

FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Mantenimiento evolutivo de ROH



HERCULES

Dr. Mikel Emaldi Manrique
m.emaldi@deusto.es

FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Agenda

- ☐ 09:45 – 11:00: Github Actions, preguntas de competencia y razonadores
- ☐ 11:00 – 11:30: Descanso
- ☐ 12:45 – 14:00: Taller



FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Mantenimiento de la ROH



HÈRCULES



FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Contenidos

- Proceso de modificación de la ROH
- Introducción a CI/CD
- CI/CD con GitHub actions
- Ejemplos



Una manera de hacer Europa

Proceso de modificación de la ROH

Protégé

- Editor de ontologías
 - Open source
 - Versiones de escritorio y [web](#)
- Ontología: descripción explícita y formal de los conceptos en un dominio determinado
 - Clases: los conceptos a describir
 - Propiedades: describen los diferentes atributos de los conceptos y las relaciones entre clases
 - Object properties: relaciones entre dos instancias de una clase
 - Data properties: relaciones entre una instancia de una clase y un XML Schema datatype

Una manera de hacer Europa

Proceso de modificación de la ROH OOPS

- OntOlogy Pitfall Scanner: <http://oops.linkeddata.es/>
 - Poveda-Villalón, María, Asunción Gómez-Pérez, and Mari Carmen Suárez-Figueroa. "OOPS!(Ontology Pitfall Scanner!): An on-line tool for ontology evaluation." International Journal on Semantic Web and Information Systems (IJSWIS) 10.2 (2014): 7-34.
- Detecta los errores y fallos más comunes de las ontologías

Una manera de hacer Europa

Proceso de modificación de la ROH Git/GitHub

- Uso de Git para el control de versiones.
- GitHub como repositorio de código.
- Uso branches y pull-request.
 - Rama master/main para producción.
 - Resto de ramas para desarrollo.
- Controles para poder integrar en master/main.
- Rama *staging* para ejecutar los tests de integración.
- Sólo se puede integrar a *master* desde *staging*.

Una manera de hacer Europa

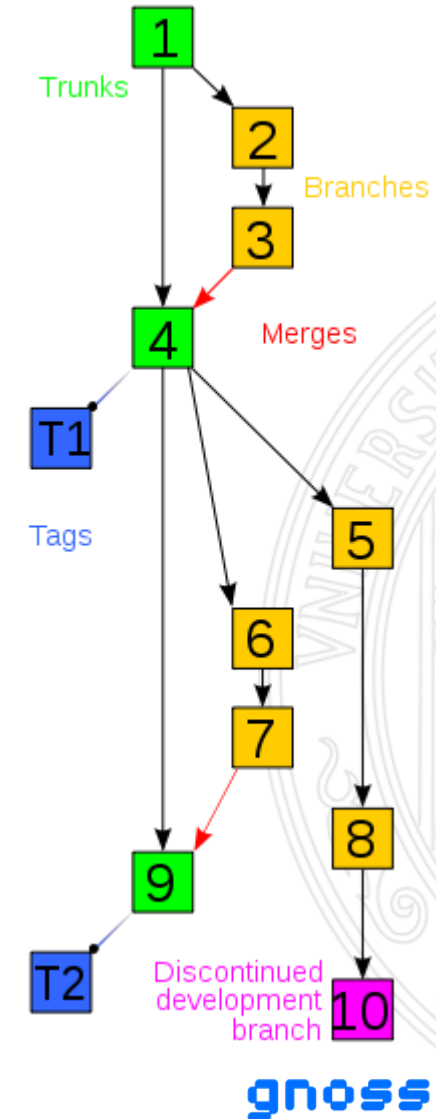
Terminology: Trunk, Branch and Tag

- **Version management:**

- all code resides in “**trunk**”
 - Where all the code resides
 - Has the latest not necessary most stable version
- code version are detached in “**branches**”
 - custom development branches
 - version branches
- snapshots for releases are “**tagged**”
 - tags are snapshots usually made on version branches

- **Release management:**

- A **release** is a snapshot of a version branch
- Releases are being deployed to server environments for live or production environments



Una manera de hacer Europa

Proceso de modificación de la ROH Git/GitHub



Introducción a CI/CD

- Continuous Integration/Continuous Delivery/Deployment
- Automatización de diferentes tareas de integración y despliegue
- CI: integración de código nuevo en el existente
 - P. ej., ejecución automática de tests unitarios y de integración.
- CD: despliegue del código nuevo en el entorno de producción
- GitHub Actions.
 - GitLab pipelines, Codeship, Travis-CI, Jenkins...



