

Linked Open Data e Semantic Web: Fondamenti e Linguaggi di Interrogazione Parte Prima

Cristiano Longo
longo@dmf.unict.it

Università di Catania, 04/12/2015

Open Knowledge

SEE HOW DATA CAN CHANGE THE WORLD

A world where knowledge creates power for the many, not the few.

A world where data frees us — to make informed choices about how we live, what we buy and who gets our vote.

A world where information and insights are accessible — and apparent — to everyone.

This is the world we choose.

da Open Knowledge Foundation.¹

¹<https://okfn.org/>

The Open Definition

Open means anyone can freely access, use, modify, and share for any purpose (subject, at most, to requirements that preserve provenance and openness).²

²<http://opendefinition.org>

Formati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

a) *formati aperti*, un formato di dati reso pubblico, documentato esaustivamente e neutro rispetto agli strumenti tecnologici necessari per la fruizione dei dati stessi (esempi sono XML, JSON, ODF, ...);

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione
[...]

in *formati aperti*,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione [...]

in formati aperti,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione [...]

in *formati aperti*,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione [...]

in *formati aperti*,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione [...]

in *formati aperti*,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti

Dalle *LINEE GUIDA NAZIONALI PER LA VALORIZZAZIONE DEL PATRIMONIO INFORMATIVO PUBBLICO*:

b) *dati di tipo aperto*, i dati che presentano le seguenti caratteristiche:

1) sono disponibili secondo i termini di una *licenza* che ne permetta l'utilizzo da parte di chiunque, anche per finalità commerciali, in formato disaggregato (esempi di licenze sono Creative Commons e Apache License);

2) sono *accessibili* attraverso le tecnologie dell'informazione e della comunicazione
[...]

in *formati aperti*,

sono adatti all'utilizzo automatico da parte di programmi per elaboratori

e sono provvisti dei relativi *metadati*;

3) sono resi disponibili *gratuitamente* [...] oppure sono resi disponibili ai costi marginali sostenuti per la loro riproduzione e divulgazione.

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- **Economici**

- redazione di business plan (vedi l'articolo sui flussi turistici³),
- creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea)⁸;
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- Economici

- redazione di business plan (vedi l'articolo sui flussi turistici³),
- creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea)⁸;
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- Economici
 - redazione di business plan (vedi l'articolo sui flussi turistici³),
 - creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea)⁸;
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- Economici
 - redazione di business plan (vedi l'articolo sui flussi turistici³),
 - creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea),⁸
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- Economici
 - redazione di business plan (vedi l'articolo sui flussi turistici³),
 - creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea)⁸;
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA

L'apertura dei dati in possesso della Pubblica Amministrazione riveste particolare importanza. Alcune ricadute dell'apertura di questi dati:

- Economici
 - redazione di business plan (vedi l'articolo sui flussi turistici³),
 - creazione di nuove imprese basate sui dati (vedi ad esempio *Open Data 500*⁴);
- realizzazione di nuovi servizi per i cittadini (vedi ad esempio PETRUSINO,⁵ ConfiscatiBene,⁶ Ordnance Survey,⁷ Europea)⁸;
- trasparenza (vedi ad esempio soldipubblici.it oppure <http://parlamento17.openpolis.it/>);
- governo partecipato (uno strumento ad esempio è <http://www.normattiva.it/>).

³<http://opendatasicilia.it/2015/05/13/flussi-turistici-e-fruizione-dei-beni-culturali-catania-dal-2012-al-2013/>

⁴<http://www.opendata500.com/us/list/>

⁵<http://petrusino.opendatasicilia.it/>

⁶<http://www.confiscatibene.it/it>

⁷<https://www.ordnancesurvey.co.uk/innovate/showcase>

⁸<http://labs.europeana.eu/apps>

Dati Aperti della PA - Esempi

Alcune esempi di pubbliche amministrazioni che hanno pubblicato i dati:

- Centrali

- Camera (<http://dati.camera.it/>) e Senato (<http://dati.senato.it/>) della Repubblica Italiana,
- ISTAT (<http://dati.istat.it/>),
- Ministero per i Beni e le Attività Culturali e Turismo (<http://www.beniculturali.it/mibac/export/MiBAC/sito-MiBAC/MenuPrincipale/Trasparenza/Open-Data/index.html>, <http://www.beniculturali.it/mibac/opencms/MiBAC/sito-MiBAC/MenuPrincipale/LuoghiDellaCultura/Ricerca/index.html>);

- Regioni

- Emilia Romagna (<http://dati.emilia-romagna.it>),
- Lombardia (<https://www.dati.lombardia.it>),
- Piemonte (<http://www.dati.piemonte.it>),
- Puglia (www.dati.puglia.it),
- Toscana (<http://dati.toscana.it>),
- Umbria (<http://dati.umbria.it>);

- Provincie - Provincia Autonoma di Trento e Bolzano (<http://dati.trentino.it>);

- Comuni

- Catania (<http://opendata.comune.catania.gov.it>),
- Palermo (http://www.comune.palermo.it/opendata_menus.php),
- Matera (<http://dati.comune.matera.it>).

Dati Aperti della PA - Tipologie

Riportiamo a titolo esemplificativo l'elenco di categorie dei dataset presenti su <http://dati.emilia-romagna.it/> :

- Agricoltura
- Ambiente
- Bilancio
- Bilancio Consuntivo
- Bilancio di Previsione
- Cartografia
- Dati di Bilancio
- Edilizia
- Geologia
- Geomorfologia
- Mobilità
- Pianificazione agricola
- Processi Geologici
- Suolo
- Territorio
- Turismo
- amministrazione
- istruzione
- popolazione

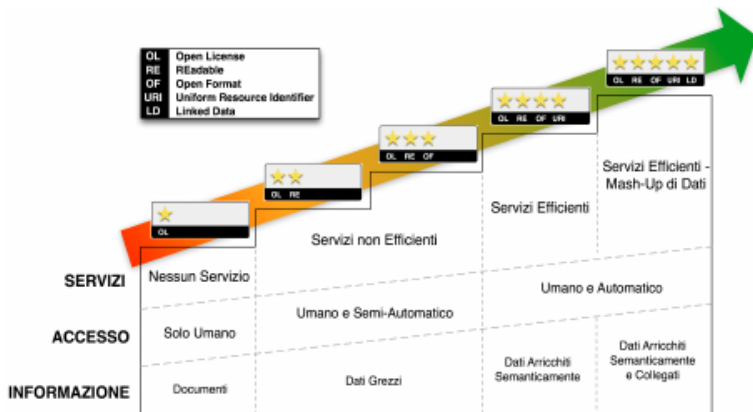
World Wide Web Consortium

Il *World Wide Web Consortium* (in breve *W3C*, vedi <http://www.w3.org>) è un consorzio di standardizzazione per il Web che conta 403 membri tra aziende e organizzazioni governative: CNR, Microsoft Corporation, Apple Inc., Intel Corporation, Facebook, Google Inc., . . .

Altri standard sviluppati in seno al W3C sono: URL, HTTP, XML, HTML, CSS, SOAP, WSDL, Javascript.

Classificazione 5 Stelle

Il w3c propone un modello per la qualità degli open data denominato *classificazione a 5 stelle*.⁹



⁹<http://5stardata.info>

Classificazione 5 Stelle - Dimensioni

La classificazione 5 stelle viene estesa dall'AgID con alcune *dimensioni* esplicative:

- **INFORMAZIONE** - descrive la qualità dell'informazione fornita insieme ai dati;
- **ACCESSO** - descrive la facilità con cui utenti e programmi riescono ad accedere ai dati;
- **SERVIZI** - riguarda le tipologie e l'*efficienza* dei servizi che possono essere realizzati a partire dai dati.

Classificazione 5 Stelle - Dimensioni

La classificazione 5 stelle viene estesa dall'AgID con alcune *dimensioni* esplicative:

- **INFORMAZIONE** - descrive la qualità dell'informazione fornita insieme ai dati;
- **ACCESSO** - descrive la facilità con cui utenti e programmi riescono ad accedere ai dati;
- **SERVIZI** - riguarda le tipologie e l'*efficienza* dei servizi che possono essere realizzati a partire dai dati.

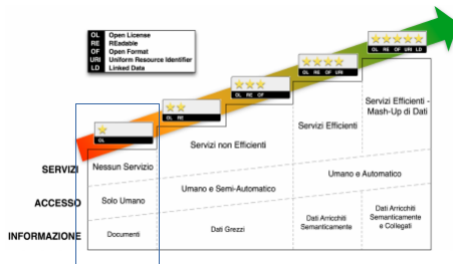
Classificazione 5 Stelle - Dimensioni

La classificazione 5 stelle viene estesa dall'AgID con alcune *dimensioni* esplicative:

- *INFORMAZIONE* - descrive la qualità dell'informazione fornita insieme ai dati;
- *ACCESSO* - descrive la facilità con cui utenti e programmi riescono ad accedere ai dati;
- *SERVIZI* - riguarda le tipologie e l'*efficienza* dei servizi che possono essere realizzati a partire dai dati.

Classificazione 5 Stelle - Una Stella

Una Stella: Open Licence

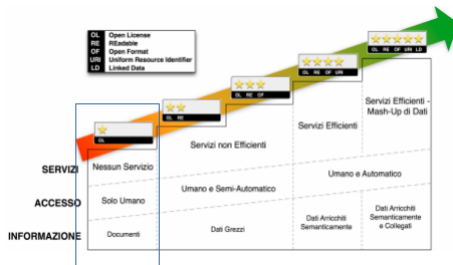


La *prima stella* si ottiene rilasciando i dati in qualunque formato ma con una *licenza aperta*. Rientrano in questa categoria ad esempio le scansioni dei documenti.

