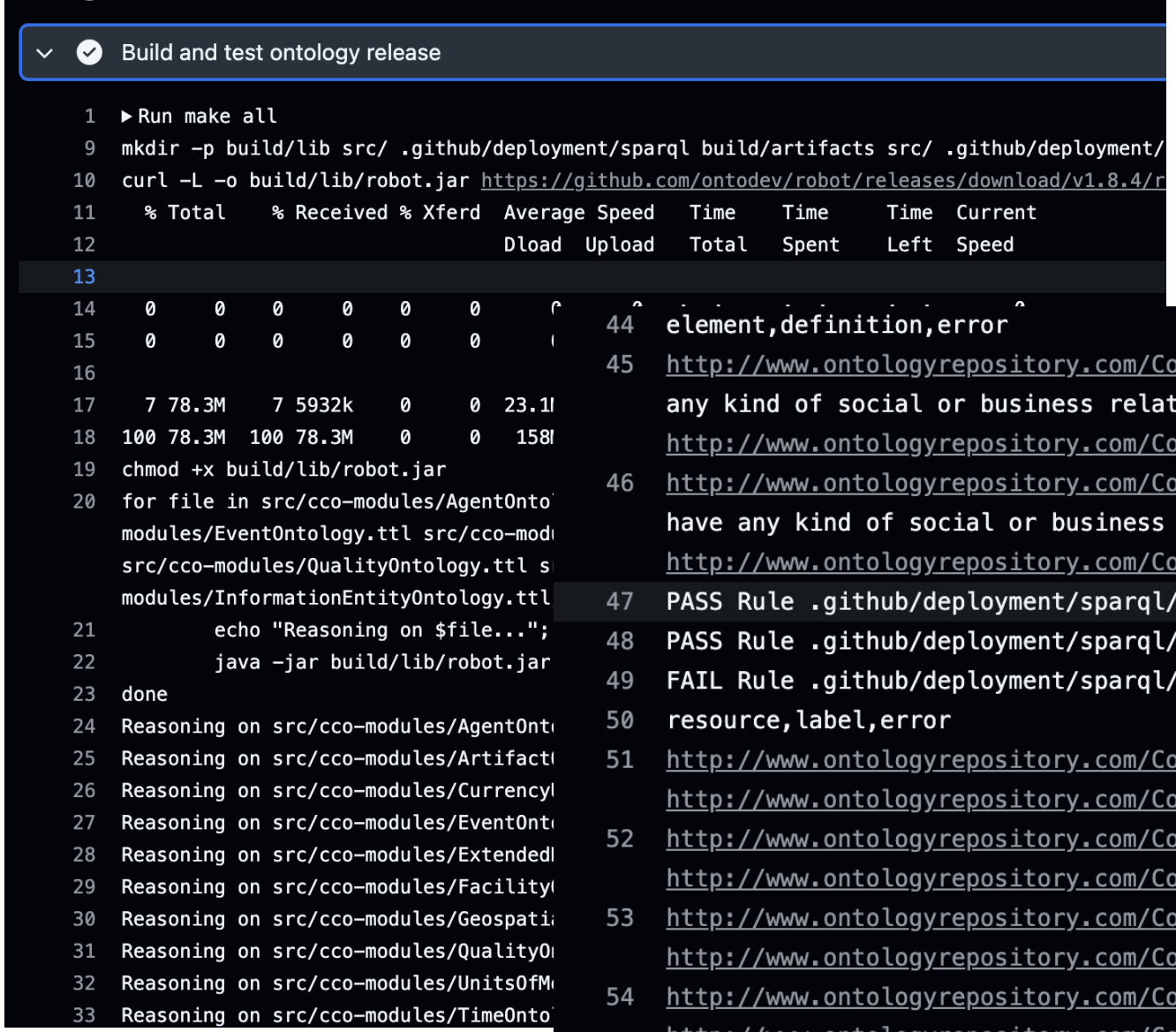
Diagram Key

- Hexagons are ontologies
- Rectangles are SPARQL or text files
- Rounded boxes are spreadsheets
- Dotted lines involve automated tests
- ":" prefix means ROBOT command