Introducción a CI/CD

Tipos de Test

Test Type	Description	Impact on CI Build
Unit	Tests discrete units of code to verify correct behaviour; written and performed by developers as part of the development process.	Easily included in the build cycle; very low impact on build speed.
Integration/Component	Tests to verify specific components, including their interaction with other internal and external components; may exercise code not exposed to clients or end users.	May be included in the build cycle; tends to run longer than unit tests so there is some impact on build speed.

FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Preguntas de competencia ASIO.



HĒRCULES



Una manera de hacer Europa

Preguntas de competencia ASIO.

- Una serie de consultas SPARQL sobre un dataset modelado según la ROH
 - t-box: contiene los axiomas que definen las clases y las relaciones en una ontología
 - a-box: contiene los asertos acerca de los *individuales* (instancias) del dominio.
- Su objetivo es comprobar que la ROH cumple con los requisitos establecidos

Una manera de hacer Europa

Preguntas de competencia ASIO. SPARQL

- **SPARQL Protocol and RDF Query Language**
- Lenguaje para la consultas de grafos RDF.
- Se basa en buscar patrones de *tripletas* con la forma ?sujeto ?predicado ?objeto
- <https://github.com/HerculesCRUE/GnossDeustoBackend/tree/master/validation-questions/sparql-query>

Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q1 - Centros de investigación que trabajan en un área/disciplina específica:

```
PREFIX roh: <http://purl.org/roh#>
PREFIX uneskos: <http://purl.org/roh/unesco-individuals#>

SELECT ?centro WHERE {
    ?centro a roh:ResearchGroup ;
            roh:hasKnowledgeArea uneskos:120304 .
}
```


Preguntas de competencia ASIO.

- Q4 - Centros/estructuras de investigación que posean sellos de calidad asociados:

```
PREFIX roh: <http://purl.org/roh#>
SELECT ?center ?accreditationTitle
WHERE {
  ?center a roh:ResearchGroup ;
          roh:hasAccreditation ?accreditation .
  ?accreditation roh:title ?accreditationTitle
}
```

FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q5 - Listado de los centros/estructuras de investigación que hayan realizado proyectos y su respectiva convocatoria:

```
SELECT DISTINCT ?centro ?fundingProgram ?project
WHERE {
  ?fundingAmount roh:grants ?centro .
  ?funding ro:hasPart ?fundingAmount ;
    roh:fundedBy ?fundingProgram ;
    roh:supports ?project .
  OPTIONAL {
    ?project roh:projectStatus ?projectStatus .
  }
  FILTER (!BOUND(?projectStatus) || ?projectStatus != "PROPOSAL_SUBMITTED")
}
```

Preguntas de competencia ASIO.

- Q6 - Listado de la producción científica en un determinado rango de fechas de un centro/estructura de investigación en un área/disciplina:

```
SELECT DISTINCT ?organization ?researchObject ?knowledgeArea
WHERE {
    ?researchObject      a roh:ResearchObject ;
                        vivo:dateIssued ?dateIssued ;
                        roh:hasKnowledgeArea ?knowledgeArea ;
                        bibo:authorList ?authorList .

    ?authorList rdfs:member ?author .
    ?author roh:hasPosition ?position .
    ?organization a foaf:Organization .
    ?position vivo:relates ?organization .
    ?dateIssued vivo:dateTime ?dateTime .
    FILTER (YEAR(?dateTime) >= "2010"^^xsd:integer && YEAR(?dateTime) <= "2020"^^xsd:integer)
}
```

Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q7 - Artículos publicados en revistas, según las comunidades autónomas:

```
SELECT
(COUNT(DISTINCT ?publication) as ?publicationCount) ?comunidadAutonoma
WHERE {
    ?publication a iao:IAO_0000013 ;
                 vivo:hasPublicationVenue      ?journal ;
                 bibo:authorList                ?authorList .
    ?journal     a bibo:Journal                ;
                 vivo:dateIssued               ?dateTimeValue .
    ?authorList  rdfs:member                   ?author .
    ?dateTimeValue vivo:dateTime                ?dateTime .
    ?author vivo:relatedBy ?position ;
            a foaf:Person .
    ?position    a          ?positionClass      ;
                 vivo:relates ?organization .
    ?organization a ?organizationClass ;
                 gn:locatedIn ?location .
    ?location <http://www.geonames.org/ontology#parentADM1> ?comunidadAutonoma .
}
GROUP BY ?comunidadAutonoma
```

Preguntas de competencia ASIO.

