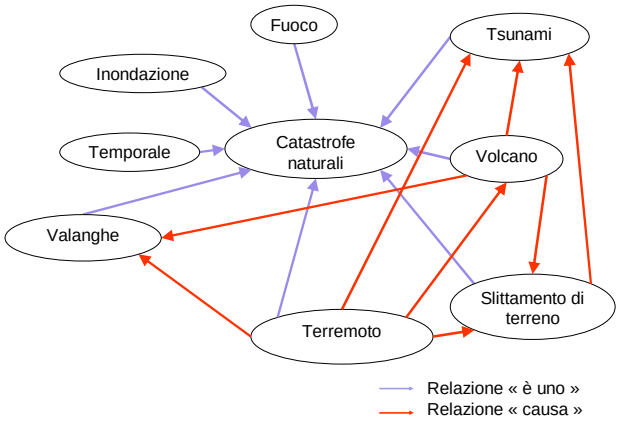
Capitolo 4°

ONTOLOGIE PER LE APPLICAZIONI GEOGRAFICHE

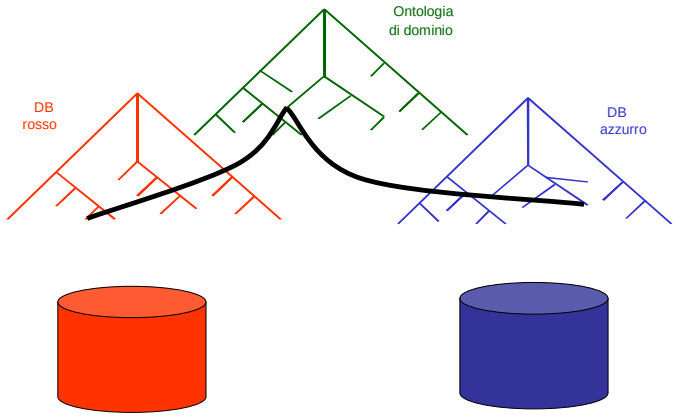
Esempio d'ontologia



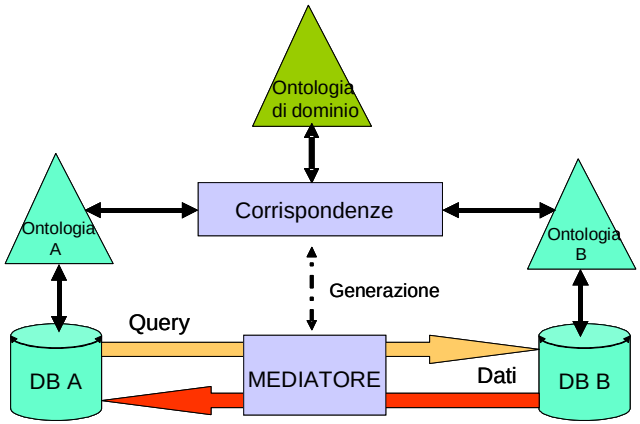
Ontologie geografiche

- 4.1 – Introduzione
- 4.2 – Basi teoriche delle ontologie spaziali
- 4.3 – Ingegnerizzazione
- 4.4 – Progetto TOWNTOLOGY
- 4.5 – OWL e Protégé
- 4.6 – Conclusioni

Interoperabilità attraverso un'ontologia



Corrispondenza con mediatori



4.1 – Introduzione

- Οντος = L'essere ; Λογια = discorso
- **Aristotele:** "lo studio dell'essere in quanto essere"
- **Def1**: teoria degli oggetti e delle loro relazioni
- **Def2**: teoria delle entità, specialmente delle entità che esistono nel linguaggio
- **Def3**: specificazione esplicita di una concettualizzazione (Gruber)

Differenze

- Ontologia (“o” maiuscola):
  - una disciplina filosofica
- Un'ontologia (“o” minuscola):
  - un artefatto inventato per descrivere il significato del vocabolario

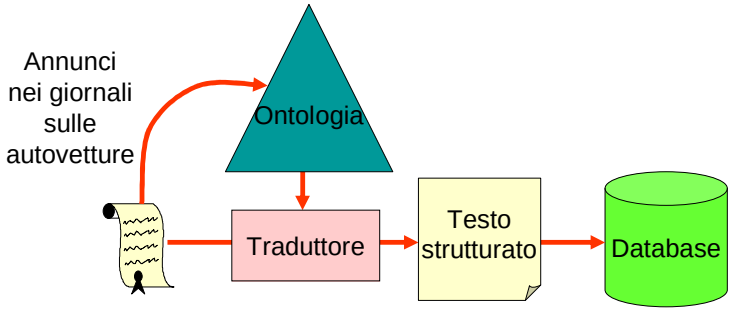
Definizione di Guarino

- **Nicola Guarino** : *"in IA, un'ontologia rappresenta un artefatto d'ingegneria costituta da un vocabolario utilizzato per costruire una realtà, accompagnata da un insieme d'ipotesi implicite concernente il significato delle parole e del vocabolario"*

Un'ontologia non è solo:

- un catalogo del mondo, una tassonomia, una lista d'oggetti
- un'ontologia non è riducibile a un'analisi puramente cognitiva, è piuttosto il lato oggettivo delle cose
- Non è proibita l'esistenza di differenti ontologie per descrivere la stessa problematica

Esempio di uso ed utilizzazione d'ontologia



Classificazioni differenti (Kavouras)

Ontology	Category_type
CORINE Land Cover	Peat bog
	Water course
	Water body
MEGRIN	Bog
	Canal
	Lake/ pond
	Salt marsh
	Salt pan
	Watercourse
WordNet	Body of water
	Bog
	Canal
	Lake
	Pond
	Salt pan
	Watercourse
	Watercourse

Ontologia = Concettualizzazione

- **Idea di base** : sostituire il dominio dell'interpretazione semantica (=concettualizzazione) con una base d'ontologie
- ➔ ridefinire un'ontologia come un ente matematico

Ontologia = Concettualizzazione

- Immensa descrizione in estensione, poche regole
- Fornitura di tutti i fatti possibili e plausibili
- Organizzazione da domini, contesti, applicazioni
- Dove trovare la lista dei termini?
- Esiste un'autorità per descrivere ad esempio una sedia?