- **INFORMAZIONE:** *documenti* - i dati sono incorporati all'interno di documenti senza struttura;
- **ACCESSO:** *solo umano* - solo gli umani sono in grado di leggere i documenti senza struttura e quindi dare un senso ai dati in esso presenti;
- **SERVIZI:** *nessuno* può essere abilitato a meno di significativi interventi umani di estrazione ed elaborazione;

Classificazione 5 Stelle - Una Stella

Una Stella: Open Licence

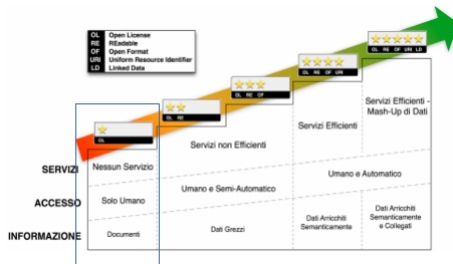


La *prima stella* si ottiene rilasciando i dati in qualunque formato ma con una *licenza aperta*. Rientrano in questa categoria ad esempio le scansioni dei documenti.

- **INFORMAZIONE:** *documenti* - i dati sono incorporati all'interno di documenti senza struttura;
- **ACCESSO:** *solo umano* - solo gli umani sono in grado di leggere i documenti senza struttura e quindi dare un senso ai dati in esso presenti;
- **SERVIZI:** *nessuno* può essere abilitato a meno di significativi interventi umani di estrazione ed elaborazione;

Classificazione 5 Stelle - Una Stella

Una Stella: Open Licence

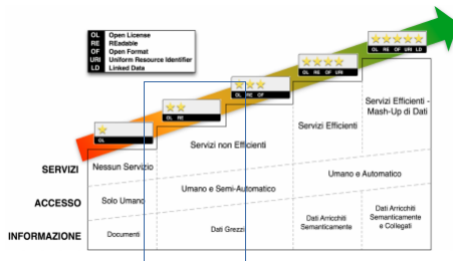


La *prima stella* si ottiene rilasciando i dati in qualunque formato ma con una *licenza aperta*. Rientrano in questa categoria ad esempio le scansioni dei documenti.

- **INFORMAZIONE:** *documenti* - i dati sono incorporati all'interno di documenti senza struttura;
- **ACCESSO:** *solo umano* - solo gli umani sono in grado di leggere i documenti senza struttura e quindi dare un senso ai dati in esso presenti;
- **SERVIZI:** *nessuno* può essere abilitato a meno di significativi interventi umani di estrazione ed elaborazione;

Classificazione 5 Stelle - Due Stelle

Due Stelle: Open Licence, (Machine) Readable



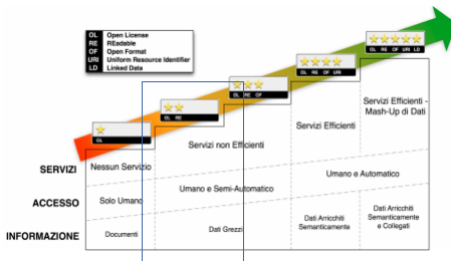
La *seconda stella* si ottiene se i dati sono forniti in un formato leggibile da un agente automatico.

Rientrano in questa categoria ad esempio i files in formato *excel*.

- **INFORMAZIONE:** *dati grezzi (o semi-strutturati)* - i dati sono leggibili anche da un programma ma necessita un intervento umano per interpretarli;
- **ACCESSO:** *umano e semi-automatico* - i software possono leggere i dati ma non sono in grado di interpretarli automaticamente;
- **SERVIZI:** *non efficienti* - servizi realizzati ad-hoc e devono incorporare al loro interno i dati;

Classificazione 5 Stelle - Due Stelle

Due Stelle: Open Licence, (Machine) Readable



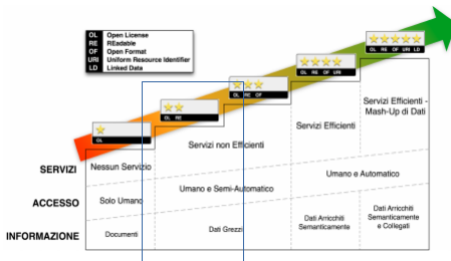
La *seconda stella* si ottiene se i dati sono forniti in un formato leggibile da un agente automatico.

Rientrano in questa categoria ad esempio i files in formato *excel*.

- **INFORMAZIONE:** *dati grezzi (o semi-strutturati)* - i dati sono leggibili anche da un programma ma necessita un intervento umano per interpretarli;
- **ACCESSO:** *umano e semi-automatico* - i software possono leggere i dati ma non sono in grado di interpretarli automaticamente;
- **SERVIZI:** *non efficienti* - servizi realizzati ad-hoc e devono incorporare al loro interno i dati;

Classificazione 5 Stelle - Due Stelle

Due Stelle: Open Licence, (Machine) Readable



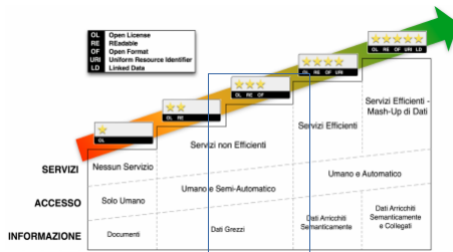
La *seconda stella* si ottiene se i dati sono forniti in un formato leggibile da un agente automatico.

Rientrano in questa categoria ad esempio i files in formato *excel*.

- **INFORMAZIONE:** *dati grezzi (o semi-strutturati)* - i dati sono leggibili anche da un programma ma necessita un intervento umano per interpretarli;
- **ACCESSO:** *umano e semi-automatico* - i software possono leggere i dati ma non sono in grado di interpretarli automaticamente;
- **SERVIZI:** *non efficienti* - servizi realizzati ad-hoc e devono incorporare al loro interno i dati;

Classificazione 5 Stelle - Tre Stelle

Tre Stelle: Open Licence, (Machine) Readable, Open Format

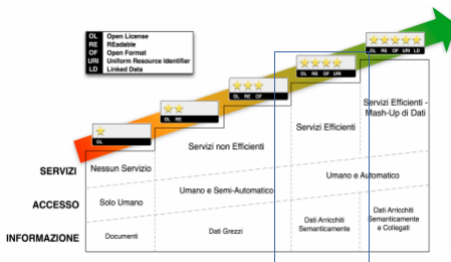


La *terza stella* viene attribuita se i dati sono rilasciati in un formato *aperto*. Rientrano in questa categoria ad esempio i files *json*, *csv*, *xml*.

- **INFORMAZIONE:** *dati grezzi (o semi-strutturati)* - i dati sono leggibili anche da un programma ma necessita un intervento umano per interpretarli;
- **ACCESSO:** *umano e semi-automatico* - i software possono leggere i dati ma non sono in grado di interpretarli automaticamente;
- **SERVIZI:** *non efficienti* - servizi realizzati ad-hoc e devono incorporare al loro interno i dati;

Classificazione 5 Stelle - Quattro Stelle

Quattro Stelle: Open Licence, (Machine) Readable, Open Format, URI

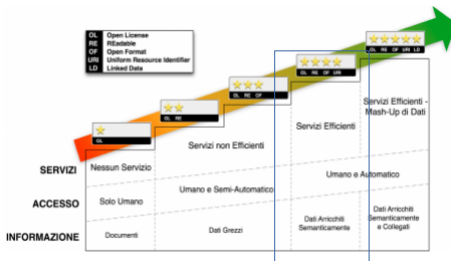


La *quarta stella* si ottiene esponendo i dati con le tecnologie del web semantico (RDF e SPARQL).

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti* - servizi che sfruttano accessi diretti a Web per reperire i dati.

Classificazione 5 Stelle - Quattro Stelle

Quattro Stelle: Open Licence, (Machine) Readable, Open Format, URI

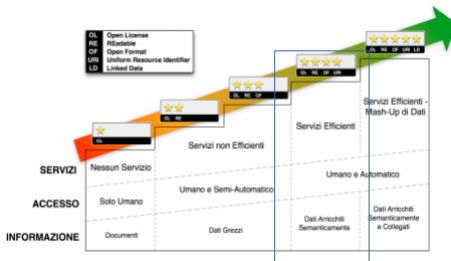


La *quarta stella* si ottiene esponendo i dati con le tecnologie del web semantico (RDF e SPARQL).

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti* - servizi che sfruttano accessi diretti a Web per reperire i dati.

Classificazione 5 Stelle - Quattro Stelle

Quattro Stelle: Open Licence, (Machine) Readable, Open Format, URI

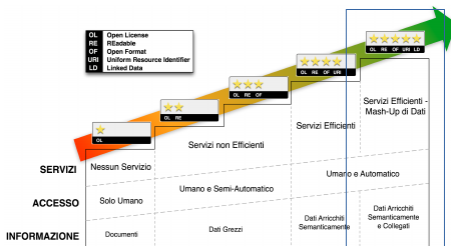


La *quarta stella* si ottiene esponendo i dati con le tecnologie del web semantico (RDF e SPARQL).

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti* - servizi che sfruttano accessi diretti a Web per reperire i dati.

