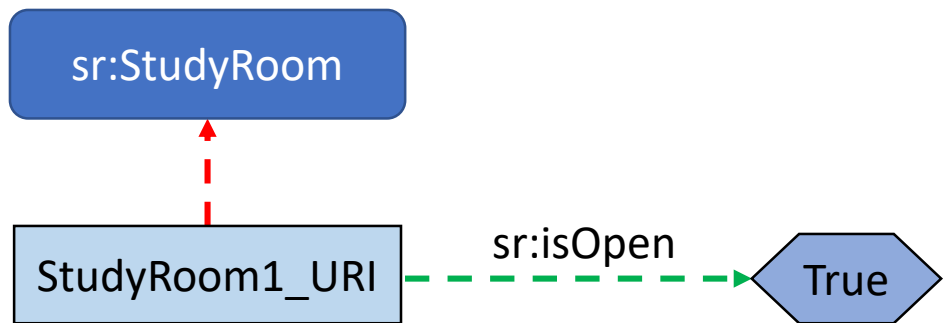
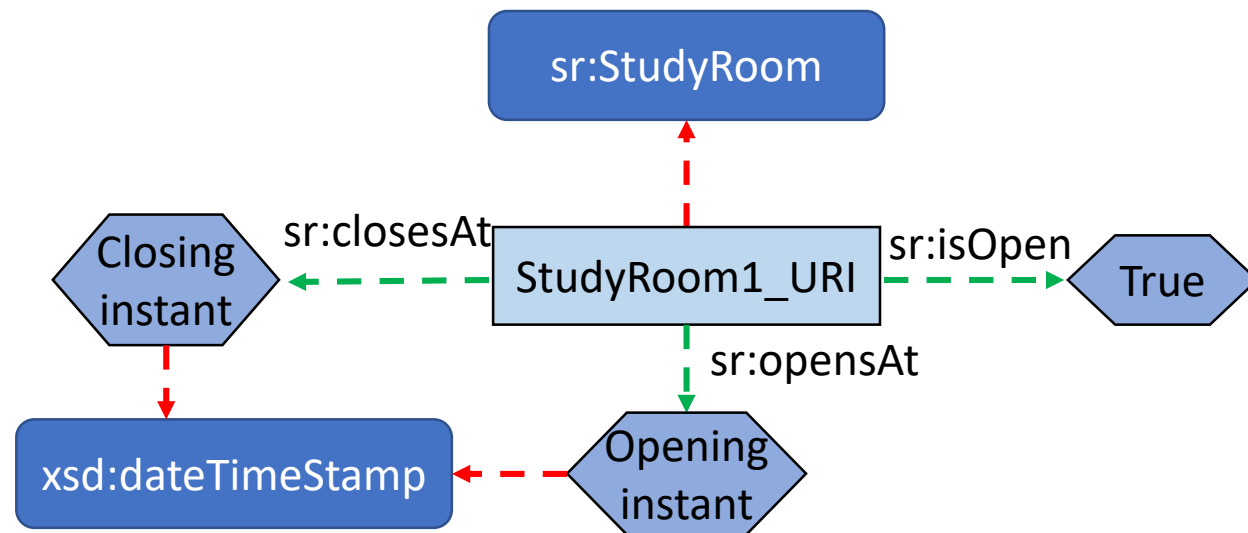


Aggiunta orari di chiusura e apertura

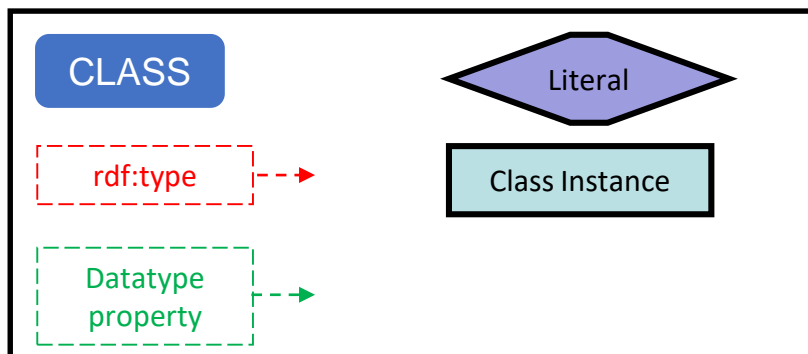
Prima



Dopo



Legenda



Spiegazione

Aggiunge closesAt che dice quando l'aula studio verrà chiusa. Fa anche comodo sapere quando verrà aperta, quindi c'è opensAt. Es. all'utente che fa una query sono restituite anche le aule che aprono tra 15 minuti. Entrambe le relazioni hanno come range il built-in xsd:dateTimeStamp, che rappresenta un'ora, una data ed una timezone. Nota: ci sono dei vincoli su queste relazioni: closesAt dovrà essere presente (o essere considerata) solo se isOpen è true, mentre opensAt quando isOpen è false. Questo vincolo è facile da far rispettare a livello applicativo: viene imposto dal controller di aula studio.