Ontologia di dominio o d'applicazione

- Costruire un'ontologia è simile alla modellazione concettuale dei dati
- Al livello applicazione/dominio, un'ontologia puo' includere vincoli, regole di gestione, regole derivate, ecc.
- Nessuna considerazione di memorizzazione

Concetti

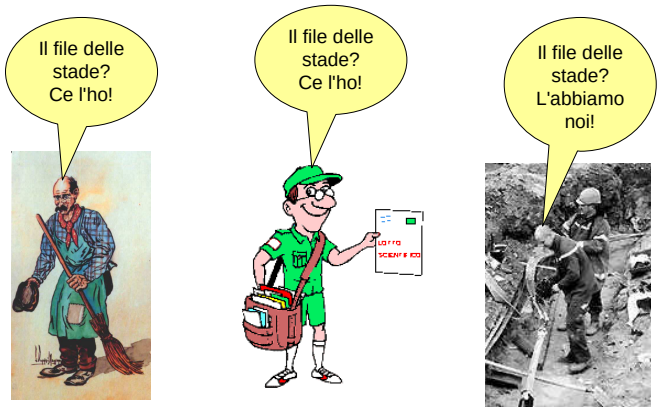
- Distinguere termini e concetti
- Al livello matematico :

Ontologia = grafo tra concetti
= rete semantica

Esempi sulle strade

- Distanza (km o miglie) ➔ sintattico
- Strade e autostrade ➔semantico

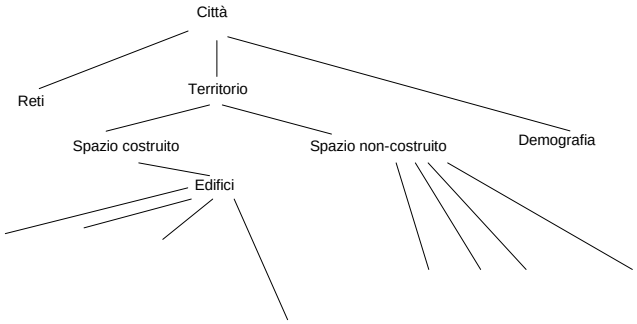
Abbiamo il file di strade!



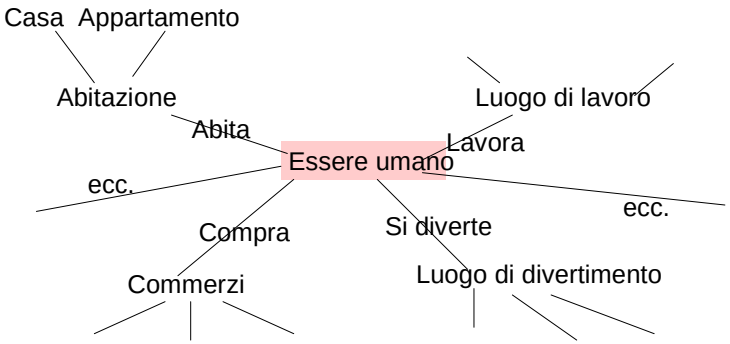
Abbiamo il file di strade!

	Netturbini	Postini	Ditta del gas
Strade private	No	Si	??
Strade municipali	Si	Si	Generalmente si
Strade con gas	?	?	Si
Strade senza gas	?	?	No
	234	251	241

Inizio d'ontologia urbana



Ontologia (seguito)



MicroKosmos

Linguaggi informatici

- KIF
- Derivati da XML
  - SHOE
  - XOL
  - RDF e RDF(S)
  - OIL
  - DAML+OIL
  - **OWL**

Linguaggi d'ontologia

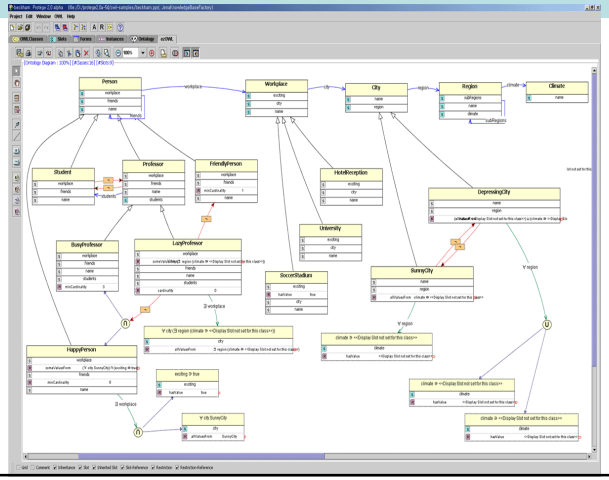
	KIF/OKBC/ CG/CycL	UML	Topic Maps / XTM	RDF(S)	DAML + OIL	OWL
Description	*Legacy KR Languages*	*Universal Modeling Language*	*Topic Maps/XML Topic Maps*	*Resource Description Framework*	*DARPA ML + Ontology Inference*	*Web Ontology Language*
Governance	ANSI /Other	OMG	ISO	W3C	DARPA	W3C
Years since proposed	*>5*	*>5*	*>5*	*>3*	*3 or less*	*3 or less*
Commercial Support (as a KRL)	□	□	□□	□□□□	□□□	□
Open Source Support	*Yes*	*Yes*	*Yes*	*Yes*	*Yes*	*Coming*

□ 2 or less vendors □□□ 10 or less vendors
□□ 5 or less vendors □□□□ > 10 vendors

Utilizzo

WEB	Ontologie
Informazioni in diversi formati	La modellazione, potrebbe facilitare il processo di ricerca su fonti eterogenee.
Mancanza di una struttura unica	Sono XML-based, quindi consentono la descrizione dei contenuti in maniera strutturata.

Ontologia con OWL



4.2 – Basi teoriche delle ontologie spaziali

- Oggetti spaziali
  - classi
  - descrizione
- Relazioni spaziali
  - topologiche
  - direzionali
  - distanza
  - mereologiche

<geo:Continent rdf:ID="SouthAmerica">

Oggetti e relazioni spaziali

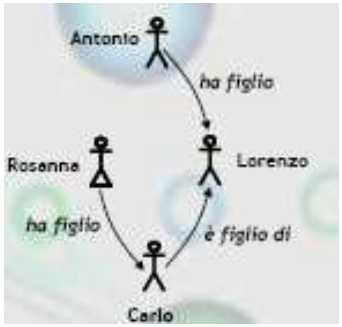
- Oggetti geografici
- Relazioni classiche (Egenhofer, ecc)

Concetti

- Primitivi
  - essere umano
  - Maschio
  - Femmina
- Derivati tramite restrizione
  - un uomo è un essere umano ed è maschio
  - una donna è un essere umano ed è femmina
  - una madre è una donna che ha_figlio almeno un essere umano
  - un padre è un uomo che ha_figlio almeno un essere umano
  - un genitore è o un padre o una madre
  - Un nonno è un uomo che ha_figlio un genitore