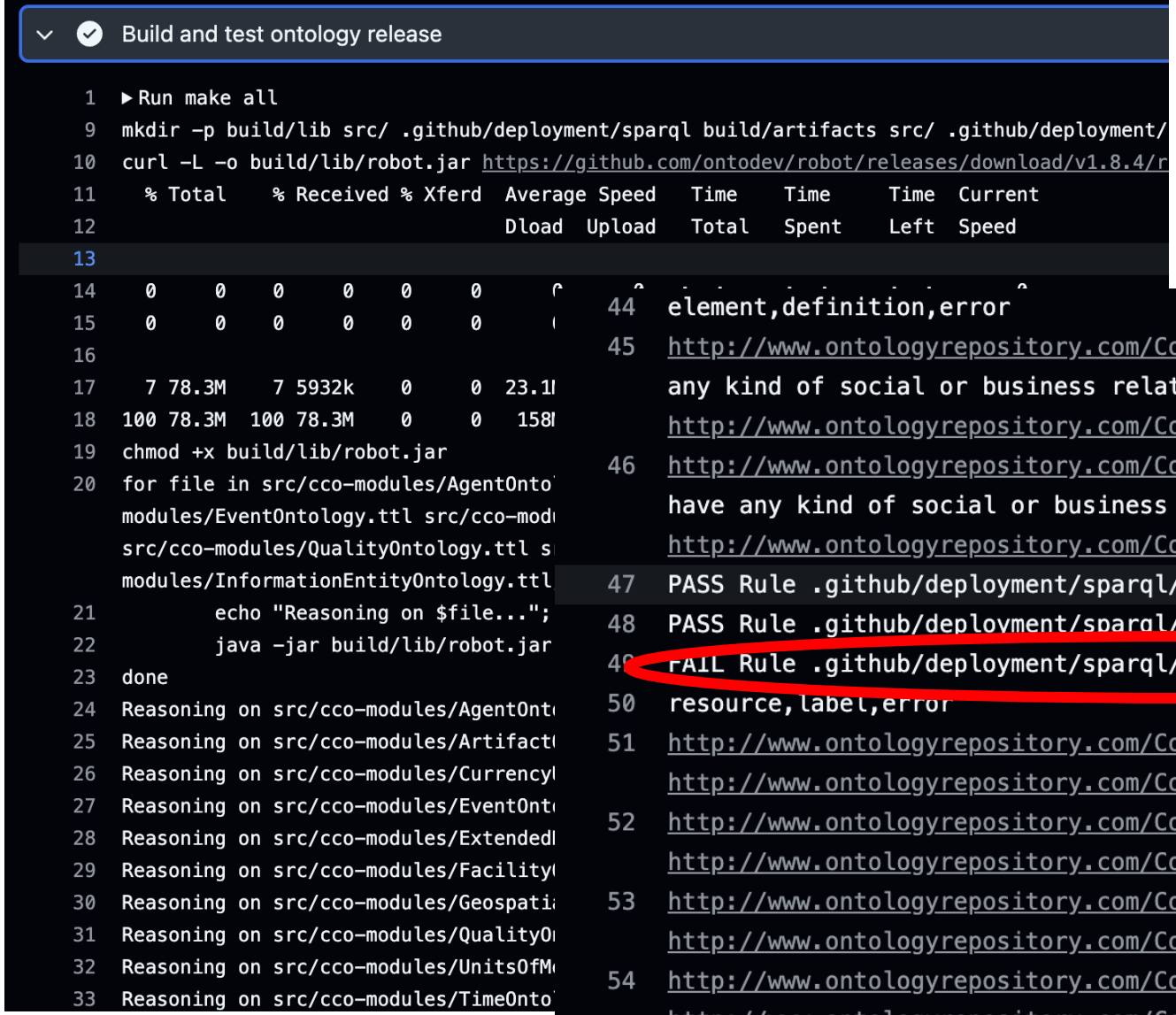


https://github.com/tmprd/ontology-pipeline/blob/master/docs/Architecture.md

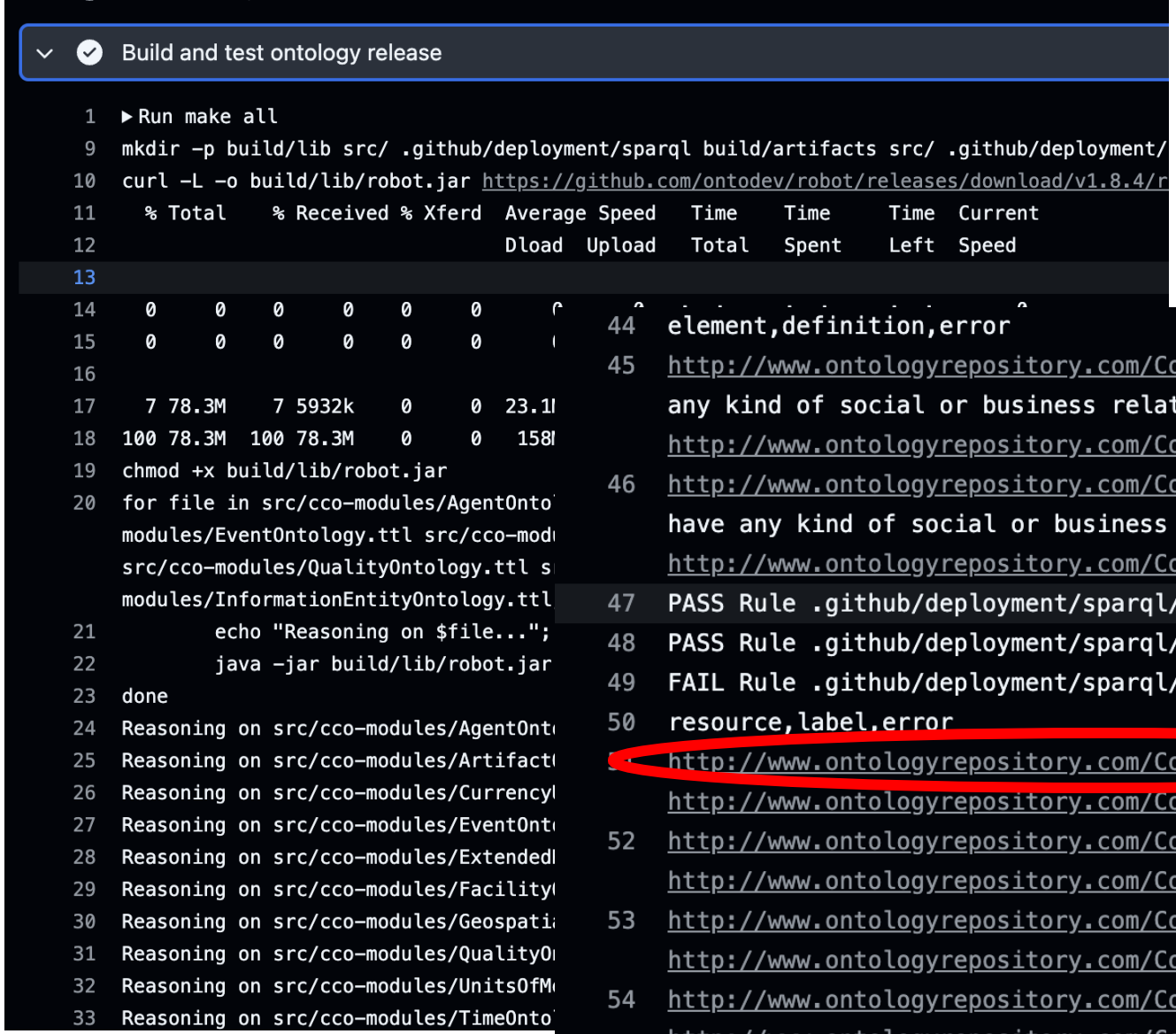

Build and test ontology release
1 ▶ Run make all
9 mkdir -p build/lib src/ .github/deployment/sparql build/artifacts src/ .github/deployment/
10 curl -L -o build/lib/robot.jar https://github.com/ontodev/robot/releases/download/v1.8.4/r
11 % Total % Received % Xferd Average Speed Time Time Time Current
12 Dload Upload Total Spent Left Speed
13
14 0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0
15 0 0 0 0 0 0 0 0 --:--:-- --:--:-- --:--:-- 0
16
17 7 78.3M 7 5932k 0 0 23.1M 0 0:00:03 --:--:-- 0:00:03 23.1M
18 100 78.3M 100 78.3M 0 0 158M 0 --:--:-- --:--:-- --:--:-- 297M
19 chmod +x build/lib/robot.jar
20 for file in src/cco-modules/AgentOntology.ttl src/cco-modules/ArtifactOntology.ttl src/cco
modules/EventOntology.ttl src/cco-modules/ExtendedRelationOntology.ttl src/cco-modules/Fac
src/cco-modules/QualityOntology.ttl src/cco-modules/UnitsOfMeasureOntology.ttl src/cco-mod
modules/InformationEntityOntology.ttl; do \
21 echo "Reasoning on $file..."; \
22 java -jar build/lib/robot.jar reason --input $file --reasoner HermiT; \
23 done
24 Reasoning on src/cco-modules/AgentOntology.ttl...
25 Reasoning on src/cco-modules/ArtifactOntology.ttl...
26 Reasoning on src/cco-modules/CurrencyUnitOntology.ttl...
27 Reasoning on src/cco-modules/EventOntology.ttl...
28 Reasoning on src/cco-modules/ExtendedRelationOntology.ttl...
29 Reasoning on src/cco-modules/FacilityOntology.ttl...
30 Reasoning on src/cco-modules/GeospatialOntology.ttl...
31 Reasoning on src/cco-modules/QualityOntology.ttl...
32 Reasoning on src/cco-modules/UnitsOfMeasureOntology.ttl...
33 Reasoning on src/cco-modules/TimeOntology.ttl...




```
element,definition,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate,"x is_affiliated_with y iff x and y are
any kind of social or business relationship.",WARNING: The following ontology elements have the same cco:defi
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate and http://www.ontologyrepository.com/Co
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with,"x is_affiliated_with y iff x and y
have any kind of social or business relationship.",WARNING: The following ontology elements have the same cco
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with and http://www.ontologyrepository.c
PASS Rule .github/deployment/sparql/duplicate_label.sparql: 0 violation(s)
PASS Rule .github/deployment/sparql/exactly_1_prefLabel_per_lang.sparql: 0 violation(s)
FAIL Rule .github/deployment/sparql/min_1_eng_def.sparql: 86 violation(s)
resource,label,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_father,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_father
http://www.ontologyrepository.com/CommonCoreOntologies/has_father_in_law,,WARNING: Missing definition for
```




```
element,definition,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate,"x is_affiliated_with y iff x and y are
any kind of social or business relationship.",WARNING: The following ontology elements have the same cco:defi
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate and http://www.ontologyrepository.com/Co
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with,"x is_affiliated_with y iff x and y
have any kind of social or business relationship.",WARNING: The following ontology elements have the same cco
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with and http://www.ontologyrepository.c
PASS Rule .github/deployment/sparql/duplicate_label.sparql: 0 violation(s)
PASS Rule .github/deployment/sparql/exactly_1_prefLabel_per_lang.sparql: 0 violation(s)
FAIL Rule .github/deployment/sparql/min_1_eng_def.sparql: 86 violation(s)
resource,label,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_father,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_father
http://www.ontologyrepository.com/CommonCoreOntologies/has_father_in_law,,WARNING: Missing definition for
```