- Q11A - Encontrar el research object más antiguo organización:

```
SELECT DISTINCT ?researchObject ?date
WHERE {
    ?researchObject    a roh:ResearchObject ;
                      vivo:dateIssued ?dateIssued ;
                      bibo:authorList ?authorList .
    ?dateIssued        vivo:dateTime ?date .
    ?authorList        ?p ?author .
    ?author            a foaf:Person ;
                      roh:hasPosition ?position .
    ?position          vivo:relates <http://purl.org/roh/data#centro-investigacion-1> .
} ORDER BY ASC(?date) LIMIT 1
```

Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q11B - Encontrar el research object más reciente de una organización:

```
SELECT DISTINCT ?researchObject ?date
WHERE {
    ?researchObject    a roh:ResearchObject ;
                      vivo:dateIssued ?dateIssued ;
                      bibo:authorList ?authorList .
    ?dateIssued vivo:dateTime ?date .
    ?authorList ?p ?author .
    ?autor    a foaf:Person ;
              roh:hasPosition ?position .
    ?position vivo:relates <http://purl.org/roh/data#centro-investigacion-1> .
} ORDER BY DESC(?date) LIMIT 1
```

Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q15 - Listar proyectos con el mismo subject area o con subject areas relacionadas por parentesco, mirando en el árbol UNESKOS:

```
SELECT DISTINCT ?project
WHERE {
    {
        ?project a vivo:Project ;
            roh:hasKnowledgeArea ?knowledgeArea .
        ?knowledgeArea skos:broader+|skos:narrower+|skos:related+ uneskos:1203 .
    }
    UNION
    {
        uneskos:1203 skos:broader+ ?topKnowledgeArea .
        ?topKnowledgeArea skos:narrower+ ?otherKnowledgeArea .
        ?project roh:hasKnowledgeArea ?otherKnowledgeArea ;
            a vivo:Project .
    }
}
```


Una manera de hacer Europa

Preguntas de competencia ASIO.

- Q36 - Listar los grupos ordenados por financiación recibida:

```
SELECT ?organization ?fundingProgram (SUM(?monetaryAmount) as ?totalFunding)
WHERE {
    ?fundingProgram      a roh:FundingProgram ;
                          roh:funds ?funding .
    ?funding ro:hasPart ?fundingAmount .
    ?fundingAmount       roh:grants ?organization ;
                          roh:monetaryAmount ?monetaryAmount .
} GROUP BY ?organization ?fundingProgram
```

Razonamiento

- Principales servicios de un razonador:
 - Comprobar si una clase es subclase de otra.
 - Comprobar la consistencia de la ontología, si en base a los axiomas es posible que se pueda declarar una instancia para cada una de las clases.
 - Una clase será inconsistente si no se puede declarar ninguna clase.

Razonamiento

- Diferentes razonadores OWL existentes en el mercado:
 - Apache Jena
 - Hermit
 - ELK
 - Fact++
 - Pellet



FONDO EUROPEO DE DESARROLLO REGIONAL
(FEDER)