Relazioni, Assersioni, Fatti

- Relazioni
  - Primitive
    - ha_figlio
  - Derivate
    - è figlio di è l'inversa di ha figlio
- Asserzioni
  - Antonio, Lorenzo e Carlo sono uomini
  - Rosanna è una donna
- Fatti
  - Antonio ha_figlio Lorenzo
  - Rosanna ha_figlio Carlo
  - Carlo è_figlio_di Lorenzo



Conseguenze

- Una macchina in grado di capire un linguaggio ontologico sa inferire restrizioni, fatti e asserzioni
- Restrizioni:
  - un nonno è un genitore
- Fatti
  - Lorenzo ha_figlio Carlo
- Asserzioni
  - Antonio è un nonno
  - Lorenzo è un padre
  - Rosanna è una madre

Uso di un linguaggio ontologico

- Concetti
  - astrazioni del dominio applicativo
  - tipicamente visti come insiemi
- Relazioni
  - esprimono l'esistenza di relazioni tra i concetti del dominio tipicamente viste come relazioni binarie tra gli individui
- Assiomi/restrizioni:
  - formalizzano quali combinazioni di concetti e relazioni sono ammissibili
- Individui elementi degli insiemi definiti dai concetti
- Asserzioni didichiarano l'appartenenza di un individuo ad un insieme
- Fatti legano due individui tramite una relazione

Concetto

- Un concetto, noto anche come classe, può rappresentare da un punto di vista estensionale, un insieme di oggetti), o da un punto di vista intensionale, un'idea
- Si distingue tra concetti primitivi di cui non si dà nessuna definizione, e concetti derivati dai concetti primitivi
- Esempio sulle parentele
- Concetti primitivi:
  - Class(essere_umano)
  - Class(maschio)
- Concetti derivati:
  - Class(Uomo intersectionOf(essere_umano machio))

Relazione

- Una relazione, nota anche come proprietà, rappresentada un punto di vista estensionale, un insieme di coppie di oggetti, da un punto di vista intensionale, un legame tra concetti
- Si distingue tra relazioni primitive e relazioni derivate, come per gli oggetti;ma anche tra relazioni che legano tra loro concetti, e relazioni che legano un concetto ad un tipo primitivo (stringa, intero, data, etc.)
- Esempio sulle parentele
- Relazione primitiva:
  - ObjectProperty(ha_figlio)
- Relazione derivata:
  - ObjectProperty(è_figlio_di inverseOf(ha_figlio))
  - Data propertyDatatypeProperty(nato_il range(xsd:date))

Assiomi

- Assiomi servono per caratterizzare concetti e relazioni
- Esempi di assiomi sono
  - Definizioni di concetti per intersezione un uomo è un essere umano ed è maschio
    - Class(Uomo intersectionOf(essere_umano machio))
  - Definizioni di concetti per intersezione congiunzione
    - un genitore è o un padre o una madre
    - Class(genitore unionOf(padre madre))
    - Definizioni di relazioni
    - ha_figlio è l'inversa di è_figlio_di ObjectProperty(è_figlio_di inverseOf(ha_figlio)

Restrizioni

- Anche le restrizioni servono per caratterizzare concetti e relazioni
- Esempi di restrizioni sono
- Definizioni di concetti per restrizione
  - Un padre ha almeno un figlio
  - Class(padre restriction(pp:ha_figlio minCardinality(1))
- Specifica delle proprietà algebriche di una relazione
- la relazione discende_da è transitiva ObjectProperty(discende_da transitive)

Asserzione

- Un'asserzione rappresenta l'esistenza di un individua che appartiene all'insieme definito da un certo concetto o l'assegnamento di un tipo ad un individuo
- Esempio sulle parentele
  - Antonio è un uomoIndividual(Antonio type(uomo)
- Un fatto rappresenta l'esistenza di una relazione tra due individui o l'assegnazione di un valore ad una datatypeProperty
- Esempio sulle parentele
  - Antonio ha figlio Lorenzo
  - Individual(Antonio value(ha_figlio Lorenzo))

4.3 – Ingegnerizzazione

- Come costruire un'ontologia?
- Approcci possibili
  - Top-down
  - Bottom-up
  - Dizionari
  - Ecc.
- Come trovare il consenso tra gli attori?
- Come verificare il contenuto?

Progettazione collaborativa

- Interoperabilità dei sistemi
- Uso d'ontologie
- Ontologia = vocabolario
- = rete semantica

- Progetto *Towntology* per una ontologia urbana

Approccio top-down

- Definire i concetti di più alto livello
- Creare la rete semantica dall'alto
- Aggiungere i concetti specifici partendo dai concetti più generali.
- In alcuni casi: difficoltà di integrare gli oggetti reali

Approccio bottom-up

- Partire dagli oggetti comuni
- Aggregarli in oggetti più generali
- Poco a poco costruire concetti più generali

Consenso

- Due attori hanno due visioni differenti del mondo
- « *Ad ognuno la sua verità* »
- Risolvere i conflitti
- Ci sono due definizioni dello stesso concetto
  - definire due concetti differenti
  - conservare le due definizioni

4.4 – Progetto TOWNTOLOGY

- Creazione di una ontologia per l'urbanistica
- Prima tappa a Lione (2002-2003)
  - Pianificazione stradaria (in francese)
  - ≅ 900 concetti
- Seconda tappa (2003-2004)
  - Impostazione di una rete COST
  - Estensione a altre lingue
  - Descrizione della mobilità
- Sito web:
  - http://lisi.insa-lyon.fr/~twonto

TOWNTOLOGY

- Difficoltà di dare le definizioni
- Com scegliere la buona definizione
- Quando ci sono due definizioni
  - Stesso oggetto o due oggetti?
- Ricercare il consensus sulle definizioni
- Obiettivo di Towntology
  - Strumento per raccogliere le varie definizioni degli oggetti urbani
  - Pre-ontologia / ontologia prima del consensus

Principi di Towntology

- Presentazione visuale
- Rete semantica
- Struttura d'ipertesto
- Definizioni multiple
- Origine delle definizioni
- Possibilità d'aggiornamento
- Foto e disegni

- 9 relazioni:
  - *è fatto di*
  - *è composto di*
  - *è localizzato a*
  - *è utilizzato per*
  - *è localizzato su*
  - *è un*
  - *è un sotto-insieme di*
  - *dipende da*
  - *è uno strumento per*