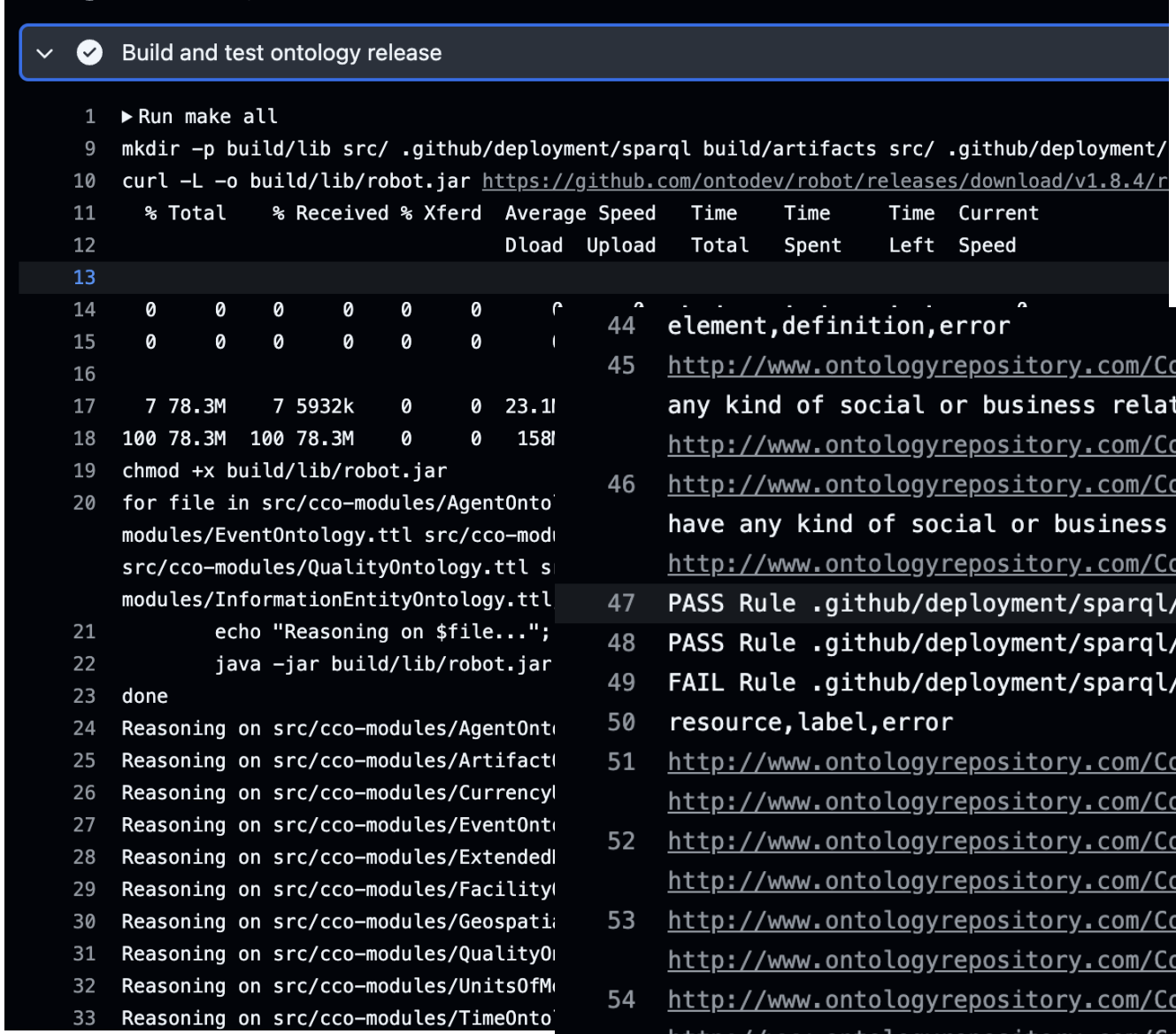


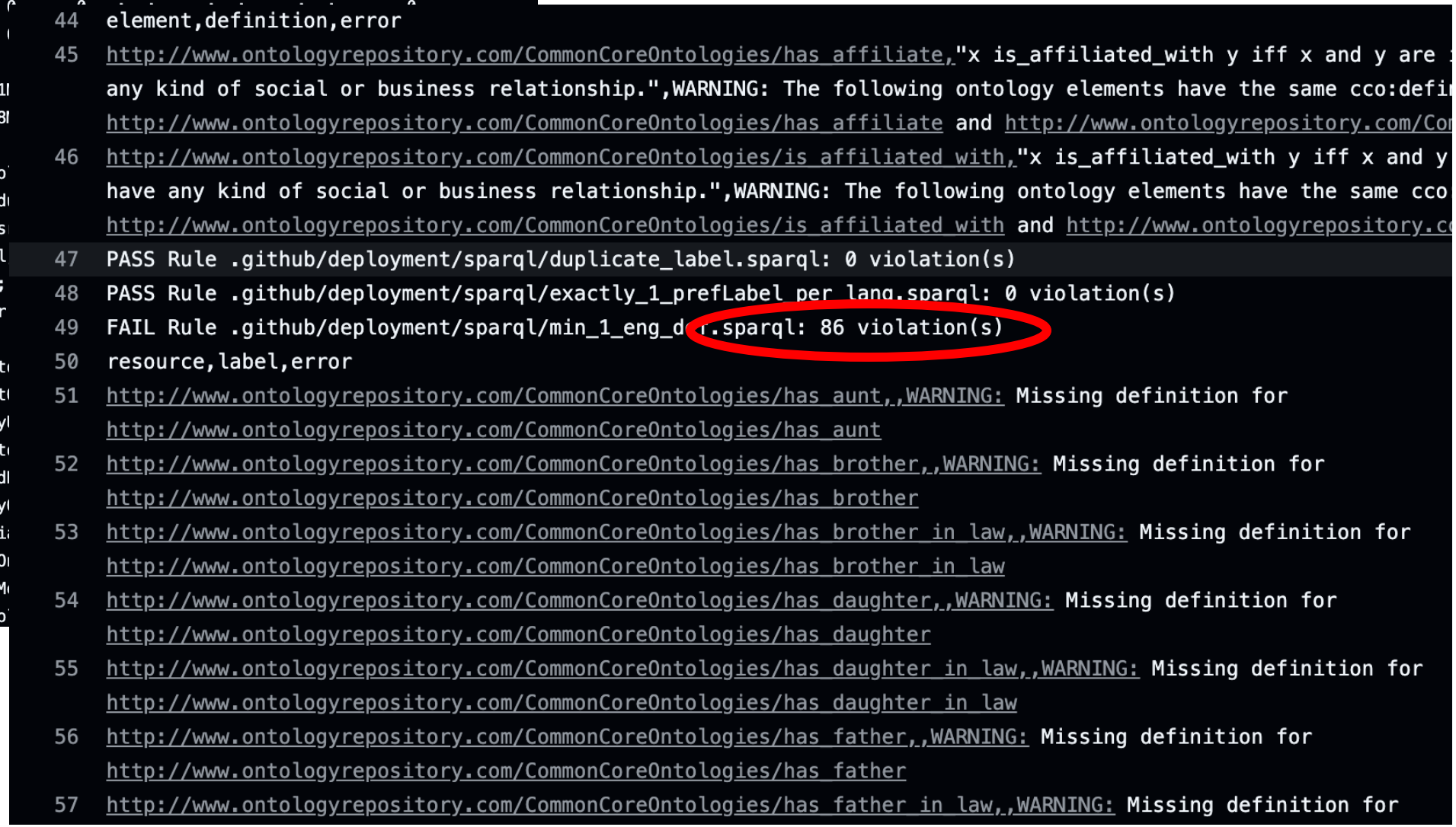

```
element,definition,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate,"x is_affiliated_with y iff x and y are
any kind of social or business relationship.",WARNING: The following ontology elements have the same cco:defi
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate and http://www.ontologyrepository.com/Co
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with,"x is_affiliated_with y iff x and y
have any kind of social or business relationship.",WARNING: The following ontology elements have the same cco
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with and http://www.ontologyrepository.c
PASS Rule .github/deployment/sparql/duplicate_label.sparql: 0 violation(s)
PASS Rule .github/deployment/sparql/exactly_1_prefLabel_per_lang.sparql: 0 violation(s)
FAIL Rule .github/deployment/sparql/min_1_eng_def.sparql: 86 violation(s)
resource,label,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_father,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_father
http://www.ontologyrepository.com/CommonCoreOntologies/has_father_in_law,,WARNING: Missing definition for
```

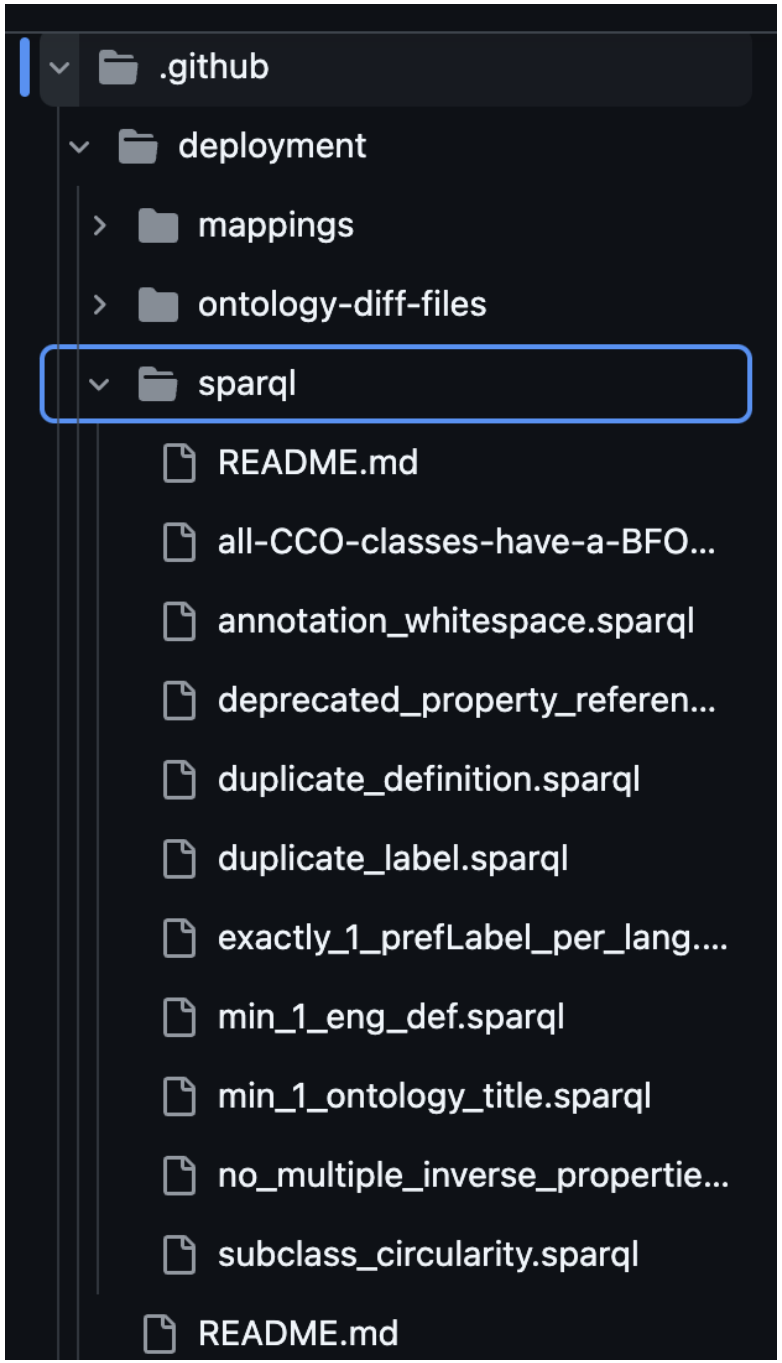
https://www.ontologyrepository.com/CommonCoreOntologies/has_aunt