Classificazione 5 Stelle - Cinque Stelle

Cinque Stelle: Open Licence, (Machine) Readable, Open Format, URI, *Linked* Data

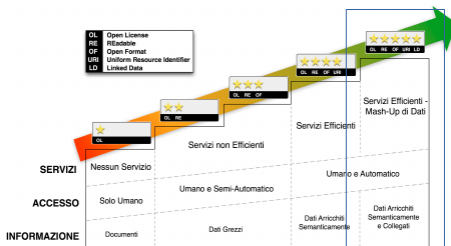


La *quinta stella* viene attribuita quando i dati contengono riferimenti a dataset di terze parti.

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti e con mashup di dati* - servizi che sfruttano sia accessi diretti a Web sia l'informazione ulteriore catturata attraverso i *link* dei dati di interesse.

Classificazione 5 Stelle - Cinque Stelle

Cinque Stelle: Open Licence, (Machine) Readable, Open Format, URI, *Linked* Data

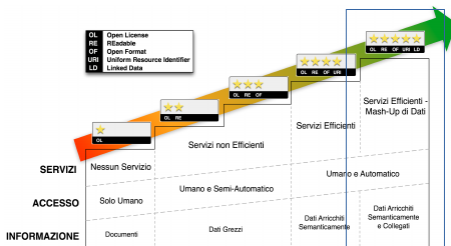


La *quinta stella* viene attribuita quando i dati contengono riferimenti a dataset di terze parti.

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti e con mashup di dati* - servizi che sfruttano sia accessi diretti a Web sia l'informazione ulteriore catturata attraverso i *link* dei dati di interesse.

Classificazione 5 Stelle - Cinque Stelle

Cinque Stelle: Open Licence, (Machine) Readable, Open Format, URI, *Linked* Data



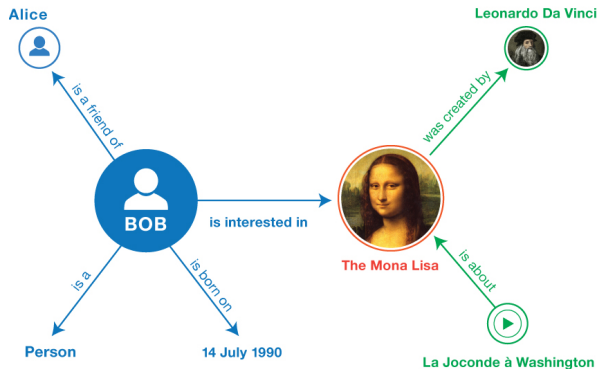
La *quinta stella* viene attribuita quando i dati contengono riferimenti a dataset di terze parti.

- **INFORMAZIONE:** *dati arricchiti semanticamente* - i dati sono descritti usando tecnologie del Web Semantico;
- **ACCESSO:** *umano e automatico* - i software sono in grado di elaborare i dati quasi senza ulteriori interventi umani (livelli 4 e 5);
- **SERVIZI:** *efficienti e con mashup di dati* - servizi che sfruttano sia accessi diretti a Web sia l'informazione ulteriore catturata attraverso i *link* dei dati di interesse.

Classificazione 5 Stelle - Cinque Stelle - Esempio

Cinque Stelle: Open Licence, (Machine) Readable, Open Format, URI, *Linked* Data

In altre parole, è possibile collegare dataset differenti ed eterogenei. Nell'esempio che segue i diversi colori indicano le collocazioni degli elementi e delle informazioni in dataset differenti.



Limiti del World Wide Web (1/4)

*The Web was designed as an information space, with the goal that it should be useful not only for human-human communication, but also that machines would be able to participate and help. One of the major obstacles to this has been the fact that **most information on the Web is designed for human consumption**, and even if it was derived from a database with well defined meanings (in at least some terms) for its columns, that **the structure of the data is not evident to a robot browsing the web.***

Semantic Web Roadmap, Tim Berners-Lee, 1998.

Limiti del World Wide Web (2/4)

Alcuni problemi nell'interpretazione di testi derivano da:

Lingue Differenti e.g. *Parigi* e *Paris* possono indicare la stessa città.

Omonimie e.g. esistono svariate città chiamate *Paris* nel mondo (Arkansas, Idaho, Illinois, Kentucky, Maine, Michigan, Missouri, New York, ...);

Limiti del World Wide Web (3/4)

La situazione si complica in presenza di contenuti multimediali.



Limiti del World Wide Web (4/4)

Come conseguenza, spesso è impossibile eseguire su web ricerche *complesse* ottenendo risultati accurati. Ad esempio, cercando sul web *"Federico II places"* non si ottengono risultati in prima pagina su Federico II, ma solo sull'omonima università:

- ① Università degli Studi di Napoli "Federico II" — OPEN Places
- ② AOU - Policlinico "Federico II" - Napoli, Italy - Hospital — Facebook
- ③ Federico II Ingegneria Via Claudio - College and University — Facebook
- ④ MARIA CATERINA FONTE - www.docenti.unina.it

II Web Semantico (1/2)

[...] the Semantic Web approach instead develops languages for expressing information in a machine processable form.

Semantic Web Roadmap, Tim Berners-Lee, 1998.

About: [Frederick II, Holy Roman Emperor](#)
[About DBpedia](#)

An Entity of Type : [agent](#), from Named Graph : <http://dbpedia.org>,
within Data Space : dbpedia.org

Frederick II (26 December 1194 – 13 December 1250), was one of the most powerful Holy Roman Emperors of the Middle Ages and head of the House of Hohenstaufen. His political and cultural ambitions, based in Sicily and stretching through Italy to Germany, and even to Jerusalem, were enormous; however, his enemies, especially the popes, prevailed, and his dynasty collapsed soon after his death.

Property	Value
<code>dbpedia-owl:abstract</code>	■ 1250-01-01 00:00:00 (xsd:date)
<code>dbpedia-owl:activeYearsStartYear</code>	■ 1220-01-01 00:00:00 (xsd:date)
<code>dbpedia-owl:birthDate</code>	■ 1194-12-26 (xsd:date)
<code>dbpedia-owl:birthPlace</code>	■ dbpedia:lesi ■ dbpedia:Kingdom_of_Italy_(medieval) ■ dbpedia:Marche
<code>dbpedia-owl:deathDate</code>	■ 1250-12-13 (xsd:date)
<code>dbpedia-owl:deathPlace</code>	■ dbpedia:Torremaqqiore

Figure : Federico II su dbpedia.org

Il Web Semantico

Il *Web Semantico* nasce per associare informazioni *strutturate* alle pagine web, che solitamente sono composte da testo libero.¹⁰

I linguaggi di rappresentazione usati nel Web semantico hanno una sintassi rigorosa e sono dotati di una semantica formale.

Questo rende possibile effettuare interrogazioni complesse sui dataset, ottenendo dei risultati precisi anche se a volte parziali:

Q = “Luoghi di nascita di Federico II e dei suoi parenti stretti” .

¹⁰Vedi *Semantic Web Roadmap*, Tim Berners-Lee, 1998.

Il Web Semantico

Il *Web Semantico* nasce per associare informazioni *strutturate* alle pagine web, che solitamente sono composte da testo libero.¹⁰

I *linguaggi di rappresentazione* usati nel Web semantico hanno una *sintassi rigorosa* e sono dotati di una *semantica formale*.

Questo rende possibile effettuare interrogazioni complesse sui dataset, ottenendo dei risultati precisi anche se a volte parziali:

Q = “Luoghi di nascita di Federico II e dei suoi parenti stretti” .

¹⁰Vedi *Semantic Web Roadmap*, Tim Berners-Lee, 1998.

Il Web Semantico

Il *Web Semantico* nasce per associare informazioni *strutturate* alle pagine web, che solitamente sono composte da testo libero.¹⁰

I *linguaggi di rappresentazione* usati nel Web semantico hanno una *sintassi rigorosa* e sono dotati di una *semantica formale*.

Questo rende possibile effettuare interrogazioni complesse sui dataset, ottenendo dei risultati precisi anche se a volte parziali:

Q = “Luoghi di nascita di Federico II e dei suoi parenti stretti” .

¹⁰Vedi *Semantic Web Roadmap*, Tim Berners-Lee, 1998.

Linked Data (1/2)

I dataset nel Web Semantico possono essere *collegati* tra loro. Ad esempio, una stessa risorsa può essere descritta sotto diversi aspetti in dataset differenti.

Ad esempio, la città di Catania è presente:

- come pubblica amministrazione nel dataset del sistema pubblico di connettività e cooperazione¹¹ con la url
`http://spcdata.digitpa.gov.it/Amministrazione/c_c351`;
- come divisione amministrativa nel dataset `http://www.geonames.org`;
- come area territoriale nel dataset dell'ISTAT
`http://linkedstat.spaziodati.eu/`.

¹¹`http://spcdata.digitpa.gov.it/`

Linked Data (2/2)

È possibile effettuare interrogazioni che coinvolgano diversi dataset (anche eterogenei).

Ad esempio, la seguente query può essere eseguita interrogando un data set contenente dati storici ed uno sulle strutture ricettive:

Q = "Strutture ricettive nei luoghi di nascita di Federico II e dei suoi parenti stretti."

Linked Open Data Cloud (1/2)

The Semantic Web is a web of data, in some ways like a global database.

Semantic Web Roadmap, Tim Berners-Lee, 1998.