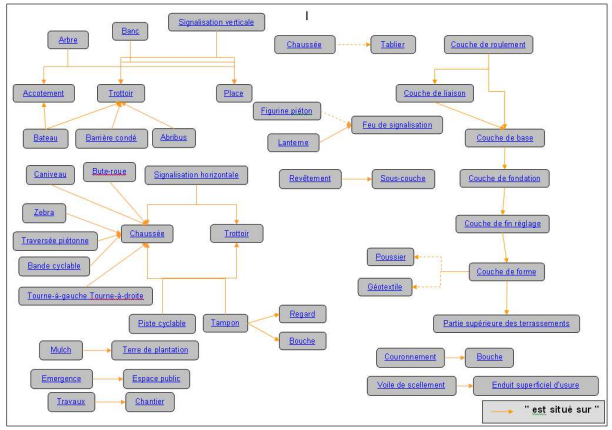
Progetto *Towntology*

- Progettare un'ontologia
  - città
  - urbanistica
- Lavoro attuale: INSA (LIRIS + EDU)
- Progetto COST
  - Université de Liège
  - Queens University of Belfast
  - Münster Universität
  - Universidad Politécnica de Madrid
  - Università della Basilicata

A Lione

- Pianificazione delle strade
  - Attualmente >800 concetti (in francese)
  - Grafo, relazioni, definizioni, foto, ecc.
  - Solo in francese

TOWNTOLOGY

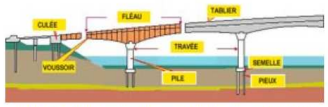


Esempio : Tablier (=piano stradale)

TABLIER
Plate-forme horizontale qui supporte la chaussée sur un pont.

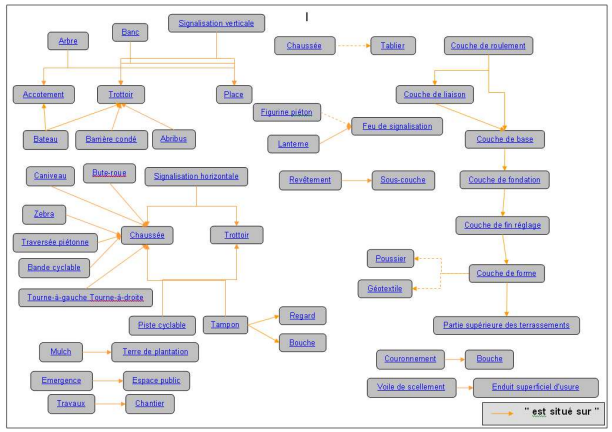
Dictionnaire de la voirie

Source : http://www.saone-et-loire.equipement.gouv.fr/IRI/RBc/images%20RBc/RBcc02a.jpg



Source : http://www.gironde.equipement.gouv.fr/pont-aquitaine/photos/glossaire.jpg

Grafo dell'ontologia



Pista ciclabile

PISTE CYCLABLE
Aménagement en site propre séparé de la circulation générale automobile. Elle peut être située sur un trottoir, sur une chaussée toutefois séparée physiquement de la circulation automobile par une bordure, une bande de stationnement V.P., ...
La largeur de l'aménagement est de 1,50 mètre pour une piste unidirectionnelle, et de 3,00 mètres pour une piste bidirectionnelle. Le revêtement peut être un enrobé, un asphalte, un béton désactivé ou un sable stabilisé.
Ce type d'aménagement s'adresse à un usage ludique ou familial et permet aux usagers de se rendre vers des zones de loisirs. Il est recommandé d'aménager les pistes cyclables le long des voies où il y a peu d'intersections, les voies à vitesse élevée et à grand débit, les anciennes voies ferrées, les chemins de halage.

Dictionnaire de la voirie

Source : Dictionnaire de la voirie

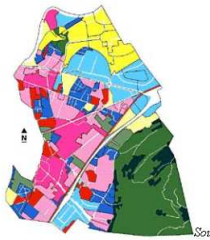
Source : http://membres.lycos.fr/leroyaccidentiprate.jpg

Source : http://www.rail.lu/images/riggingeologiste.jpg

Esempio: piano regolatore

PLAN D'OCCUPATION DES SOLS (P.O.S.)
Document d'urbanisme opposable aux tiers qui fixe les règles générales et les servitudes d'utilisation des sols. Composé de documents graphiques qui localisent des zones, d'un règlement qui fixe le droit des sols applicable à chaque zone et d'annexes techniques.

Dictionnaire de la voirie



Source : http://www.ville-st-martin-dheres.fr/images/photos/pos.gif

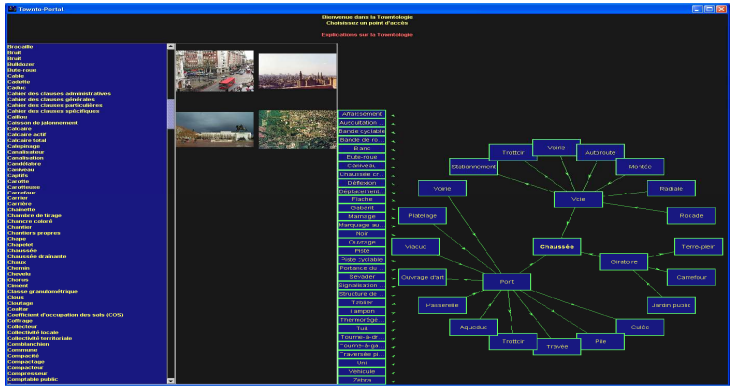
Inizio di un'ontologia urbana

```
<ONTOLOGY>
  <HEAD>
    <TITLE>Transports</TITLE>
    <LANGUAGE>français</LANGUAGE>
    <CUSTODIAN>Christophe BERTHET</CUSTODIAN>
    <LAST_MODIF_DATE>2004/6/25</LAST_MODIF_DATE>
  </HEAD>
  <BODY>
    <RELATION_TYPES>
      <RELATION_TYPE ID="100001" ORIGINATOR="Christophe BERTHET" INSERTION_DATE="2004/06/21">
        <RELATION_NAME>dépend de</RELATION_NAME>
        <TERMS />
        <RELATION_DEF />
        <RELATION_PROPERTIES SYMMETRIC="false" TRANSITIVE="false" MAYBEOPTIONAL="false" />
      </RELATION_TYPE>

        .........

    </BODY>
</ONTOLOGY>
```