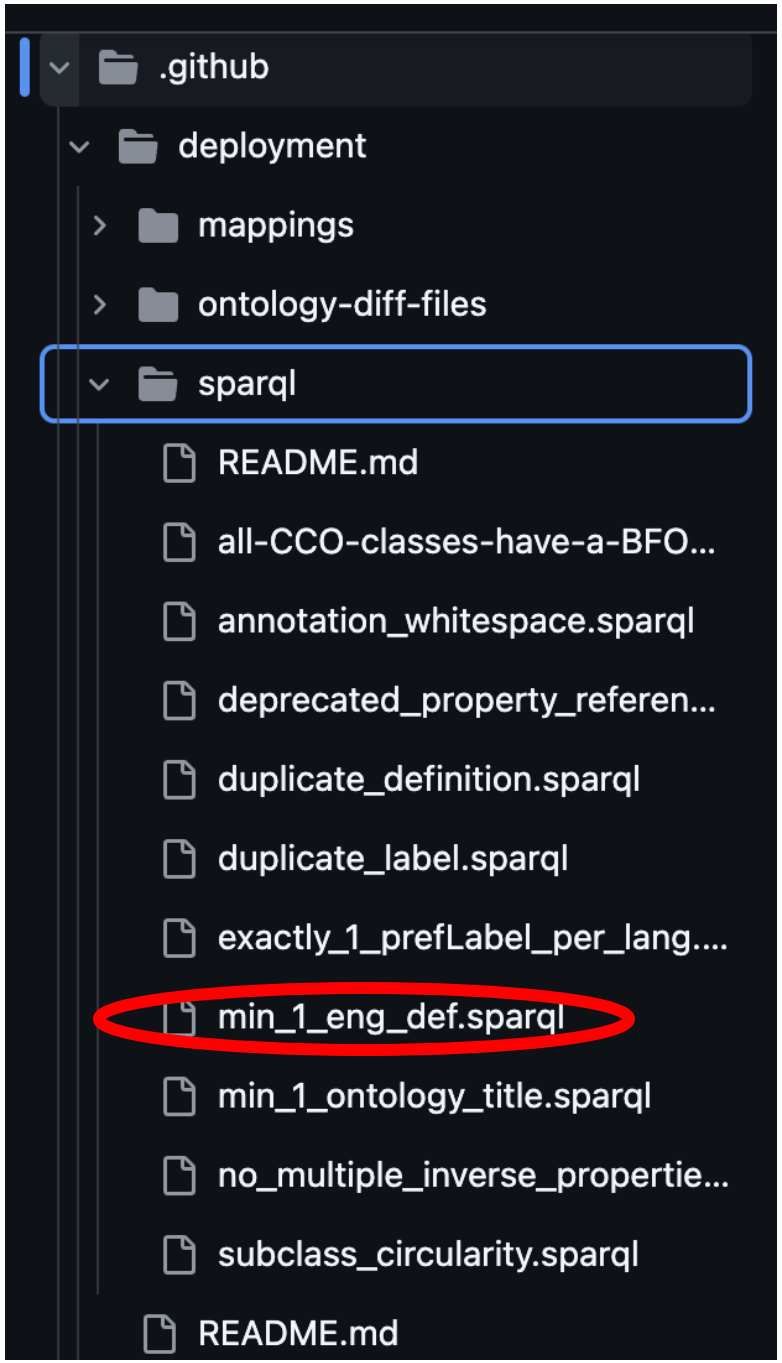
This XML file does not appear to have any style information associated with it. The document tree is shown below.


```
▼<rdf:RDF xmlns="http://www.w3.org/2002/07/owl#" xmlns:owl="http://www.w3.org/2002/07/owl#" xmlns:rdf="http://www.w3.or
 xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:CommonCoreOntologies="http://www.ontologyrepository.com/Commo
   <Ontology/>
   <!--
       ///////////////////////////////////////////////////////////////////////////////////////
       //
       // Annotation properties
       //
       ///////////////////////////////////////////////////////////////////////////////////////
         -->
   <!--  http://www.ontologyrepository.com/CommonCoreOntologies/is_curated_in_ontology  -->
   <AnnotationProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies/is_curated_in_ontology"/>
   <!--
       ///////////////////////////////////////////////////////////////////////////////////////
       //
       // Object Properties
       //
       ///////////////////////////////////////////////////////////////////////////////////////
         -->
   <!--  http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt  -->
  ▼<ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt">
     <rdfs:subPropertyOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies/has_familial_relationship_
     <inverseOf rdf:resource="http://www.ontologyrepository.com/CommonCoreOntologies/is_aunt_of"/>
     <CommonCoreOntologies:is_curated_in_ontology
     rdf:datatype="http://www.w3.org/2001/XMLSchema#anyURI">http://www.ontologyrepository.com/CommonCoreOntologies/Mid/A
     <rdfs:label xml:lang="en">has aunt</rdfs:label>
   </ObjectProperty>
   <!--  http://www.ontologyrepository.com/CommonCoreOntologies/has_familial_relationship_to  -->
   <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies/has_familial_relationship_to"/>
   <!--  http://www.ontologyrepository.com/CommonCoreOntologies/is_aunt_of  -->
   <ObjectProperty rdf:about="http://www.ontologyrepository.com/CommonCoreOntologies/is_aunt_of"/>
 </rdf:RDF>
 <!--  Generated by the OWL API (version 4.5.6) https://github.com/owlcs/owlapi  -->
```

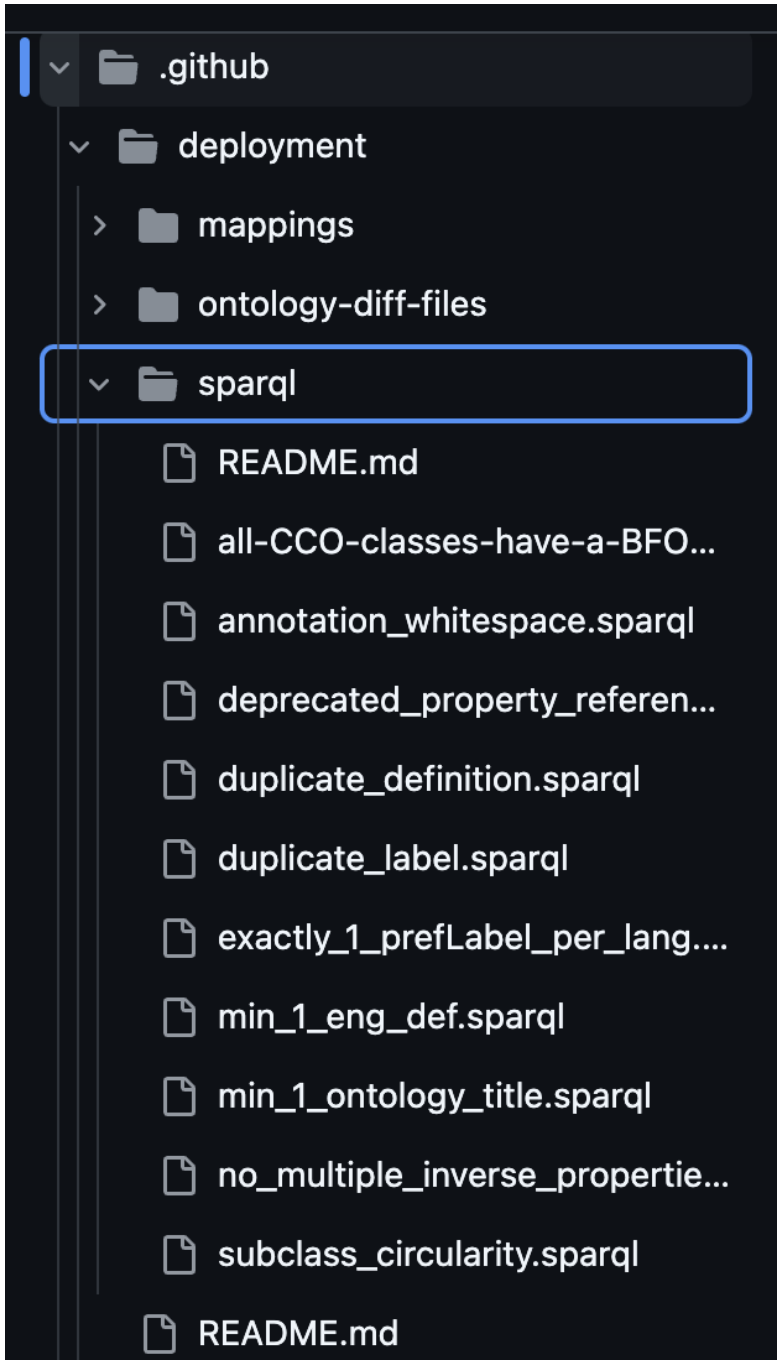

Build and test ontology release
▶ Run make all
mkdir -p build/lib src/ .github/deployment/sparql build/artifacts src/ .github/deployment/
curl -L -o build/lib/robot.jar https://github.com/ontodev/robot/releases/download/v1.8.4/r
% Total % Received % Xferd Average Speed Time Time Time Current
Dload Upload Total Spent Left Speed
0 0 0 0 0 0
0 0 0 0 0 0
7 78.3M 7 5932k 0 0 23.1
100 78.3M 100 78.3M 0 0 158
chmod +x build/lib/robot.jar
for file in src/cco-modules/AgentOnto
modules/EventOntology.ttl src/cco-mod
src/cco-modules/QualityOntology.ttl s
modules/InformationEntityOntology.ttl
echo "Reasoning on $file...";
java -jar build/lib/robot.jar
done
Reasoning on src/cco-modules/AgentOnt
Reasoning on src/cco-modules/Artifact
Reasoning on src/cco-modules/Currency
Reasoning on src/cco-modules/EventOnt
Reasoning on src/cco-modules/Extended
Reasoning on src/cco-modules/Facility
Reasoning on src/cco-modules/Geospati
Reasoning on src/cco-modules/QualityO
Reasoning on src/cco-modules/UnitsOfM
Reasoning on src/cco-modules/TimeOnto