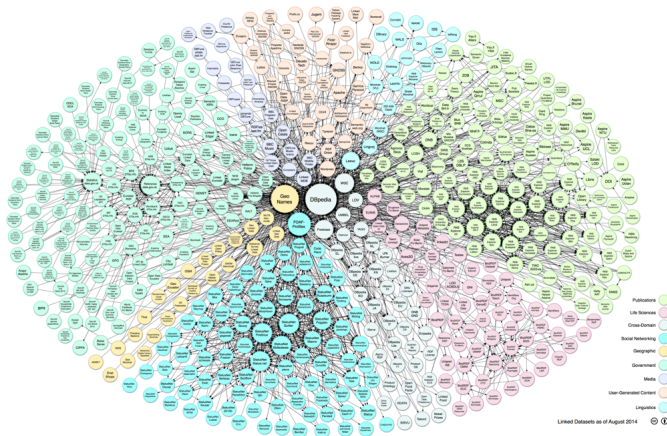


Figure : Linked Open Data Cloud

Linked Open Data Cloud (2/2)

Nel *Linked Open Data Cloud* sono presenti 365 dataset (fonte <http://stats.lod2.eu/>).

Alcuni dataset:

- *DBPedia* (dbpedia.org) corrispondente a wikipedia.org;
- *Linked Movie Database* (<http://linkedmdb.org/>) controparte sul Web Semantico di *Internet Movie Database* (<http://www.imdb.com/>);
- *Linked GeoData* (<http://linkedgeo.org>) contiene i dati di *OpenStreetMap* (<http://www.openstreetmap.org/>);
- *AGROVOC* (<http://aims.fao.org/agrovoc>) è il dataset della FAO (<http://fao.org>);
- *Europeana* (<http://pro.europeana.eu/linked-open-data>) contiene dati su beni culturali e tradizioni Europee.

Ontologie

I linguaggi di rappresentazione usati nel Web Semantico si basano tutti sulla nozione di *ontologia*, mutuata dall'ambito dei sistemi di rappresentazione della conoscenza.

Una *ontologia* è una descrizione *parziale* del mondo:

- descrive una porzione del mondo, spesso è limitata ad un'unico *dominio di conoscenza*;
- non si assume che i fatti non esplicitamente presenti nell'ontologia siano falsi (*Open World Assumption*).

Essa è costituita da un insieme finito di *affermazioni*. Ad esempio:

- Tutti gli esseri umani sono mortali;
- Socrate è mortale;
- Alice è la madre di Roberto.

Ontologie - Affermazioni

Le affermazioni contenute in una ontologia sono di tre tipi:

Constraints: impongono dei vincoli *semantici* sul dominio di conoscenza che si va a rappresentare. La notazione richiama quella insiemistica;

$$\textit{HumanBeing} \sqsubseteq \textit{Mortal}$$

Property Assertions: impongono una relazione tra due elementi del dominio;

$$\textit{Alice} \textit{ motherOf } \textit{Bob}$$

Class Assertions: indicano l'appartenenza di un elemento ad un insieme.

$$\textit{HumanBeing}(\textit{Socrate})$$

Ontologie - Sintassi

Riportiamo la definizione formale per la sintassi delle ontologie.

Siano N_C , N_P , N_I tre insiemi infiniti, numerabili e a due a due disgiunti di nomi di *classe*, *proprietà* e *individuo*, rispettivamente.

Una *ontologia* è un insieme finito di asserzioni dei seguenti tipi:

(Constraints)	$C \sqsubseteq D$
	$P \sqsubseteq Q$
	$\text{dom}(P) \sqsubseteq C$
	$\text{range}(P) \sqsubseteq C$

(Class Assertions)	$C(a)$
--------------------	--------

Property Assertions	$a P b$ (equivalente $P(a, b)$)
---------------------	----------------------------------

dove $C, D \in N_C$, $P, Q \in N_P$ e $a, b \in N_I$.

Si noti che la grammatica per i vincoli ivi riportata è *minimale*. Esistono linguaggi di rappresentazione che permettono di esprimere vincoli più complessi.

Ontologie - Esempio

Riportiamo un esempio di ontologia. Siano $Human, Woman, Man, Male, Female \in N_C$, $relative, child \in N_P$, $Alice, Bob, Charlie \in N_I$.

$$\mathcal{O} = \{ \begin{array}{l} Woman \sqsubseteq Human, \\ Man \sqsubseteq Human, \\ Woman \sqsubseteq Female, \\ Man \sqsubseteq Male, \\ Woman(Alice), \\ Man(Bob), \\ Alice \text{ relative } Bob, \\ Alice \text{ child } Charlie \end{array} \}$$

Ontologie - L'Editor Protégé

*Protégé*¹² è una suite per la modellazione di ontologie del Web Semantico, disponibile sia in versione web, che in versione installabile localmente (*Protégé Desktop*).

Permette di definire gerarchie di classi (tab *Classes*) a partire dalla classe radice *Thing*. Per ogni classe è possibile definire *label* e *comment* come annotazioni.

Analogamente è possibile definire gerarchie di proprietà (tab *Object Properties*) a partire dalla proprietà radice *topObjectProperty* e di definire annotazioni per le proprietà. Inoltre, è possibile imporre dei vincoli di dominio e codominio per le proprietà.

Infine è possibile definire degli individui (tab *Individuals*) associarli a delle classi di appartenenza e metterli in relazione tra loro attraverso delle proprietà.

¹²<http://protege.stanford.edu/products.php>

Ontologie - L'Editor Protégé

*Protégé*¹² è una suite per la modellazione di ontologie del Web Semantico, disponibile sia in versione web, che in versione installabile localmente (*Protégé Desktop*).

Permette di definire gerarchie di classi (tab *Classes*) a partire dalla classe radice *Thing*. Per ogni classe è possibile definire *label* e *comment* come annotazioni.

Analogamente è possibile definire gerarchie di proprietà (tab *Object Properties*) a partire dalla proprietà radice *topObjectProperty* e di definire annotazioni per le proprietà. Inoltre, è possibile imporre dei vincoli di dominio e codominio per le proprietà.

Infine è possibile definire degli individui (tab *Individuals*) associarli a delle classi di appartenenza e metterli in relazione tra loro attraverso delle proprietà.

¹²<http://protege.stanford.edu/products.php>

Ontologie - L'Editor Protégé

*Protégé*¹² è una suite per la modellazione di ontologie del Web Semantico, disponibile sia in versione web, che in versione installabile localmente (*Protégé Desktop*).

Permette di definire gerarchie di classi (tab *Classes*) a partire dalla classe radice *Thing*. Per ogni classe è possibile definire *label* e *comment* come annotazioni.

Analogamente è possibile definire gerarchie di proprietà (tab *Object Properties*) a partire dalla proprietà radice *topObjectProperty* e di definire annotazioni per le proprietà. Inoltre, è possibile imporre dei vincoli di dominio e codominio per le proprietà.

Infine è possibile definire degli individui (tab *Individuals*) associarli a delle classi di appartenenza e metterli in relazione tra loro attraverso delle proprietà.

¹²<http://protege.stanford.edu/products.php>

Ontologie - L'Editor Protégé

*Protégé*¹² è una suite per la modellazione di ontologie del Web Semantico, disponibile sia in versione web, che in versione installabile localmente (*Protégé Desktop*).

Permette di definire gerarchie di classi (tab *Classes*) a partire dalla classe radice *Thing*. Per ogni classe è possibile definire *label* e *comment* come annotazioni.

Analogamente è possibile definire gerarchie di proprietà (tab *Object Properties*) a partire dalla proprietà radice *topObjectProperty* e di definire annotazioni per le proprietà. Inoltre, è possibile imporre dei vincoli di dominio e codominio per le proprietà.

Infine è possibile definire degli individui (tab *Individuals*) associarli a delle classi di appartenenza e metterli in relazione tra loro attraverso delle proprietà.

¹²<http://protege.stanford.edu/products.php>

Ontologie - Interpretazioni

Per definire la semantica delle ontologie è necessario prima introdurre il concetto di interpretazione.

Una *interpretazione* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ è una coppia $\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}$ dove:

- $\Delta^{\mathcal{I}}$ è un insieme non vuoto;
- $\cdot^{\mathcal{I}}$ è una funzione (polimorfa) che associa
 - ad ogni nome di concetto in N_C un sottoinsieme di $\Delta^{\mathcal{I}}$,
 - ad ogni nome di proprietà in N_P una relazione su $\Delta^{\mathcal{I}}$,
 - ad ogni nome di individuo in N_I un elemento di $\Delta^{\mathcal{I}}$.

Consideriamo ad esempio la seguente ontologia

$\mathcal{O} = \{ \text{Woman} \sqsubseteq \text{Human}, \text{Man} \sqsubseteq \text{Human}, \text{Woman} \sqsubseteq \text{Female}, \text{Man} \sqsubseteq \text{Male}, \\ \text{Woman}(\text{Alice}), \text{Man}(\text{Bob}), \text{Alice relative Bob}, \text{Alice child Charlie} \}$

Una possibile interpretazione \mathcal{I} è la seguente:

$\Delta^{\mathcal{I}}$	=	\mathbb{N}
$\text{Alice}^{\mathcal{I}}$	=	0
$\text{Bob}^{\mathcal{I}}$	=	1
$\text{Charlie}^{\mathcal{I}}$	=	2
$\text{Human}^{\mathcal{I}}$	=	$\{0, 1, 2\}$
$\text{Male}^{\mathcal{I}}$	=	$\{1, 2\}$
$\text{Female}^{\mathcal{I}}$	=	$\{0\}$
$\text{Man}^{\mathcal{I}}$	=	$\{1, 2\}$
$\text{Woman}^{\mathcal{I}}$	=	$\{0\}$

NB: è sufficiente prendere in considerazione ai nostri fini i simboli che compaiono nell'ontologia.