Descrizione di un concetto

```
<CONCEPT_NAME>Accident de la route </CONCEPT_NAME>
<TERMS />
<CONCEPT_DOMAIN ID="200001" />
<CONCEPT_DEFS>
  <CONCEPT_DEF ORIGINATOR="Christophe BERTHET" INSERTION_DATE="2004/06/21">
    <CONCEPT_DEF_SOURCE>
      <AUTHORS />
      <REF>Glossaires – Promotion Of Results in Transport Research and Learning</REF>
    </CONCEPT_DEF_SOURCE>
    <CONCEPT_DEF_TEXT>Définition utilisée pour les statistiques dans la plupart des pays : il
      s'agit d'une collision ayant lieu sur la voie publique et qui implique au moins un véhicule
      roulant. Sont considérés comme accidents de la route les accidents provoquant
      uniquement des dégâts matériels et les accidents occasionnant des
      blessures.</CONCEPT_DEF_TEXT>
  </CONCEPT_DEF>
</CONCEPT_DEFS>
<MULTIMEDIA />
</CONCEPT>
```

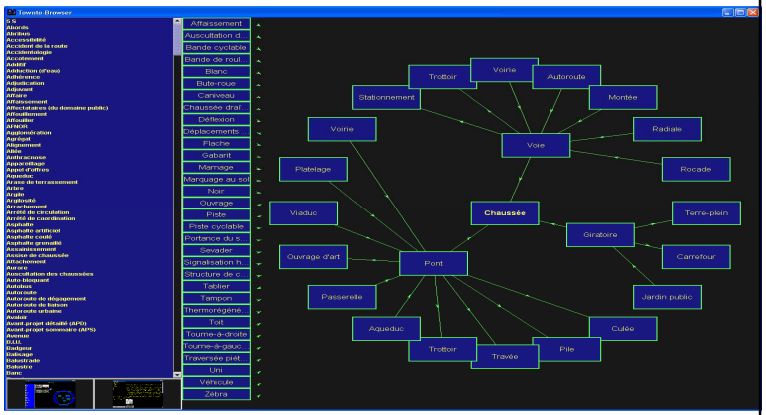
Portale



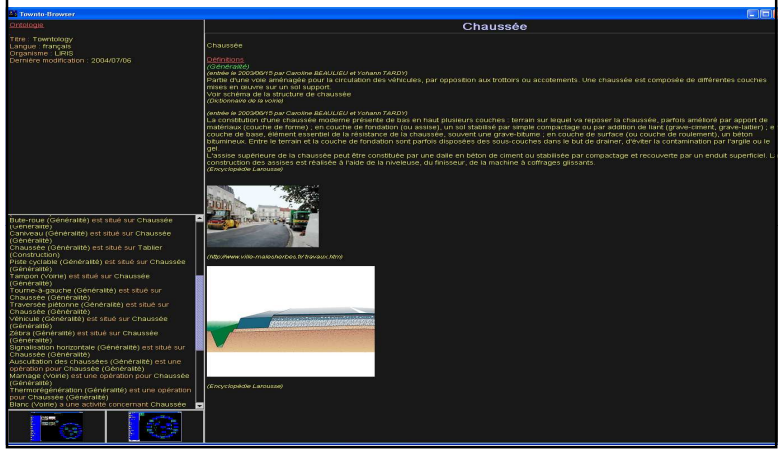
Sistema visuale

- Browser
- Editore d'ontologia
- Editore d'immagini

Interfaccia grafica (browser)



Visualizzazione del contenuto

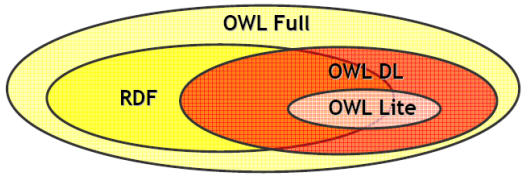


4.5 – OWL e Protégé

- Dal punto di visto prattico, un'ontologia è
  - una descrizione formale di concetti in un dominio (classi)
  - le proprietà di ciascun concetto (slot)
  - le restrizioni sugli slot (facets)
- ONTOLOGIA + UN INSIEME DI ISTANZE DI CLASSI = BASE DI CONOSCENZA

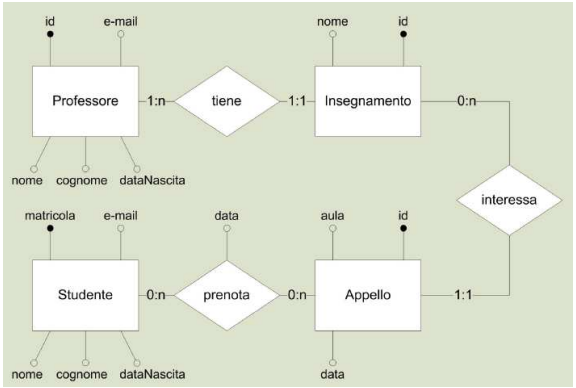
OWL

- RDF: Resource Description Framework
  - Rappresentazione delle rete semantiche
  - (nodo1, concetto, nodo2)
- RDFS: Resource Description Framework Schema
  - Risorse, proprietà, potto-classi, superclassi, istanziazione, ereditarietà,restrizioni sulle proprietà
- OWL: Ontology Web Language
  - Ontologie più complicate
  - Cardinalità delle proprietà,
  - Classi come unione o intersezione di altre classi



- **OWL Full:**
  - concepito per gli utenti che desiderano la *massima espressività* e libertà sintattica di *RDF senza* richiedere *garanzie* computazionali.
- **OWL DL (Description Logics):**
  - include tutti i costrutti di OWL ma pone *alcune restrizioni* sul loro utilizzo per garantire la *decidibilità* e la *trattabilità* computazionale
- **OWL Lite:**
  - concepito per fornire agli utenti una gerarchia di classificazione e vincoli semplici, un *sottoinsieme funzionale da utilizzare per semplici implementazioni*

Esempio: Università



Definizione delle Classi: Appello

```
<owl:Class rdf:ID="Appello">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haAula" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haDataAppello" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#interessa" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Definizione delle Classi: Insegnamento

```
<owl:Class rdf:ID="Insegnamento">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haNome" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#tenuto" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Definizione delle classi:Persona

```
<owl:Class rdf:ID="Persona">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haNome" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haCognome" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Definizione delle Classi: Prenotazione

```
<owl:Class rdf:ID="Prenotazione">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haDataPrenotazione" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haAppello" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haStudente" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Definizione delle Classi: Professore e Studente

```
<owl:Class rdf:ID="Professore">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#tiene" />
        <owl:minCardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:minCardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>

  <owl:Class rdf:ID="Studente">
    <rdfs:subClassOf>
      <owl:Restriction>
        <owl:onProperty rdf:resource="#haMatricola" />
        <owl:cardinality rdf:datatype="&xsd;nonNegativeInteger">1</owl:cardinality>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:Class>
```