element,definition,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate,"x is_affiliated_with y iff x and y are
any kind of social or business relationship.",WARNING: The following ontology elements have the same cco:defi
http://www.ontologyrepository.com/CommonCoreOntologies/has_affiliate and http://www.ontologyrepository.com/Co
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with,"x is_affiliated_with y iff x and y
have any kind of social or business relationship.",WARNING: The following ontology elements have the same cco
http://www.ontologyrepository.com/CommonCoreOntologies/is_affiliated_with and http://www.ontologyrepository.c
PASS Rule .github/deployment/sparql/duplicate_label.sparql: 0 violation(s)
PASS Rule .github/deployment/sparql/exactly_1_prefLabel_per_lang.sparql: 0 violation(s)
FAIL Rule .github/deployment/sparql/min_1_eng_d r.sparql: 86 violation(s)
resource,label,error
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_aunt
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_brother_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_daughter_in_law
http://www.ontologyrepository.com/CommonCoreOntologies/has_father,,WARNING: Missing definition for
http://www.ontologyrepository.com/CommonCoreOntologies/has_father
http://www.ontologyrepository.com/CommonCoreOntologies/has_father_in_law,,WARNING: Missing definition for

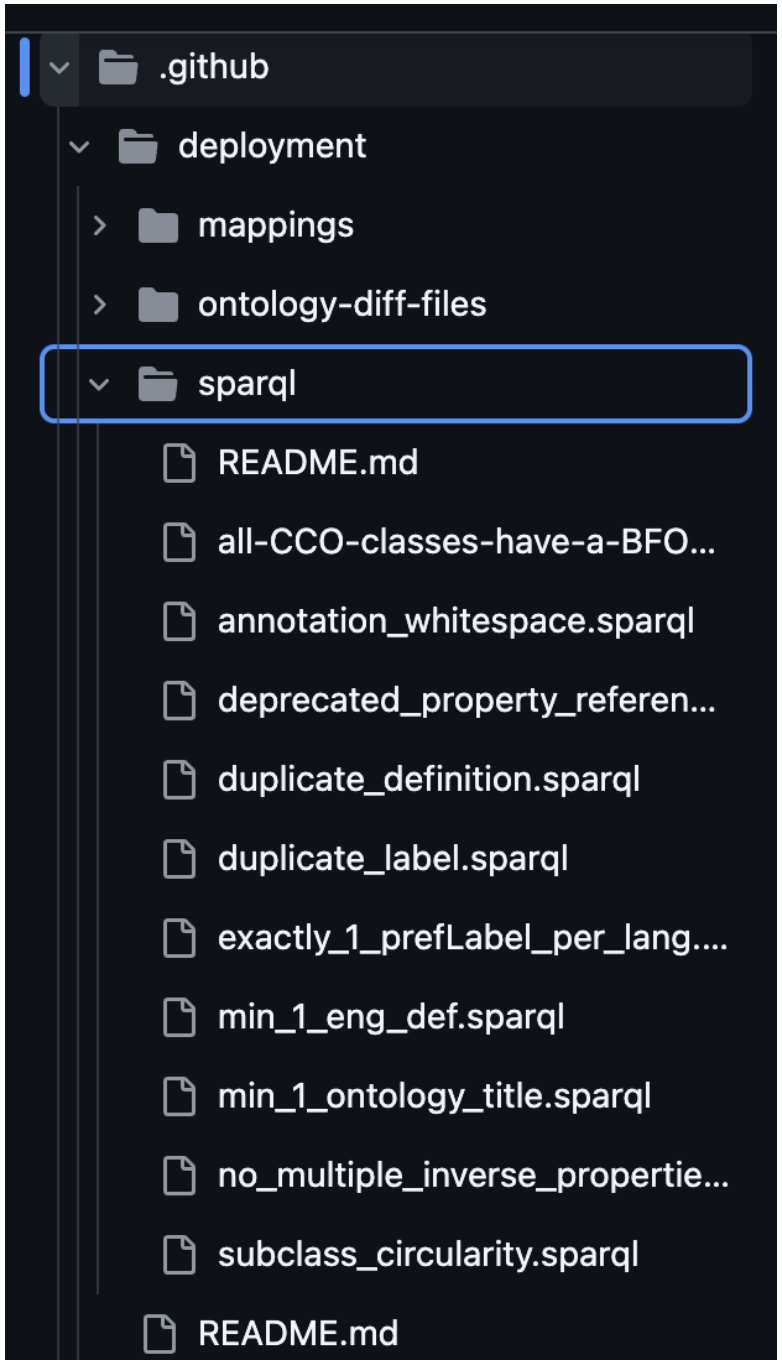

.github
deployment
mappings
ontology-diff-files
sparql
README.md
all-CCO-classes-have-a-BFO...
annotation_whitespace.sparql
deprecated_property_referen...
duplicate_definition.sparql
duplicate_label.sparql
exactly_1_prefLabel_per_lang....
min_1_eng_def.sparql
min_1_ontology_title.sparql
no_multiple_inverse_propertie...
subclass_circularity.sparql
README.md




```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an English language tag.
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```




```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an English language tag.
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