Ontologie - Interpretazioni

Per definire la semantica delle ontologie è necessario prima introdurre il concetto di interpretazione.

Una *interpretazione* $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ è una coppia $\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}}$ dove:

- $\Delta^{\mathcal{I}}$ è un insieme non vuoto;
- $\cdot^{\mathcal{I}}$ è una funzione (polimorfa) che associa
 - ad ogni nome di concetto in N_C un sottoinsieme di $\Delta^{\mathcal{I}}$,
 - ad ogni nome di proprietà in N_P una relazione su $\Delta^{\mathcal{I}}$,
 - ad ogni nome di individuo in N_I un elemento di $\Delta^{\mathcal{I}}$.

Consideriamo ad esempio la seguente ontologia

$\mathcal{O} = \{ \text{Woman} \sqsubseteq \text{Human}, \text{Man} \sqsubseteq \text{Human}, \text{Woman} \sqsubseteq \text{Female}, \text{Man} \sqsubseteq \text{Male}, \\ \text{Woman}(\text{Alice}), \text{Man}(\text{Bob}), \text{Alice relative Bob}, \text{Alice child Charlie} \}$

Una possibile interpretazione \mathcal{I} è la seguente:

$\Delta^{\mathcal{I}}$	=	\mathbb{N}
$\text{Alice}^{\mathcal{I}}$	=	0
$\text{Bob}^{\mathcal{I}}$	=	1
$\text{Charlie}^{\mathcal{I}}$	=	2
$\text{Human}^{\mathcal{I}}$	=	$\{0, 1, 2\}$
$\text{Male}^{\mathcal{I}}$	=	$\{1, 2\}$
$\text{Female}^{\mathcal{I}}$	=	$\{0\}$
$\text{Man}^{\mathcal{I}}$	=	$\{1, 2\}$
$\text{Woman}^{\mathcal{I}}$	=	$\{0\}$

NB: è sufficiente prendere in considerazione ai nostri fini i simboli che compaiono nell'ontologia.

Ontologie - Soddisfacibilità

La semantica formale di cui sono equipaggiate le ontologie abilita l'esecuzione automatica di *reasoning tasks*. Quello fondamentale è la verifica di *Soddisfacibilità*, che permette di controllare che una ontologia non sia autocontraddittoria.

La nozione di soddisfacibilità per le ontologie è definita come segue. Sia $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ una interpretazione.

$$\begin{array}{ll}
 \mathcal{I} \text{ soddisfa } C \sqsubseteq D & \iff C^{\mathcal{I}} \subseteq D^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } P \sqsubseteq Q & \iff P^{\mathcal{I}} \subseteq Q^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } \text{dom}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(x \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } \text{range}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(y \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } C(a) & \iff a^{\mathcal{I}} \in C^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } a P b & \iff [a^{\mathcal{I}}, b^{\mathcal{I}}] \in P^{\mathcal{I}}
 \end{array}$$

per ogni $C, D \in N_C$, $P, Q \in PNames$, $a, b \in N_I$.

\mathcal{I} soddisfa una ontologia \mathcal{O} se e solo se \mathcal{I} soddisfa tutti i vincoli e le asserzioni in \mathcal{O} .

Una ontologia \mathcal{O} è detta *soddisfacibile* (o anche *consistente*) se e solo se esiste una interpretazione \mathcal{I} che la soddisfa.

Ontologie - Soddisfacibilità

La semantica formale di cui sono equipaggiate le ontologie abilita l'esecuzione automatica di *reasoning tasks*. Quello fondamentale è la verifica di *Soddisfacibilità*, che permette di controllare che una ontologia non sia autocontraddittoria.

La nozione di soddisfacibilità per le ontologie è definita come segue. Sia $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ una interpretazione.

$$\begin{array}{ll}
 \mathcal{I} \text{ soddisfa } C \sqsubseteq D & \iff C^{\mathcal{I}} \subseteq D^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } P \sqsubseteq Q & \iff P^{\mathcal{I}} \subseteq Q^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } \text{dom}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(x \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } \text{range}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(y \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } C(a) & \iff a^{\mathcal{I}} \in C^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } a P b & \iff [a^{\mathcal{I}}, b^{\mathcal{I}}] \in P^{\mathcal{I}}
 \end{array}$$

per ogni $C, D \in N_C$, $P, Q \in PNames$, $a, b \in N_I$.

\mathcal{I} soddisfa una ontologia \mathcal{O} se e solo se \mathcal{I} soddisfa tutti i vincoli e le asserzioni in \mathcal{O} .

Una ontologia \mathcal{O} è detta *soddisfacibile* (o anche *consistente*) se e solo se esiste una interpretazione \mathcal{I} che la soddisfa.

Ontologie - Soddisfacibilità

La semantica formale di cui sono equipaggiate le ontologie abilita l'esecuzione automatica di *reasoning tasks*. Quello fondamentale è la verifica di *Soddisfacibilità*, che permette di controllare che una ontologia non sia autocontraddittoria.

La nozione di soddisfacibilità per le ontologie è definita come segue. Sia $\mathcal{I} = (\Delta^{\mathcal{I}}, \cdot^{\mathcal{I}})$ una interpretazione.

$$\begin{array}{ll}
 \mathcal{I} \text{ soddisfa } C \sqsubseteq D & \iff C^{\mathcal{I}} \subseteq D^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } P \sqsubseteq Q & \iff P^{\mathcal{I}} \subseteq Q^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } \text{dom}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(x \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } \text{range}(P) \sqsubseteq C & \iff (\forall [x, y] \in P^{\mathcal{I}})(y \in C^{\mathcal{I}}) \\
 \mathcal{I} \text{ soddisfa } C(a) & \iff a^{\mathcal{I}} \in C^{\mathcal{I}} \\
 \mathcal{I} \text{ soddisfa } a P b & \iff [a^{\mathcal{I}}, b^{\mathcal{I}}] \in P^{\mathcal{I}}
 \end{array}$$

per ogni $C, D \in N_C$, $P, Q \in PNames$, $a, b \in N_I$.

\mathcal{I} soddisfa una ontologia \mathcal{O} se e solo se \mathcal{I} soddisfa tutti i vincoli e le asserzioni in \mathcal{O} .

Una ontologia \mathcal{O} è detta *soddisfacibile* (o anche *consistente*) se e solo se esiste una interpretazione \mathcal{I} che la soddisfa.

Ontologie - Soddisfacibilità - Esempi (1/2)

Consideriamo la seguente ontologia

$$\mathcal{O} = \{ \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human}, \\ \text{Socrate teacherOf Plato} \}$$

Consideriamo la seguente interpretazione (di Herbrandt) \mathcal{I} :

$$\begin{aligned} \text{Socrate}^{\mathcal{I}} &= \text{Socrate} \\ \text{Plato}^{\mathcal{I}} &= \text{Plato} \\ \text{Human}^{\mathcal{I}} &= \{ \text{Socrate}, \text{Plato} \} \\ \text{teacherOf}^{\mathcal{I}} &= \{ [\text{Socrate}, \text{Plato}] \} \end{aligned}$$

$$\begin{aligned} \mathcal{I} \text{ soddisfa } \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human} &\iff (\forall [x, y] \in \text{teacherOf}^{\mathcal{I}})(x \in \text{Human}^{\mathcal{I}}) \\ \mathcal{I} \text{ soddisfa } \text{Socrate teacherOf Plato} &\iff [\text{Socrate}^{\mathcal{I}}, \text{Plato}^{\mathcal{I}}] \in \text{teacherOf}^{\mathcal{I}} \end{aligned}$$

Quindi \mathcal{I} soddisfa \mathcal{O} .

Quindi \mathcal{O} è soddisfacibile.

Ontologie - Soddisfacibilità - Esempi (1/2)

Consideriamo la seguente ontologia

$$\mathcal{O} = \{ \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human}, \\ \text{Socrate teacherOf Plato} \}$$

Consideriamo la seguente interpretazione (di Herbrandt) \mathcal{I} :

$$\begin{aligned} \text{Socrate}^{\mathcal{I}} &= \text{Socrate} \\ \text{Plato}^{\mathcal{I}} &= \text{Plato} \\ \text{Human}^{\mathcal{I}} &= \{ \text{Socrate}, \text{Plato} \} \\ \text{teacherOf}^{\mathcal{I}} &= \{ [\text{Socrate}, \text{Plato}] \} \end{aligned}$$

$$\begin{aligned} \mathcal{I} \text{ soddisfa } \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human} &\iff (\forall [x, y] \in \text{teacherOf}^{\mathcal{I}})(x \in \text{Human}^{\mathcal{I}}) \\ \mathcal{I} \text{ soddisfa } \text{Socrate teacherOf Plato} &\iff [\text{Socrate}^{\mathcal{I}}, \text{Plato}^{\mathcal{I}}] \in \text{teacherOf}^{\mathcal{I}} \end{aligned}$$

Quindi \mathcal{I} soddisfa \mathcal{O} .

Quindi \mathcal{O} è soddisfacibile.