Sussunzione

```
<owl:Class rdf:about="#Professore">
      <rdfs:subClassOf rdf:resource="#Persona" />
  </owl:Class>

  <owl:Class rdf:about="#Studente">
    <rdfs:subClassOf rdf:resource="#Persona" />
  </owl:Class>
```

Proprietà

```
<owl:DatatypeProperty rdf:ID="haNome">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="&owl;Thing" />
      <rdfs:range rdf:resource="&xsd;string" />
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="haCognome">
    <rdf:type rdf:resource="&owl;FunctionalProperty" />
    <rdfs:domain rdf:resource="&owl;Persona" />
    <rdfs:range rdf:resource="&xsd;string" />
  </owl:DatatypeProperty>

  <owl:DatatypeProperty rdf:ID="haDataDiNascita">
    <rdf:type rdf:resource="&owl;FunctionalProperty" />
    <rdfs:domain rdf:resource="&owl;Persona" />
    <rdfs:range rdf:resource="&xsd;data" />
  </owl:DatatypeProperty>
```

Proprietà

```
<owl:DatatypeProperty rdf:ID="haMatricola">
   <rdf:type rdf:resource="&owl;FunctionalProperty" />
   <rdfs:domain rdf:resource="&owl;Studente" />
   <rdfs:range rdf:resource="&xsd;nonNegativeInteger" />
</owl:DatatypeProperty>

<owl:ObjectProperty rdf:ID="tiene">
   <rdf:type rdf:resource="&owl;InverseFunctionalProperty" />
   <rdfs:domain rdf:resource="#Professore" />
   <rdfs:range rdf:resource="#Insegnamento" />
</owl:ObjectProperty>

<owl:ObjectProperty rdf:ID="tenuto">
   <owl:inverseOf rdf:resource="#tiene" />
</owl:ObjectProperty>
```

Proprietà

```
<owl:DatatypeProperty rdf:ID="haAula">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="&owl;Appello" />
      <rdfs:range rdf:resource="&xsd;string" />
   </owl:DatatypeProperty>

   <owl:DatatypeProperty rdf:ID="haDataAppello">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="&owl;Appello" />
      <rdfs:range rdf:resource="&xsd;data" />
   </owl:DatatypeProperty>

   <owl:ObjectProperty rdf:ID="interessa">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="#Appello" />
      <rdfs:range rdf:resource="#Insegnamento" />
   </owl:ObjectProperty>
```

Proprietà

```
<owl:ObjectProperty rdf:ID="haAppello">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="#Prenotazione" />
      <rdfs:range rdf:resource="#Appello" />
   </owl:ObjectProperty>

   <owl:ObjectProperty rdf:ID="haStudente">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="#Prenotazione" />
      <rdfs:range rdf:resource="#Studente" />
   </owl:ObjectProperty>

   <owl:DatatypeProperty rdf:ID="haDataPrenotazione">
      <rdf:type rdf:resource="&owl;FunctionalProperty" />
      <rdfs:domain rdf:resource="&owl;Prenotazione" />
      <rdfs:range rdf:resource="&xsd;data" />
   </owl:DatatypeProperty>
```

Protégé: un editor per la creazione di ontologie

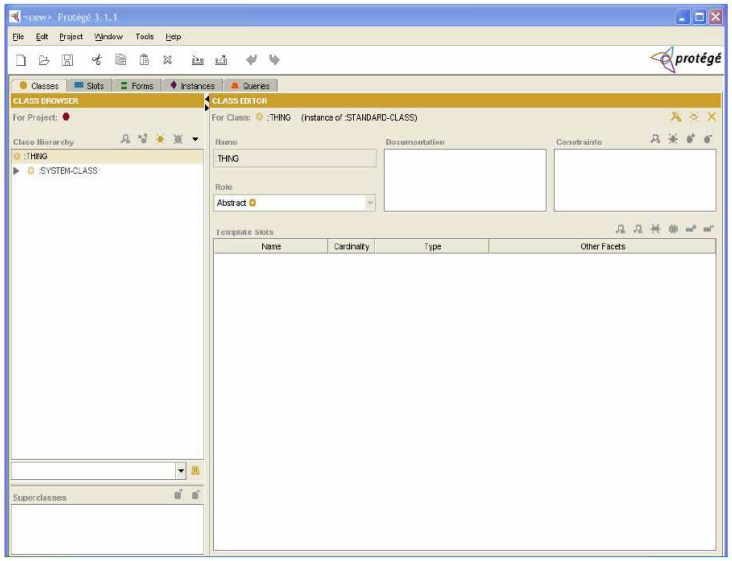
- Caratteristiche:
  - Piattaforma open-source
  - Può esportare le ontologie in vari formati: RDF(S), XML Schema e OWL
  - E' basato su Java
  - E' estendibile (esistono numerose API e plug-in)
  - Dispone di numerosi ambienti plug-and-play che consentono un rapido sviluppo delle applicazioni
  - Sviluppato dall'università di Stanford
  - (http://smi.stanford.edu/projects/protege/)

Protégé API

- Protégé fornisce un'interfaccia che altre applicazioni possono utilizzare per accedere alle knowledge bases. Queste applicazioni non necessitano di usare o visualizzare nessuna delle interfacce utente di Protégé.
- Esiste una classe che fornisce il metodo getKnowledgeBase() il quale permette di accedere al contenuto della base di conoscenza. Tali contenuti possono essere usati ad esempio da motori di inferenza (es. Drools, Mandarax, Jess, …), i quali possono compiere le loro inferenze ed eventualmente aggiornare la base di conoscenza.
- Ulteriori informazioni relative alle API di Protégé possono essere reperite su
- http://protege.stanford.edu/doc/pdk/kb-api.html