Una manera de hacer Europa

Taller



HĒRCULES



Una manera de hacer Europa

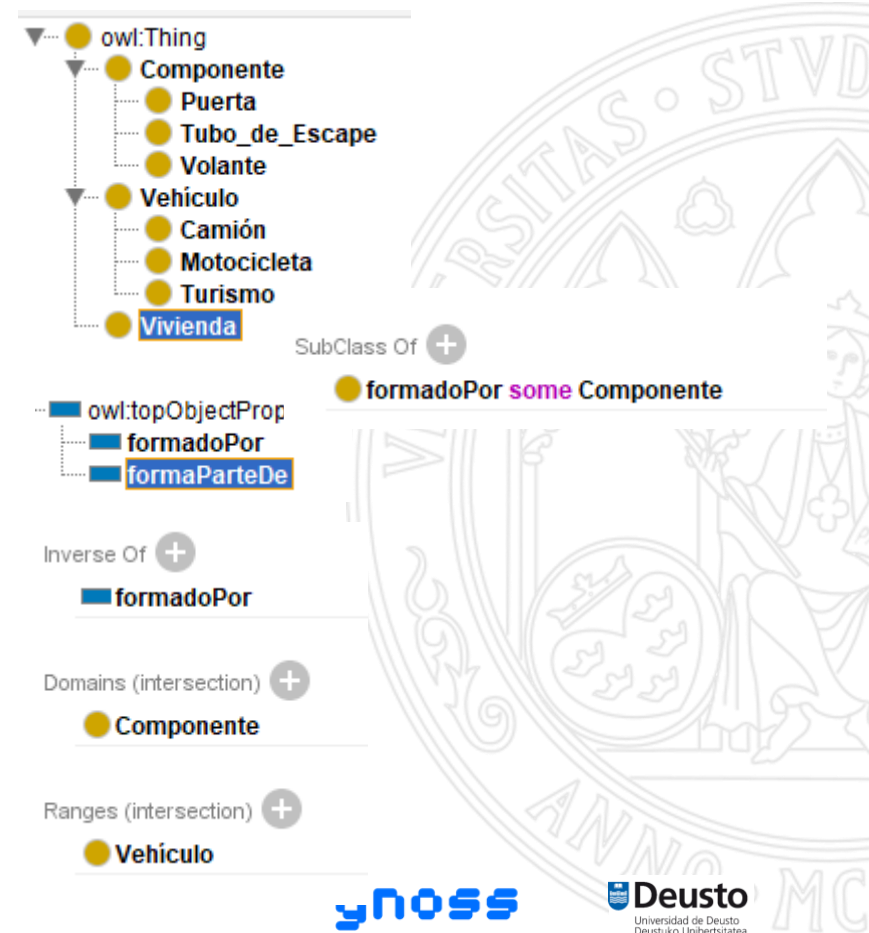
Taller

1. Desarrollo de una Ontología con Protégé y del flujo de CI/CD con GitHub Actions
 - Instalación de Protégé y desarrollo de la ontología
 - Creación del test unitario
 - Creación de un repositorio en GitHub y publicar el código
 - Desarrollo del flujo CI/CD
2. Realizar una modificación sobre la ROH e integrarla en el repositorio principal
 - Crear un 'fork' del repositorio [HerculesCRUE/GnossDeustoOnto](#)
 - Aplicar las modificaciones propuestas
 - Crear pull-request sobre el repositorio principal

Una manera de hacer Europa

1- Desarrollo de una Ontología con Protégé y del flujo de CI/CD con GitHub Actions

- Descargar e instalar Protégé:
<https://protege.stanford.edu/products.php#desktop-protege>
- Replicar la siguiente estructura de clases mostrada
- Crear las *object properties* mostradas y establecerlas como inversas
- Ejecutar el razonador y comprobar las inconsistencias.
- Corregir el error
- Pasar la ontología por OOPS y corregir errores (si los hubiera).



Una manera de hacer Europa

1- Desarrollo de una Ontología con Protégé y del flujo de CI/CD con GitHub Actions

- Instalar git o alguno de sus clientes.
- Crear un repositorio en [GitHub](#)
- Subir el código a la rama *main*.
- Implementar el test y el action cogiendo como ejemplo el repositorio [memaldi/ejemplo-ontologia](#)
- Crear una nueva rama llamada *new-feature*.
- Modificar la ontología de manera que se produzca una inconsistencia.
- Subir el código a GitHub e intentar ejecutar un pull-request a main.

Una manera de hacer Europa

2 - Realizar una modificación sobre la ROH e integrarla en el repositorio principal

- Crear un 'fork' del repositorio [HerculesCRUE/GnossDeustoOnto](#)
- Clonar el nuevo repositorio en local (git clone <URL-repo>).
- Crear una nueva *branch* para realizar las modificaciones (git checkout -b <nombre rama>).
- Añadir una modificación que cree una inconsistencia en la ontología (P. ej. Establecer dos clases disjuntas como equivalentes).
- Subir el código a la nueva rama del repositorio, y crear un pull-request hacia el repositorio original ([HerculesCRUE/GnossDeustoOnto](#))