Ontologie - Soddisfacibilità - Esempi (1/2)

Consideriamo la seguente ontologia

$$\mathcal{O} = \{ \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human}, \\ \text{Socrate teacherOf Plato} \}$$

Consideriamo la seguente interpretazione (di Herbrandt) \mathcal{I} :

$$\begin{aligned} \text{Socrate}^{\mathcal{I}} &= \text{Socrate} \\ \text{Plato}^{\mathcal{I}} &= \text{Plato} \\ \text{Human}^{\mathcal{I}} &= \{ \text{Socrate}, \text{Plato} \} \\ \text{teacherOf}^{\mathcal{I}} &= \{ [\text{Socrate}, \text{Plato}] \} \end{aligned}$$

$$\begin{aligned} \mathcal{I} \text{ soddisfa } \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human} &\iff (\forall [x, y] \in \text{teacherOf}^{\mathcal{I}})(x \in \text{Human}^{\mathcal{I}}) \\ \mathcal{I} \text{ soddisfa } \text{Socrate teacherOf Plato} &\iff [\text{Socrate}^{\mathcal{I}}, \text{Plato}^{\mathcal{I}}] \in \text{teacherOf}^{\mathcal{I}} \end{aligned}$$

Quindi \mathcal{I} soddisfa \mathcal{O} .

Quindi \mathcal{O} è soddisfacibile.

Ontologie - Soddisfacibilità - Esempi (2/2)

Consideriamo la seguente ontologia

$$\mathcal{O} = \{ \text{dom}(\text{teacherOf}) \sqsubseteq \text{Human}, \\ \text{Socrate teacherOf Plato} \}$$

La seguente interpretazione (di Herbrandt) \mathcal{I}_1 NON soddisfa \mathcal{O} :

$$\begin{aligned} \text{Socrate}^{\mathcal{I}_1} &= \text{Socrate} \\ \text{Plato}^{\mathcal{I}_1} &= \text{Plato} \\ \text{Human}^{\mathcal{I}_1} &= \{ \text{Plato} \} \\ \text{teacherOf}^{\mathcal{I}_1} &= \{ [\text{Socrate}, \text{Plato}] \} \end{aligned}$$

Infatti \mathcal{I}_1 non soddisfa $\text{dom}(\text{teacherOf}) \sqsubseteq \text{Human}$ perchè *Socrate* non è nell'insieme *Human* (interpretati con \mathcal{I}_1).

Ontologie - Implicazione

Date due Ontologie \mathcal{O} e \mathcal{O}' , si dice che \mathcal{O} *implica* \mathcal{O}' se e solo se tutte le interpretazioni che soddisfano \mathcal{O} soddisfano anche \mathcal{O}' .

La verifica di implicazione può essere utilizzata per ricavare tutte le *conseguenze logiche* di una ontologia.

Ontologie - Inferenze - Esempio

Riportiamo un esempio di ontologia. Siano $HumanBeing, Mortal \in N_C$, $teacherOf \in N_P$, $Socrate, Platone \in N_I$.

Mediante *reasoning* è possibile esplicitare ulteriori affermazioni.

$$\begin{aligned} \mathcal{O} = \{ & \mathbf{HumanBeing} \sqsubseteq \mathbf{Mortal}, \\ & \text{range}(teacherOf) \sqsubseteq \mathbf{HumanBeing}, \\ & \mathbf{HumanBeing}(\mathbf{Socrate}), \\ & \mathbf{Socrate} \text{ teacherOf } \mathbf{Platone} \} \end{aligned} \quad \Longrightarrow \quad \mathcal{O}' = \{ \mathbf{Mortal}(\mathbf{Socrate}), \}$$

Ontologie - Esempio

Riportiamo un esempio di ontologia. Siano $HumanBeing, Mortal \in N_C$,
 $teacherOf \in N_P$, $Socrate, Platone \in N_I$.

Mediante *reasoning* è possibile esplicitare ulteriori affermazioni.

$$\begin{aligned} \mathcal{O} = \{ & HumanBeing \sqsubseteq Mortal, \\ & range(teacherOf) \sqsubseteq HumanBeing, \\ & HumanBeing(Socrate), \\ & Socrate \text{ teacherOf } Platone \} \end{aligned} \quad \Rightarrow \quad \begin{aligned} \mathcal{O}' = \{ & Mortal(Socrate), \\ & HumanBeing(Platone), \\ & \} \end{aligned}$$

Ontologie - Esempio

Riportiamo un esempio di ontologia. Siano $HumanBeing, Mortal \in N_C$,
 $teacherOf \in N_P$, $Socrate, Platone \in N_I$.

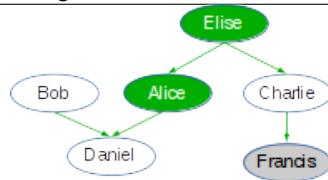
Mediante *reasoning* è possibile esplicitare ulteriori affermazioni.

$$\begin{aligned} \mathcal{O} = \{ & \mathbf{HumanBeing} \sqsubseteq \mathbf{Mortal} \\ & \text{range}(teacherOf) \sqsubseteq HumanBeing, \\ & HumanBeing(Socrate), \\ & Socrate \text{ teacherOf } Platone \} \end{aligned} \quad \Longrightarrow \quad \begin{aligned} \mathcal{O}' = \{ & Mortal(Socrate), \\ & \mathbf{HumanBeing(Platone)} \\ & \mathbf{Mortal(Platone)} \} \end{aligned}$$

Interrogazioni

Il metodo più immediato per ottenere informazioni da una ontologia è il *Conjunctive Query Answering*. Consideriamo ad esempio la seguente ontologia:

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$



Alcune interrogazioni che è possibile effettuare con il conjunctive query answering sono:

- “Trova tutti gli individui maschi.”
- “Chi sono gli individui con almeno un figlio maschio?”
- “Chi sono i figli di *Alice*?”
- “Chi sono gli individui con almeno un figlio maschio ed una femmina?”
- “Chi sono gli individui maschi con almeno un figlio maschio?”

Formule Atomiche

Per definire in maniera rigorosa le query congiuntive è necessario definire preliminarmente l'insieme delle *formule atomiche*.

Sia $V = \{x, y, z, \dots\}$ l'insieme infinito, numerabile e disgiunto da N_C , N_P e N_I delle *variabili*. Le *formule atomiche* sono espressioni dei due seguenti tipi:

$$C(\alpha), \quad P(\alpha, \beta)$$

con $\alpha, \beta \in N_I \cup V$, $C \in N_C$ e $P \in N_P$.

Esempi di formule atomiche sono:

- *HumanBeing*(x),
- x *childOf* *Alice*,
- *Bob* *childOf* x ,
- x *childOf* y ,
- *Mortal*(*Socrate*),
- *Alice* *childOf* *Elise*

con *HumanBeing*, *Mortal* $\in N_C$, *childOf* $\in N_P$, *Alice*, *Bob*, *Elise* $\in N_I$ e $x, y \in V$.

Formule Atomiche Chiuse

Una formula atomica nella quale non compaiano variabili si dice *chiusa*.

Negli esempi che seguono sono evidenziate le formule atomiche chiuse:

- *HumanBeing*(*x*),
- *x childOf Alice*,
- *Bob childOf x*,
- *x childOf y*,
- *Mortal*(*Socrate*),
- *Alice childOf Elise*

con *HumanBeing*, *Mortal* $\in N_C$, *childOf* $\in N_P$, *Alice*, *Bob*, *Elise* $\in N_I$ e *x*, *y* $\in V$.

Formule Atomiche Chiuse

Una formula atomica nella quale non compaiano variabili si dice *chiusa*.

Negli esempi che seguono sono evidenziate le formule atomiche chiuse:

- *HumanBeing*(*x*),
- *x childOf Alice*,
- *Bob childOf x*,
- *x childOf y*,
- **Mortal(Socrate)**,
- **Alice childOf Elise**

con *HumanBeing*, *Mortal* $\in N_C$, *childOf* $\in N_P$, *Alice*, *Bob*, *Elise* $\in N_I$ e *x*, *y* $\in V$.

Le asserzioni presenti nelle ontologie sono formule atomiche chiuse.

Query Congiuntive -

Una *query congiuntiva* è una congiunzione finita di formule atomiche $T_1 \wedge \dots \wedge T_n$.

Alcuni esempi di query congiuntive:

- “Trova tutti gli individui maschi.”

$$Male(x)$$

- “Chi sono gli individui con almeno un figlio maschio?”

$$y \text{ childOf } x \wedge Male(y)$$

- “Chi sono i figli di *Alice*?”

$$x \text{ childOf } Alice$$

con $x, y \in V$, $Male \in N_C$, $childOf \in N_P$ e $Alice \in N_I$.

Sostituzioni (1/2)

Per definire le *soluzioni* (risposte) delle query congiuntive introduciamo la nozione di *sostituzione*.

Una *sostituzione* $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ ($x_1, \dots, x_n \in V$, $a_1, \dots, a_n \in N_I$) è una mappa finita che associa nomi di individui a variabili.

Sia T una formula atomica e $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione.
L'*applicazione* $T\sigma$ di σ a T è la formula atomica che si ottiene sostituendo in T ad ogni occorrenza della variabile x_i il corrispondente nome di individuo a_i , per ogni $1 \leq i \leq n$.

Alcuni esempi:

$$\begin{aligned} \text{Male}(x)[x \rightarrow \text{Bob}] &= \text{Male}(\text{Bob}) \\ \text{Male}(x)[y \rightarrow \text{Bob}] &= \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}] &= \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}, y \rightarrow \text{Elise}] &= \end{aligned}$$

con $x, y \in V$, $\text{Male} \in N_C$, $\text{childOf} \in N_P$ e $\text{Alice}, \text{Bob}, \text{Elise} \in N_I$.