La piattaforma Protégé

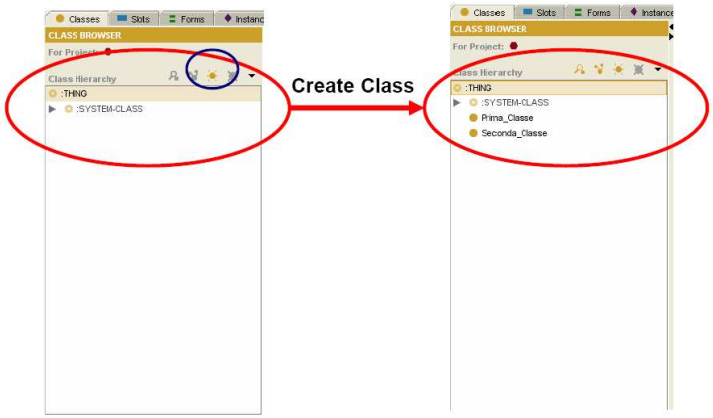
- Esistono due modalità per creare le ontologie in Protégé:
  - Il Protégé-Frames editor, consente di costruire e popolare le ontologie chesono basate su "frame", secondo il protocollo OKBC (Open Knowledge Base Connectivity protocol [www.ai.sri.com/okbc/]). In questo modello, un'ontologia è costituita da un'insieme di classi organizzate in gerarchia, rappresentanti un insieme di concetti. Le classi sono caratterizzate da slot e relazioni.
  - Il Protégé-OWL editor, consente di costruire ontologie per il Semantic Web, in particolare secondo il linguaggio OWL [www.w3.org/2004/OWL/]. Un'ontologia OWL può includere descrizioni di classi, di proprietà e le loro istanze.



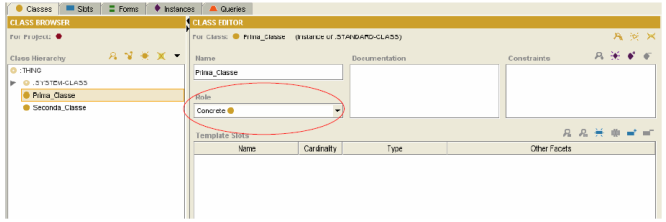
Progettazione di un'ontologia

- 1. Determinare il dominio e lo scopo dell'ontologia
- 2. Considerare la possibilità di riusare ontologie esistenti
- 3. Individuare i concetti chiave del fenomeno da descrivere
- 4. Organizzare i concetti in classi e gerarchie tra le classi
- 5. Definire le proprietà delle classi
- 6. Definire vincoli (valori leciti)sulle proprietà
- 7. Creare le istanze
- 8. Attribuire i valori alle proprietà per tutte le istanze create

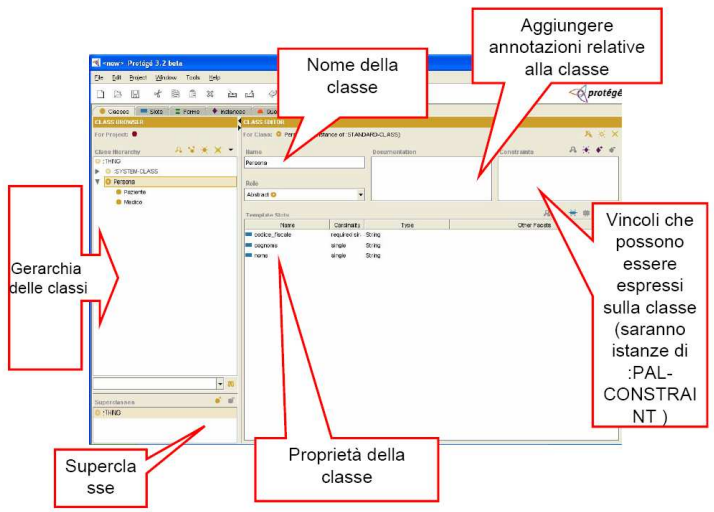
Protégé Classes Tab



Protégé Classes Tab (2)

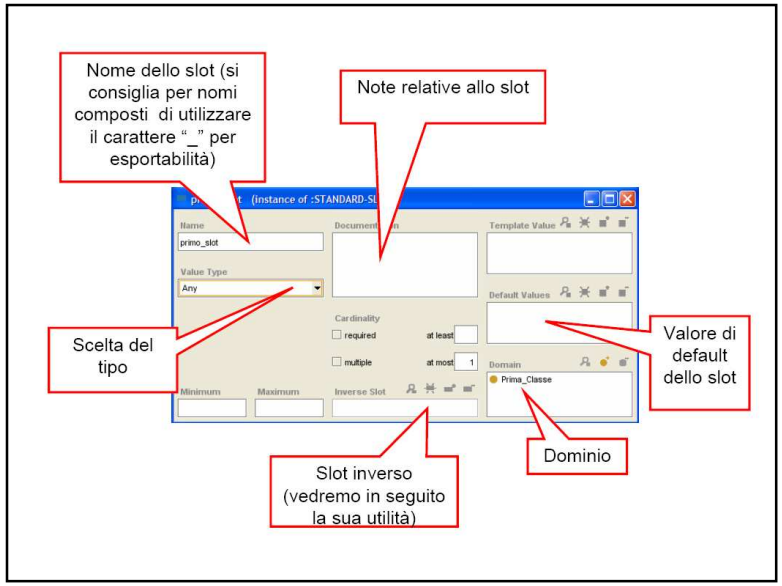


- All'interno di Protégé Classes Tab si può definire il "role" di una classe: Astratta o Concreta
- Una caratteristica importante è che Protégé supporta l'ereditarietà multipla.
- Inoltre si possono definire gli slot, i vincoli sui valori degli slot, le relazioni fra le classi e le proprietà.

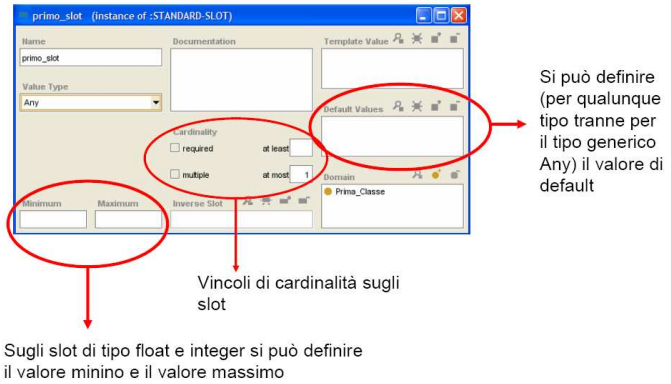


Le proprietà di una classe - slot

- Definite le classi, si devono descrivere le proprietà di queste, che nell'ontologia saranno gli slot. Gli slot possono rappresentare:
  - proprietà estrinseche (ad esempio il gusto di un vino)
  - proprietà intrinseche (ad esempio il nome)
  - parti di un oggetto, nel caso in cui questo sia strutturato (possono essere parti sia "astratte" sia "concrete")
  - le relazioni con altre classi
- Gli slot possono avere dei vincoli, ad esempio il tipo, il numero di valori (la cardinalità)…
- In Protégé i tipi previsti sono: boolean, float, integer, string, symbol (consente di enumerare i valori possibili), class, instance, any; inoltre la definizione dell'attributo può essere obbligatoria o meno (settando in maniera opportuna il flag "required") e uno slot può avere una cardinalità singola o multipla