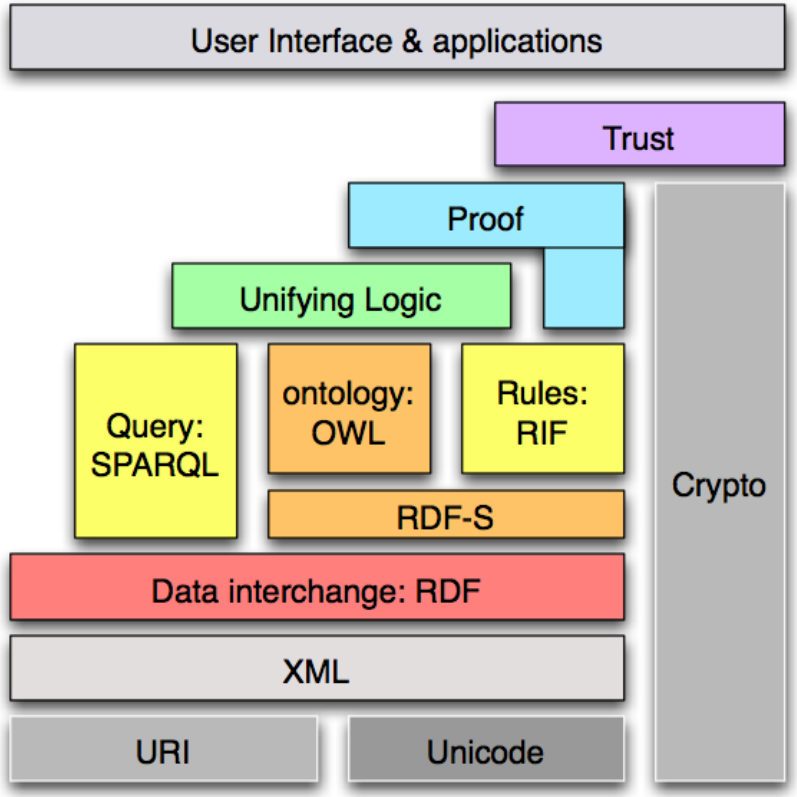



```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an English language tag.
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

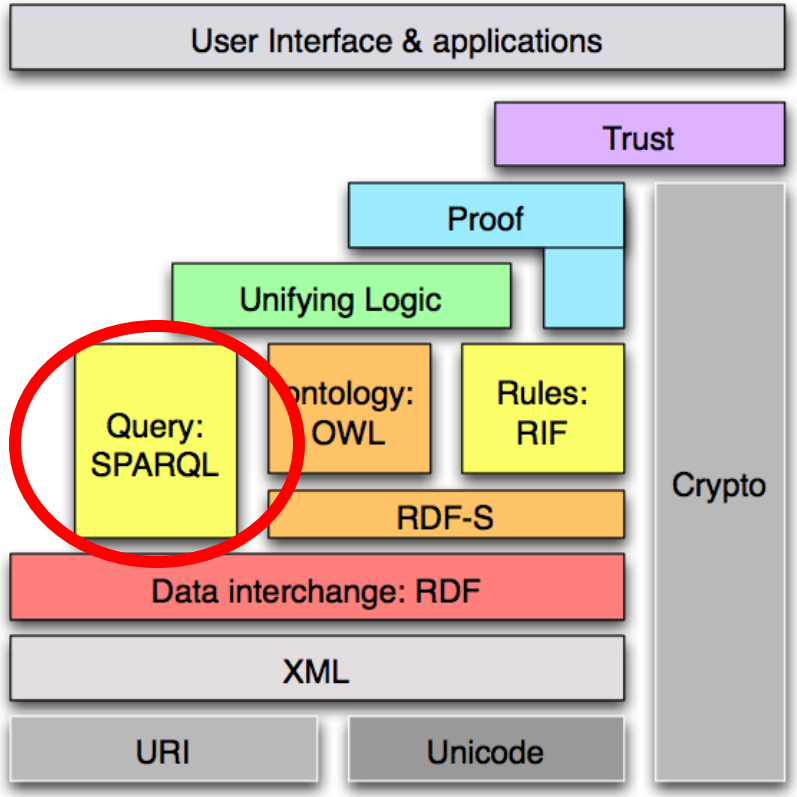
Outline

- Warmup
- The Common Core Ontologies Repository
- SPARQL

Semantic Web Stack

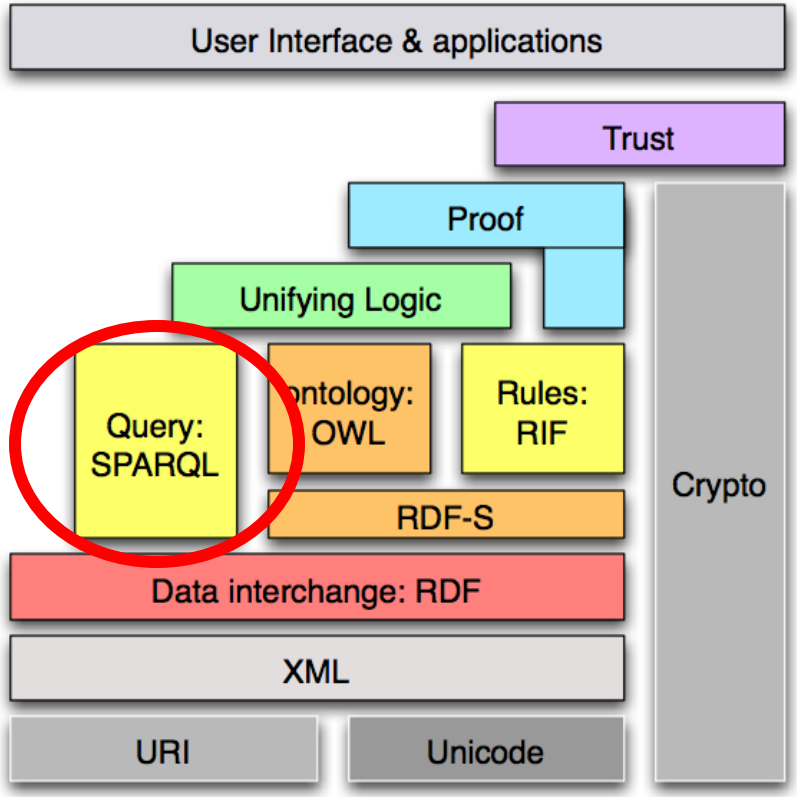


Semantic Web Stack



Semantic Web Stack

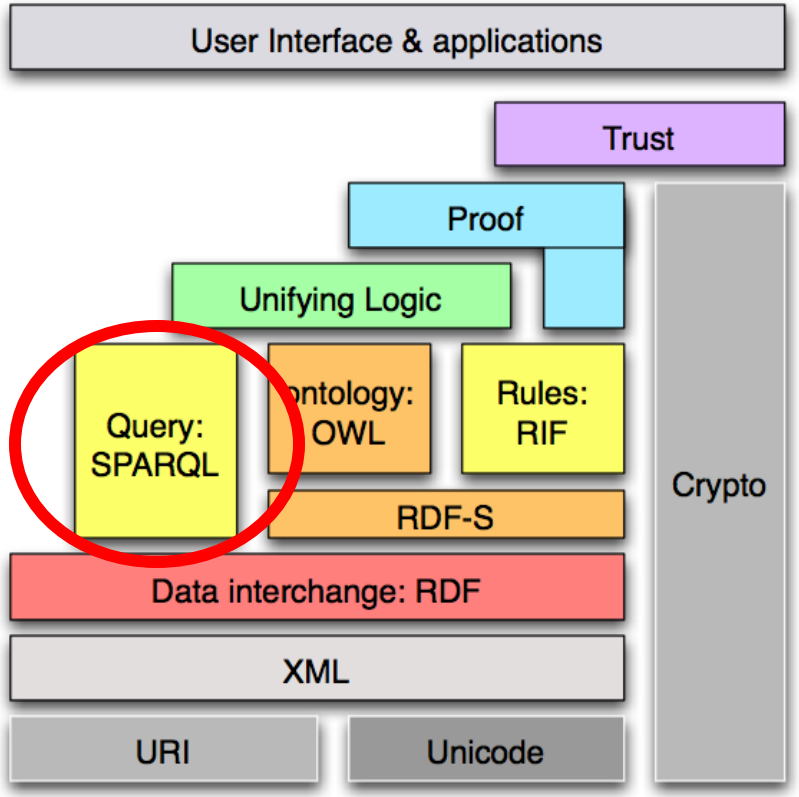
- "SPARQL" stands for:
  - **S**PARQL **P**rotocol
  - **A**nd **R**DF
  - **Q**uery **L**anguage



Semantic Web Stack

- "SPARQL" stands for:
  - **S**PARQL **P**rotocol
  - **A**nd **R**DF
  - **Q**uery **L**anguage

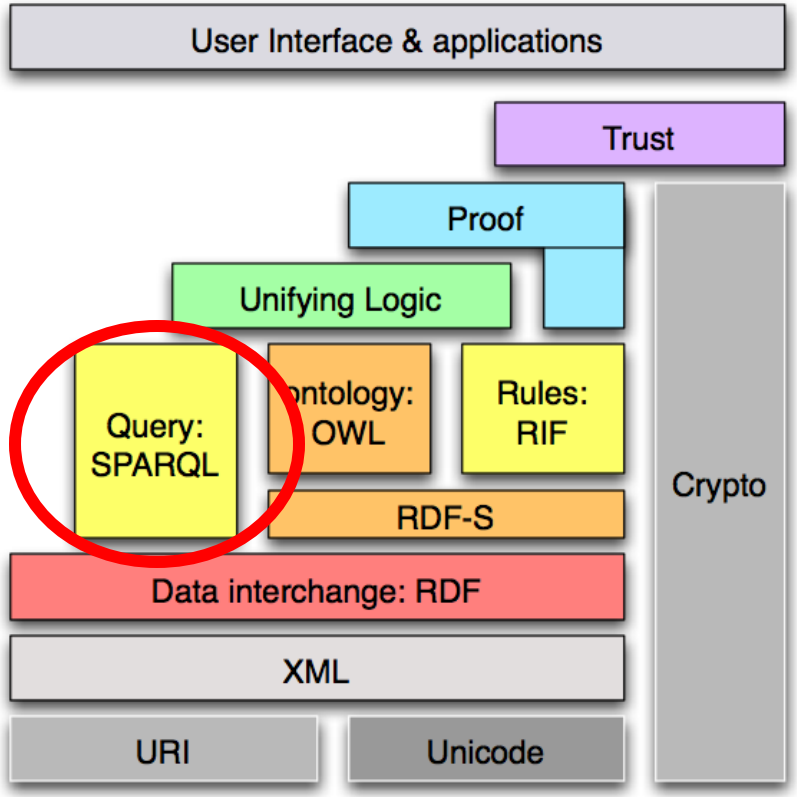
- SPARQL is a:
  - Core semantic web technology
  - *Query language* for RDF
  - A *protocol* for transmitting queries over HTTP



Semantic Web Stack

- "SPARQL" stands for:
  - **S**PARQL **P**rotocol
  - **A**nd **R**DF
  - **Q**uery **L**anguage

- SPARQL is a:
  - Core semantic web technology
  - *Query language* for RDF
  - A *protocol* for transmitting queries over HTTP



Query Languages

- Database query languages are languages used to extract from and manipulate data in information systems

- Traditionally, data was stored in *relational databases*, which were fixed, built on the closed world assumption, but somewhat easy to query using well-known languages like SQL

- RDF databases are flexible, adopt the open world assumption, and querying requires a language suited to these features

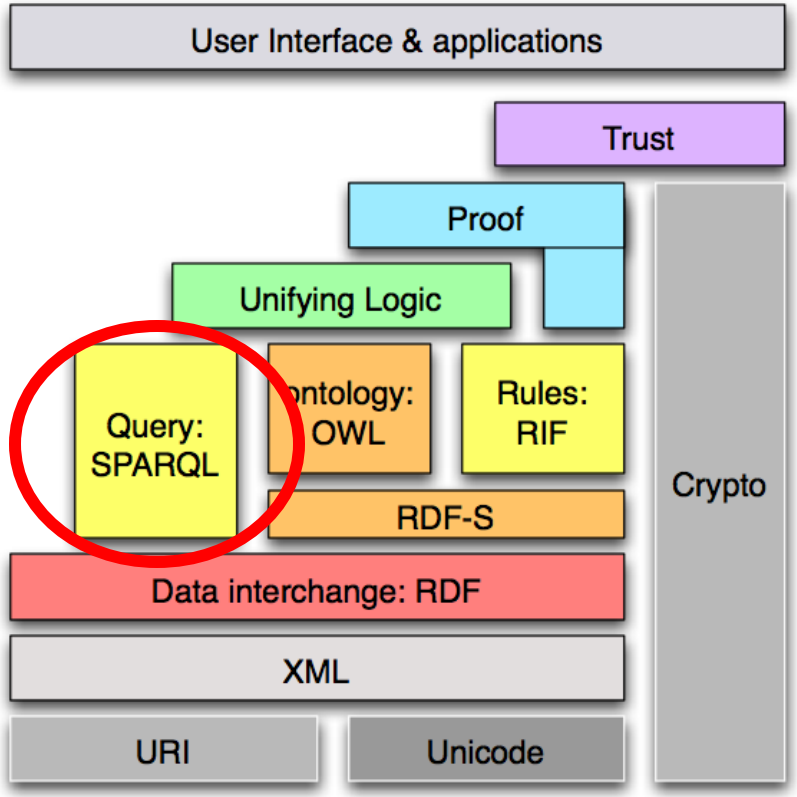
SPARQL Query Language

- SPARQL is a query language for RDF databases
- SPARQL shares much in common with query languages like SQL, and many differences