Sostituzioni (1/2)

Per definire le *soluzioni* (risposte) delle query congiuntive introduciamo la nozione di *sostituzione*.

Una *sostituzione* $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ ($x_1, \dots, x_n \in V$, $a_1, \dots, a_n \in N_I$) è una mappa finita che associa nomi di individui a variabili.

Sia T una formula atomica e $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione.
L'*applicazione* $T\sigma$ di σ a T è la formula atomica che si ottiene sostituendo in T ad ogni occorrenza della variabile x_i il corrispondente nome di individuo a_i , per ogni $1 \leq i \leq n$.

Alcuni esempi:

$$\begin{aligned} \text{Male}(x)[x \rightarrow \text{Bob}] &= \text{Male}(\text{Bob}) \\ \text{Male}(x)[y \rightarrow \text{Bob}] &= \text{Male}(x) \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}] &= \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}, y \rightarrow \text{Elise}] &= \end{aligned}$$

con $x, y \in V$, $\text{Male} \in N_C$, $\text{childOf} \in N_P$ e $\text{Alice}, \text{Bob}, \text{Elise} \in N_I$.

Sostituzioni (1/2)

Per definire le *soluzioni* (risposte) delle query congiuntive introduciamo la nozione di *sostituzione*.

Una *sostituzione* $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ ($x_1, \dots, x_n \in V$, $a_1, \dots, a_n \in N_I$) è una mappa finita che associa nomi di individui a variabili.

Sia T una formula atomica e $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione.
L'*applicazione* $T\sigma$ di σ a T è la formula atomica che si ottiene sostituendo in T ad ogni occorrenza della variabile x_i il corrispondente nome di individuo a_i , per ogni $1 \leq i \leq n$.

Alcuni esempi:

$$\begin{aligned} \text{Male}(x)[x \rightarrow \text{Bob}] &= \text{Male}(\text{Bob}) \\ \text{Male}(x)[y \rightarrow \text{Bob}] &= \text{Male}(x) \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}] &= \text{Alice childOf } y \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}, y \rightarrow \text{Elise}] &= \end{aligned}$$

con $x, y \in V$, $\text{Male} \in N_C$, $\text{childOf} \in N_P$ e $\text{Alice}, \text{Bob}, \text{Elise} \in N_I$.

Sostituzioni (1/2)

Per definire le *soluzioni* (risposte) delle query congiuntive introduciamo la nozione di *sostituzione*.

Una *sostituzione* $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ ($x_1, \dots, x_n \in V$, $a_1, \dots, a_n \in N_I$) è una mappa finita che associa nomi di individui a variabili.

Sia T una formula atomica e $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione.
L'*applicazione* $T\sigma$ di σ a T è la formula atomica che si ottiene sostituendo in T ad ogni occorrenza della variabile x_i il corrispondente nome di individuo a_i , per ogni $1 \leq i \leq n$.

Alcuni esempi:

$$\begin{aligned} \text{Male}(x)[x \rightarrow \text{Bob}] &= \text{Male}(\text{Bob}) \\ \text{Male}(x)[y \rightarrow \text{Bob}] &= \text{Male}(x) \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}] &= \text{Alice childOf } y \\ (x \text{ childOf } y)[x \rightarrow \text{Alice}, y \rightarrow \text{Elise}] &= \text{Alice childOf Elise} \end{aligned}$$

con $x, y \in V$, $\text{Male} \in N_C$, $\text{childOf} \in N_P$ e $\text{Alice}, \text{Bob}, \text{Elise} \in N_I$.

Sostituzioni (2/2)

L'applicazione di sostituzioni a query congiuntive si definisce come segue.

Sia $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione e siano T_1, \dots, T_m formule atomiche. Allora

$$(T_1 \wedge \dots \wedge T_m)\sigma =_{\text{Def}} T_1\sigma \wedge \dots \wedge T_m\sigma.$$

Alcuni esempi:

$$\begin{aligned} (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}] &=_{\text{Def}} \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, z \rightarrow \text{Bob}] &=_{\text{Def}} \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, y \rightarrow \text{Bob}] &=_{\text{Def}} \end{aligned}$$

Sostituzioni (2/2)

L'applicazione di sostituzioni a query congiuntive si definisce come segue.

Sia $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione e siano T_1, \dots, T_m formule atomiche. Allora

$$(T_1 \wedge \dots \wedge T_m)\sigma =_{\text{Def}} T_1\sigma \wedge \dots \wedge T_m\sigma.$$

Alcuni esempi:

$$\begin{array}{ll} (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}] & =_{\text{Def}} y \text{ childOf Alice} \wedge \text{Male}(y) \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, z \rightarrow \text{Bob}] & =_{\text{Def}} \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, y \rightarrow \text{Bob}] & =_{\text{Def}} \end{array}$$

Sostituzioni (2/2)

L'applicazione di sostituzioni a query congiuntive si definisce come segue.

Sia $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione e siano T_1, \dots, T_m formule atomiche. Allora

$$(T_1 \wedge \dots \wedge T_m)\sigma =_{\text{Def}} T_1\sigma \wedge \dots \wedge T_m\sigma.$$

Alcuni esempi:

$$\begin{array}{ll} (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}] & =_{\text{Def}} y \text{ childOf Alice} \wedge \text{Male}(y) \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, z \rightarrow \text{Bob}] & =_{\text{Def}} y \text{ childOf Alice} \wedge \text{Male}(y) \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, y \rightarrow \text{Bob}] & =_{\text{Def}} \end{array}$$

Sostituzioni (2/2)

L'applicazione di sostituzioni a query congiuntive si definisce come segue.

Sia $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione e siano T_1, \dots, T_m formule atomiche. Allora

$$(T_1 \wedge \dots \wedge T_m)\sigma =_{\text{Def}} T_1\sigma \wedge \dots \wedge T_m\sigma.$$

Alcuni esempi:

$$\begin{array}{ll} (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}] & =_{\text{Def}} y \text{ childOf Alice} \wedge \text{Male}(y) \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, z \rightarrow \text{Bob}] & =_{\text{Def}} y \text{ childOf Alice} \wedge \text{Male}(y) \\ (y \text{ childOf } x \wedge \text{Male}(y))[x \rightarrow \text{Alice}, y \rightarrow \text{Bob}] & =_{\text{Def}} \text{Bob childOf Alice} \wedge \text{Male}(\text{Bob}) \end{array}$$

Soluzioni per una Query

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se

$$\mathcal{O} \implies T_i \sigma$$

per ogni $1 \leq i \leq m$. In altre parole, $T_i \sigma$ deve essere in \mathcal{O} o deve essere una conseguenza di \mathcal{O} , per ogni T_i nella query.

Consideriamo l'ontologia \mathcal{O} e la query Q definite come segue:

$$\begin{aligned} \mathcal{O} = \quad & \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ & \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ & \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ & \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ & \text{Francis childOf Charlie} \} \end{aligned}$$

$$Q = \quad y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_1 = [x \rightarrow \text{Alice}, y \rightarrow \text{Daniel}]$. σ_1 è una soluzione per Q rispetto ad \mathcal{O} ?

Soluzioni per una Query - Esempio 1

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q ("Chi sono gli individui con almeno un figlio maschio?") definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \mathbf{\text{Male}(\text{Daniel})}, \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \mathbf{\text{Daniel childOf Alice}}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$

$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_1 = [x \rightarrow \text{Alice}, y \rightarrow \text{Daniel}]$. σ_1 è una soluzione per Q rispetto ad \mathcal{O} ? **SI**.

$$Q\sigma_1 = \text{Daniel childOf Alice} \wedge \text{Male}(\text{Daniel}).$$

Soluzioni per una Query - Esempio 2

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_2 = [x \rightarrow \text{Alice}, y \rightarrow \text{Bob}]$. σ_2 è una soluzione per Q rispetto ad \mathcal{O} ?

Soluzioni per una Query - Esempio 2

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q (“Chi sono gli individui con almeno un figlio maschio?”) definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_2 = [x \rightarrow \text{Alice}, y \rightarrow \text{Bob}]$. σ_2 è una soluzione per Q rispetto ad \mathcal{O} ? **NO**.

$$Q\sigma_2 = \text{Bob childOf Alice} \wedge \text{Male}(\text{Bob}).$$

Soluzioni per una Query - Esempio 3

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_3 = [x \rightarrow \text{Charlie}, y \rightarrow \text{Francis}]$. σ_3 è una soluzione per Q rispetto ad \mathcal{O} ?

Soluzioni per una Query - Esempio 3

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q (“Chi sono gli individui con almeno un figlio maschio?”) definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$

$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_3 = [x \rightarrow \text{Charlie}, y \rightarrow \text{Francis}]$. σ_2 è una soluzione per Q rispetto ad \mathcal{O} ? **NO**.

$$Q\sigma_3 = \text{Francis childOf Charlie} \wedge \text{Male}(\text{Francis}).$$

Soluzioni per una Query - Esempio 4

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_4 = [y \rightarrow \text{Daniel}]$. σ_4 è una soluzione per Q rispetto ad \mathcal{O} ?