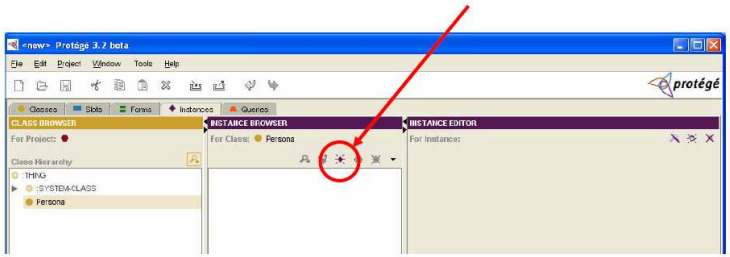


Le proprietà di una classe – slot (2)

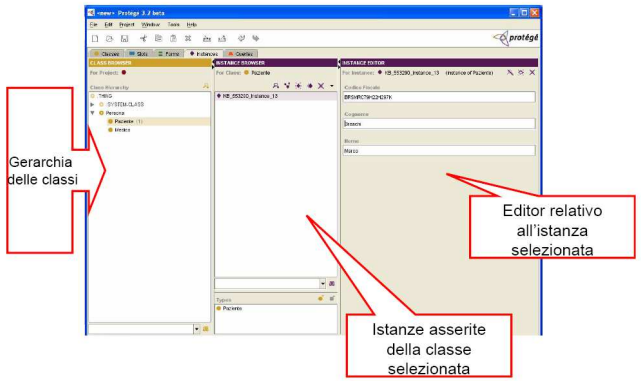


Creazione di un'istanza

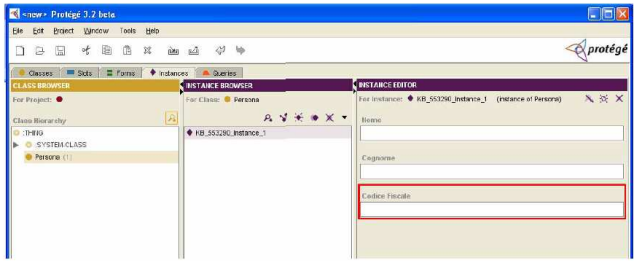
- Definire un'istanza significa:
  - 1) Scegliere una classe
  - 2) Creare un'istanza della classe
  - 3) Immettere i valori degli slot



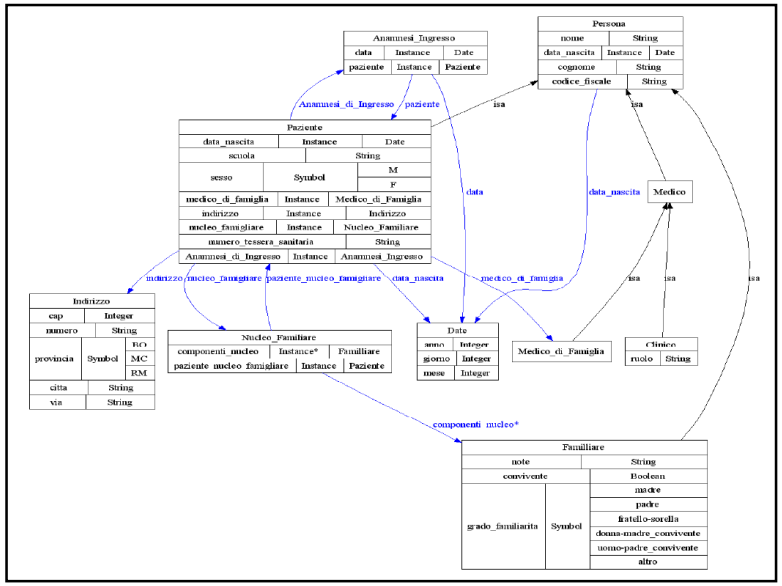
Protégé Instance Tab



Esempio di creazione di un'istanza



- Nota: Gli slot required vengono evidenziati da Protégé Instance Tab con un bordo rosso. La consistenza deve essere garantita dall'utente, in quanto Protégé consente la memorizzazione di istanze anche aventi campi required non compilati.
- Il numero di istanze presenti in una data classe viene visualizzato accanto al nome della classe.



4.6 – Conclusioni

- Importanza delle ontologie
  - organizzazione delle conoscenze
  - document retrieval
  - interoperabilità
- Bisogno di un'ontologia che copre tutto il campo del mondo reale
- Difficoltà d'ingegnerizzazione

Da GML a OWL

GML

```
<featureMember>
 <cntry02>
  <_SHAPE_>
   <MultiPolygon srsName="">
    <polygonMember>
     <Polygon>
      <outerBoundaryIs>
       <LinearRing>
        <coordinates>
         -171.84805297851562…
         9.218889236450195
        </coordinates>
       </LinearRing>
      </outerBoundaryIs>
     </Polygon>
    </polygonMember>
        ……
   </MultiPolygon>
  </_SHAPE_>
  <CNTRY_NAME>Tokelau</CNTRY_NAME>
 </cntry02>
</featureMember>
```

OWL

```
<geo:Country rdf:ID="Tokelau">
 <rdfs:label>Tokelau</rdfs:label>
 <geo:shape>
  <geo:MultiPolygon rdf:nodeID="TokelauShape">
   <geo:xyCoordinates>
    -171.84805297851562…
    9.218889236450195
   </geo:xyCoordinates>
      ……
  </geo:MultiPolygon>
 </geo:shape>
</geo:Country>
```

Portali d'Ontologie

- http://www.fb10.uni-bremen.de/anglistik/langpro/webspace/jb/info-pages/ontology/ontology-root.htm
- http://www.isi.edu/geoworlds/
- http://mcmcweb.er.usgs.gov/sdts/standard.html
- http://ontology.buffalo.edu/
- GML : http://www.opengis.net/gml/index.htm
- Beni culturali: http://eprints.ecs.soton.ac.uk/6147/
- http://esw.w3.org/topic/GeoInfo
- http://www.comp.leeds.ac.uk/brandon/cosit03ontology/
- http://mcmcweb.er.usgs.gov/sdts/SDTS_standard_oct91/part2.html.html