- SPARQL queries focus on what users want to know about the data; SQL queries focus on how the data is structured
- You will become a SPARQL Ninja™

Semantic Web Stack

- "SPARQL" stands for:
  - **S**PARQL **P**rotocol
  - **A**nd **R**DF
  - **Q**uery **L**anguage

- SPARQL is a:
  - Core semantic web technology
  - *Query language* for RDF
  - A *protocol* for transmitting queries over HTTP



HTTP Protocol

- SPARQL is protocol that specified how to send queries over the web to an endpoint using HTTP requests

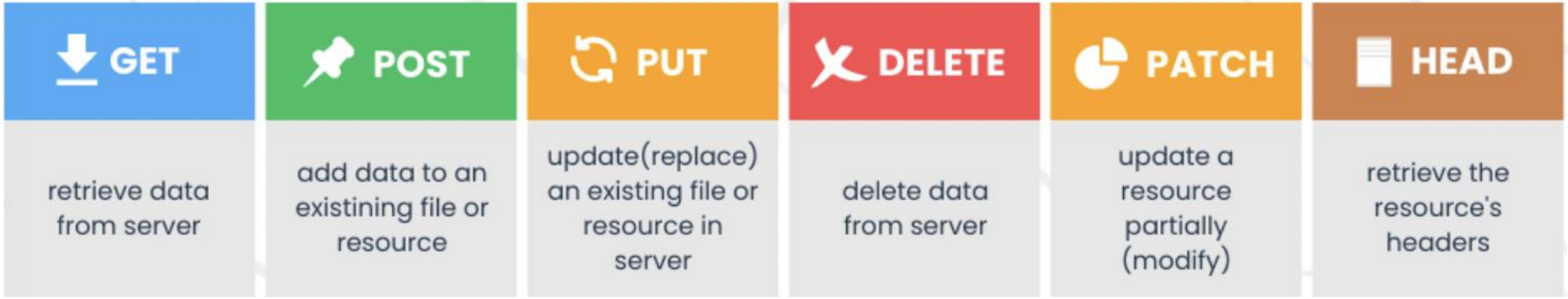
- A *protocol* in this context is a set of rules that prescribe how parties communicate

- HTTP is a protocol defined between a client – software that reads data from a server – and a server – where the data is stored

HTTP Protocol

- HTTP is a *request-response* protocol, a client has to send a request for data before the server will respond with that data

- HTTP request methods:



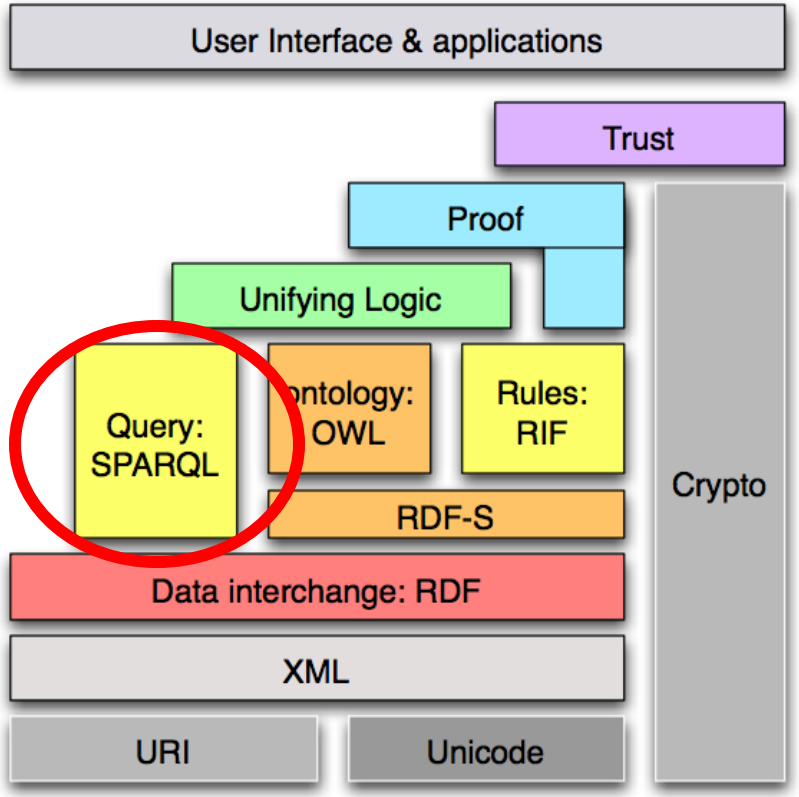
SPARQL Protocol

- Unlike other query languages, SPARQL is designed with a protocol that enables it to natively query over HTTP requests
- What this means is that data exposed by SPARQL on *any* server can be queried by any SPARQL client
- In contrast, query languages like SQL can only be queried locally
- SPARQL allows, for example, combining data from many different sources, dynamically

Semantic Web Stack

- "SPARQL" stands for:
  - **S**PARQL **P**rotocol
  - **A**nd **R**DF
  - **Q**uery **L**anguage

- SPARQL is a:
  - Core semantic web technology
  - *Query language* for RDF
  - A *protocol* for transmitting queries over HTTP



SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#


SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
   }
```

SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
```


```
SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
   }
```

SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
   }
```

Declare the namespace

then return any data…

SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
   }
```

Declare the namespace

then return any data…

…that satisfies the condition…

SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
  }
```

Declare the namespace

then return any data...

...that satisfies the condition...

...bears rdfs:label to something.

SPARQL Query Structure

```
PREFIX rdfs: http://www.w3.org/2000/01/rdf-schema#
SELECT ?subject ?label
WHERE
  {
    ?subject rdfs:label ?label .
  }
```

Declare the namespace

then return any data...

...that satisfies the condition...

...bears rdfs:label to something.

Returns list of subjects and their labels

SELECT and WHERE

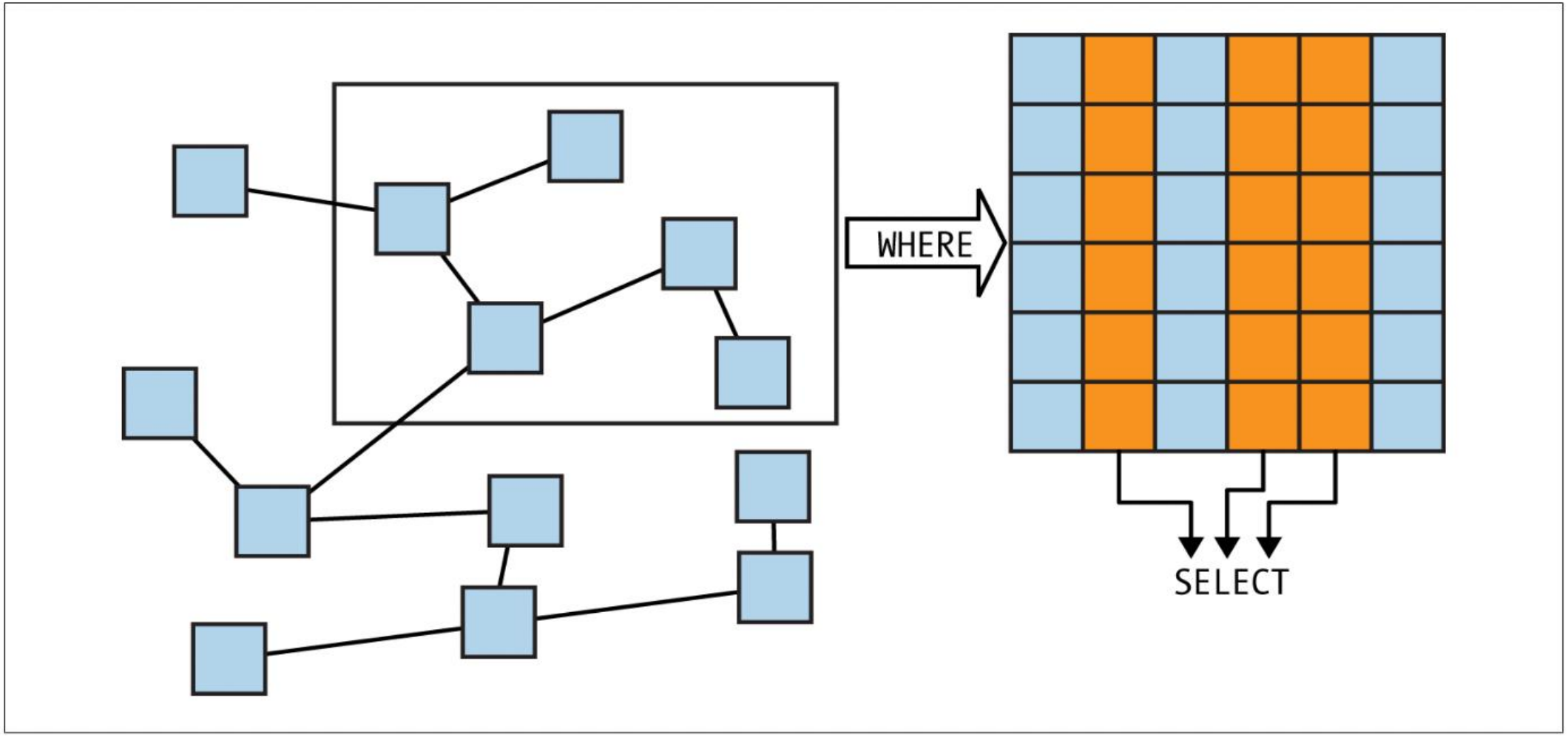


Figure 1-1. WHERE specifies data to pull out; SELECT picks which data to display

Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)

Query Text

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?subject ?label
WHERE {
 ?subject rdfs:label ?label .
} LIMIT 10
```

Endpoint

https://makg.org/sparql

Virtuoso SPARQL Query Editor

Default Data Set Name (Graph IRI)

Query Text

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?subject ?label
WHERE {
 ?subject rdfs:label ?label .
} LIMIT 10
```

Endpoint

https://makg.org/sparql

subject	label
w3.org/2002/07/owl#equivalentClass	"equivalentClass"
ht y3.org/2002/07/owl#equivalentProperty	"equivalentProperty"
ht w3.org/2002/07/owl#InverseFunctionalProperty	"InverseFunctionalProperty"
http://www.w3.org/2002/07/owl#SymmetricProperty	"SymmetricProperty"
http://www.w3.org/2002/07/owl#FunctionalProperty	"FunctionalProperty"
http://www.w3.org/2002/07/owl#inverseOf	"inverseOf"
http://www.w3.org/2002/07/owl#TransitiveProperty	"TransitiveProperty"
http://www.w3.org/2002/07/owl#Thing	"Thing"
http://www.w3.org/2002/07/owl#Class	"Class"
http://www.w3.org/2002/07/owl#Nothing	"Nothing"

OPTIONAL

- Because the basic unit of RDF is a triple, SPARQL queries by default return only triples that satisfy conditions in the WHERE clause

- That is, partial matches are not returned by default

- Consider, you might want to return the birth and death dates of everyone in a database...

OPTIONAL

```
SELECT ?subject ?birthday ?deathday
WHERE
  {
    ?subject ex:has_birthday ?birthday ;
                ex:has_deathday ?deathday .
   }
```

OPTIONAL

```
SELECT ?subject ?birthday ?deathday
WHERE
  {
    ?subject ex:has_birthday ?birthday ;
                ex:has_deathday ?deathday .
   }
```

Suppose the database is of my family, and includes my grandmother's birthday and deathday

OPTIONAL

```
SELECT ?subject ?birthday ?deathday
WHERE
  {
    ?subject ex:has_birthday ?birthday ;
                ex:has_deathday ?deathday .
   }
```

This query will return that information for my grandmother, but it will only return information for individuals who have *both* a birthday and a deathday

OPTIONAL

```
SELECT ?subject ?birthday ?deathday
WHERE
  {
    ?subject ex:has_birthday ?birthday ;
                ex:has_deathday ?deathday .
   }
```

It is plausible, however, that one might want to return individuals and their birthdays, even if they are still alive…

OPTIONAL

```
SELECT ?subject ?birthday ?deathday
WHERE
  {
    ?subject ex:has_birthday ?birthday .
    OPTIONAL
        {
          ?subject ex:has_deathday ?deathday .
        }
  }
```

OPTIONAL operates like a conditional; return everyone with a birthday and *if they have a deathday*, return that too

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
   {
      ?person rdf:type ex:Person ;
                    ex:name ?name ;
                     ex:age ?age .
    FILTER (xsd:integer(?age) >= 18)
   }
```

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
     ?person rdf:type ex:Person ;
                  ex:name ?name ;
                   ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
                ex:name ?name ;
                 ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace

SELECT variables...

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
                ex:name ?name ;
                 ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace

SELECT variables...

...WHERE...

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
            ex:name ?name ;
            ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace

SELECT variables...

...WHERE...

...someone is a Person...

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
            ex:name ?name ;
            ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace
SELECT variables...
...WHERE...
...someone is a Person...
...with a name...

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
            ex:name ?name ;
            ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace
SELECT variables...
...WHERE...
...someone is a Person...
...with a name...
...and age...

FILTER

- The FILTER keyword restricts the conditions in the WHERE clause using boolean and other functions

```
PREFIX ex: <https://example.com/>
SELECT ?person ?name
WHERE
  {
    ?person rdf:type ex:Person ;
            ex:name ?name ;
            ex:age ?age .
   FILTER (xsd:integer(?age) >= 18)
  }
```

Declare the namespace
SELECT variables...
...WHERE...
...someone is a Person...
...with a name...
...and age...
...but keep only the people over 18.

FILTER

- FILTER functions include:

Comparators: `<, >, =, <=, >=, !=`

Regular expressions: `regex(?x, "A.*")`

Test variable values: `isURI(?x), isBlank(?x), isLiteral(?x), bound(?x)`

And: `&&`

Or: `||`

Not: `!`

`()`

`YEAR(Date), MONTH(Date), DAY(Date)`
`HOURS(Date), MINUTES(Date), SECONDS(Date)`
`NOW()`

FILTER

Logical combinations of filter clauses, e.g.
FILTER (xsd:integer(?age)>18
&& xsd:integer(?age)<25)

- FILTER functions include:

Comparators: `<, >, =, <=, >=, !=`

Regular expressions: `regex(?x, "A.*")`

Test variable values: `isURI(?x), isBlank(?x), isLiteral(?x), bound(?x)`

And: `&&`

Or: `||`

Not: `!`

`()`

`YEAR(Date), MONTH(Date), DAY(Date)`
`HOURS(Date), MINUTES(Date), SECONDS(Date)`
`NOW()`

FILTER

Filter functions to restrict results, e.g.
FILTER (regex(?x, "hello", "i"))

- FILTER functions include:

Comparators: `<, >, =, <=, >=, !=`

Regular expressions: `regex(?x, "A.*")`

Test variable values: `isURI(?x), isBlank(?x), isLiteral(?x), bound(?x)`

And: &&

Or: ||

Not: !

()

```
YEAR(Date), MONTH(Date), DAY(Date)
HOURS(Date), MINUTES(Date), SECONDS(Date)
NOW()
```

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

Declare the namespace

Declare return variables

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

VALUES type means that "type" is a variable ranging over everything in the brackets

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

That is, for any resource that is…

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

...or an owl:objectProperty

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

Return resource even if the definition is missing

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

...or is missing an xsd English language tag

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

But indeed, keep only the results that don't have a definition, i.e. ?englishDefinition is unbound because it's empty

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

Ignore blank nodes

```
# Title:
#    Definition Required
# Constraint Description:
#    Any class or object property must have a non-empty definition with an Englis
# Severity:
#     Warning

PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX cco: <http://www.ontologyrepository.com/CommonCoreOntologies/>

SELECT DISTINCT ?resource ?label ?error
WHERE {
VALUES ?type {owl:Class owl:ObjectProperty}
    ?resource a ?type .
    OPTIONAL {
        ?resource cco:definition ?englishDefinition .
        FILTER (langMatches(lang(?englishDefinition), "en"))
        }
    FILTER(!bound(?englishDefinition))
    FILTER(!isBlank(?resource))
    BIND (concat("WARNING: Missing definition for ", str(?resource)) AS ?error)
}
ORDER BY ?resource
```

Format results with a description of the error

Quality Control

- As you investigate the CCO repository, you'll note there are not many SPARQL queries being run against builds

Quality Control

- As you investigate the CCO repository, you'll note there are not many SPARQL queries being run against builds

- If only there was some group of individuals that either already knew or was ~~being forced~~ willing to learn SPARQL...

WE WILL GET THERE

BUT FIRST, WE PRACTICE

Dbpedia Challenge:

https://dbpedia.org/sparql

1. What is the birthdate of Barack Obama?
2. What city is the capital of Spain?
3. What is the highest peak in the U.S.?
4. What is the release date of the movie "The Dark Knight"?
5. What country won the FIFA World Cup in 2018?
6. Who directed the movie "Jurassic Park"?

Dbpedia Challenge:
https://dbpedia.org/sparql

1. What are the ten most populous cities in China?
2. What are the birthdates of every U.S. president who served in the 20th century?
3. What three cities in the U.S. have the highest percentage of foreign-born residents?
4. Which paintings by Vincent Van Gogh are currently located in museums in the U.S.?
5. How many rdfs labels include the string "lio"?