Soluzioni per una Query - Esempio 4

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q ("Chi sono gli individui con almeno un figlio maschio?") definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$

$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_4 = [y \rightarrow \text{Daniel}]$. σ_2 è una soluzione per Q rispetto ad \mathcal{O} ? **NO**.

$$Q\sigma_4 = \text{Daniel childOf } x \wedge \text{Male}(\text{Daniel}).$$

Affinchè una sostituzione σ sia una soluzione per una query Q (a prescindere dall'ontologia) è necessario che in σ compaiano tutte le variabili di Q .

Soluzioni per una Query - Esempio 5

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_5 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}, z \rightarrow \text{Francis}]$. σ_5 è una soluzione per Q rispetto ad \mathcal{O} ?

Soluzioni per una Query - Esempio 5

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, $Q = T_1 \wedge \dots \wedge T_m$ una query congiuntiva e \mathcal{O} una ontologia.

σ è detta essere una *soluzione* per Q rispetto ad \mathcal{O} se e solo se $T_1\sigma, \dots, T_m\sigma$ compaiono in \mathcal{O} .

Consideriamo l'ontologia \mathcal{O} e la query Q ("Chi sono gli individui con almeno un figlio maschio?") definite come segue:

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

Sia $\sigma_5 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}, z \rightarrow \text{Francis}]$. σ_5 è una soluzione per Q rispetto ad \mathcal{O} ? **SI**.

$$Q\sigma_5 = \text{Charlie childOf Elise} \wedge \text{Male}(\text{Charlie}).$$

Soluzioni Minimali per una Query

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, Q una query congiuntiva e \mathcal{O} una ontologia.

σ è una *soluzione minimale* per Q rispetto a \mathcal{O} se e solo se:

- 1 σ è una soluzione per Q rispetto ad \mathcal{O} e inoltre
- 2 tutte le variabili x_1, \dots, x_n che compaiono in σ compaiono anche in Q (criterio di minimalità).

Consideriamo ad esempio

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$
$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

$\sigma_5 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}, z \rightarrow \text{Francis}]$ è una soluzione minimale per Q rispetto a \mathcal{O} ?

Soluzioni Minimali per una Query

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, Q una query congiuntiva e \mathcal{O} una ontologia.

σ è una *soluzione minimale* per Q rispetto a \mathcal{O} se e solo se:

- ① σ è una soluzione per Q rispetto ad \mathcal{O} e inoltre
- ② tutte le variabili x_1, \dots, x_n che compaiono in σ compaiono anche in Q (criterio di minimalità).

Consideriamo ad esempio

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$

$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

$\sigma_5 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}, z \rightarrow \text{Francis}]$ è una soluzione minimale per Q rispetto a \mathcal{O} ? **NO**.

$\sigma_6 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}]$ è una soluzione minimale per Q rispetto a \mathcal{O} ? .

Soluzioni Minimali per una Query

Siano $\sigma = [x_1 \rightarrow a_1, \dots, x_n \rightarrow a_n]$ una sostituzione, Q una query congiuntiva e \mathcal{O} una ontologia.

σ è una *soluzione minimale* per Q rispetto a \mathcal{O} se e solo se:

- ① σ è una soluzione per Q rispetto ad \mathcal{O} e inoltre
- ② tutte le variabili x_1, \dots, x_n che compaiono in σ compaiono anche in Q (criterio di minimalità).

Consideriamo ad esempio

$$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \\ \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \\ \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \\ \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \\ \text{Francis childOf Charlie} \}$$

$$Q = y \text{ childOf } x \wedge \text{Male}(y).$$

$\sigma_5 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}, z \rightarrow \text{Francis}]$ è una soluzione minimale per Q rispetto a \mathcal{O} ? **NO**.

$\sigma_6 = [x \rightarrow \text{Elise}, y \rightarrow \text{Charlie}]$ è una soluzione minimale per Q rispetto a \mathcal{O} ? **SI**.

Conjunctive Query Answering

Il problema del *Conjunctive Query Answering* consiste nel trovare tutte le soluzioni minimali di una query congiuntiva rispetto ad una ontologia.

Esse sono sempre in numero finito, infatti:

- le variabili che compaiono nelle soluzioni sono esattamente quelle che compaiono nella query,
- i nomi di individui che compaiono nelle soluzioni sono un sottoinsieme di quelli che compaiono nell'ontologia.

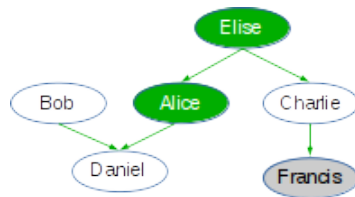
Conjunctive Query Answering

Il problema del *Conjunctive Query Answering* consiste nel trovare tutte le soluzioni minimali di una query congiuntiva rispetto ad una ontologia.

Esempio 1: "Trova tutti gli individui maschi."

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = \text{Male}(x)$



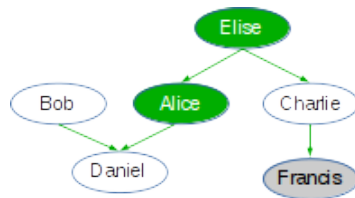
x
Bob
Charlie
Daniel

Conjunctive Query Answering - Esempio 2

“Chi sono gli individui con almeno un figlio maschio?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = y \text{ childOf } x \wedge \text{Male}(y)$



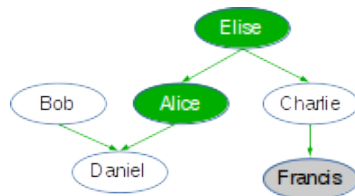
x	y
<i>Elise</i>	<i>Charlie</i>
<i>Alice</i>	<i>Daniel</i>
<i>Bob</i>	<i>Daniel</i>

Conjunctive Query Answering - Esempio 3

“Chi sono i figli di *Elise*?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \text{Francis childOf Charlie} \}$

$Q = x \text{ childOf Elise}$



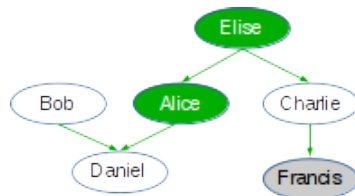
x

Conjunctive Query Answering - Esempio 3

“Chi sono i figli di *Elise*?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}), \text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}), \text{Alice childOf Elise}, \text{Charlie childOf Elise}, \text{Daniel childOf Alice}, \text{Daniel childOf Bob}, \text{Francis childOf Charlie} \}$

$Q = x \text{ childOf Elise}$

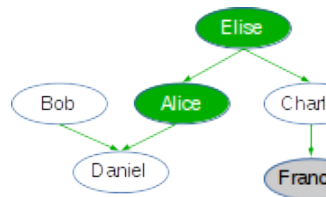


x
Alice
Charlie

Conjunctive Query Answering - Esempio 4

“Chi sono gli individui con almeno un figlio maschio ed una femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$



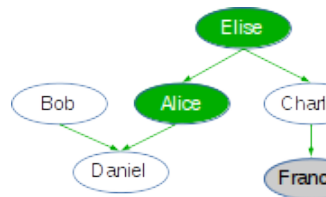
x	y	z

Conjunctive Query Answering - Esempio 4

“Chi sono gli individui con almeno un figlio maschio ed una femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = y \text{ childOf } x \wedge z \text{ childOf } x \wedge \text{Male}(y) \wedge \text{Female}(z)$



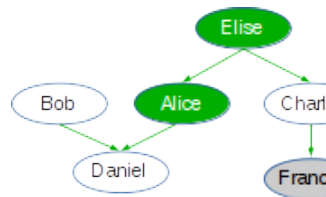
x	y	z

Conjunctive Query Answering - Esempio 4

“Chi sono gli individui con almeno un figlio maschio ed una femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = y \text{ childOf } x \wedge z \text{ childOf } x \wedge \text{Male}(y) \wedge \text{Female}(z)$

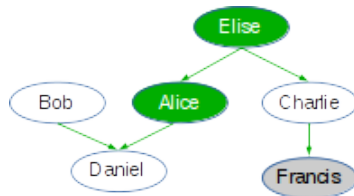


x	y	z
Elise	Charlie	Alice

Conjunctive Query Answering - Esempio 5

“Chi sono gli individui maschi con almeno una figlia femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$



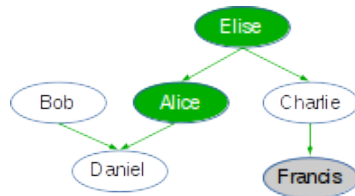
x	y

Conjunctive Query Answering - Esempio 5

“Chi sono gli individui maschi con almeno una figlia femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = \text{Male}(x) \wedge y \text{ childOf } x \wedge \text{Female}(y)$



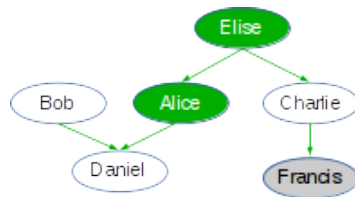
x	y

Conjunctive Query Answering - Esempio 5

“Chi sono gli individui maschi con almeno una figlia femmina?”

$\mathcal{O} = \{ \text{Female}(\text{Elise}), \text{Female}(\text{Alice}), \text{Male}(\text{Bob}),$
 $\text{Male}(\text{Charlie}), \text{Male}(\text{Daniel}),$
 $\text{Alice childOf Elise}, \text{Charlie childOf Elise},$
 $\text{Daniel childOf Alice}, \text{Daniel childOf Bob},$
 $\text{Francis childOf Charlie} \}$

$Q = \text{Male}(x) \wedge y \text{ childOf } x \wedge \text{Female}(y)$



Nessuna soluzione