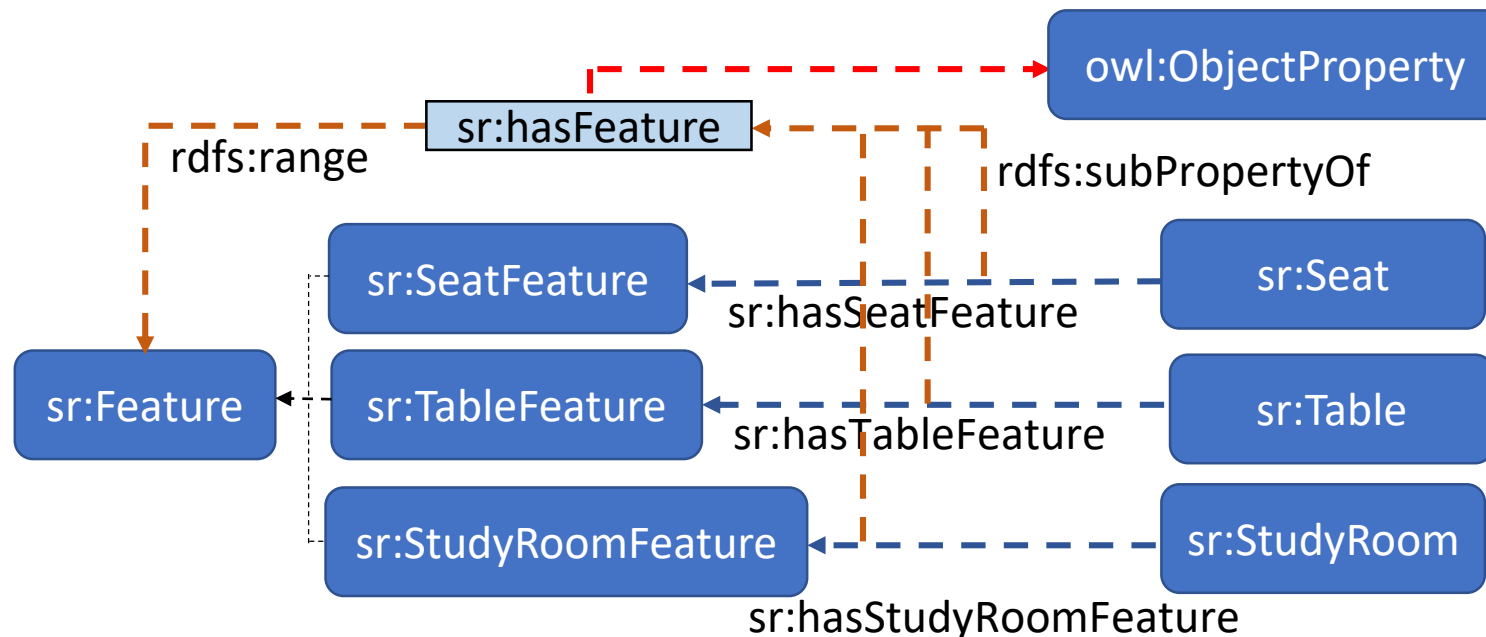
Modellazione diversi tipi di features

Prima

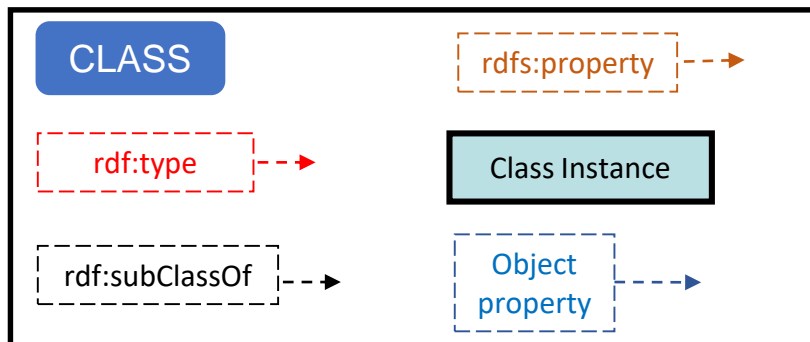


A livello applicativo, solo le istanze di sr:StudyRoom, sr:Table ed sr:Seat sono in relazione a sr:hasFeature. Inoltre, istanze di sr:Feature associate ad una di queste classi non sono associate alle altre due. Ma questi vincoli non appaiono nell'ontologia.

Dopo



Legenda



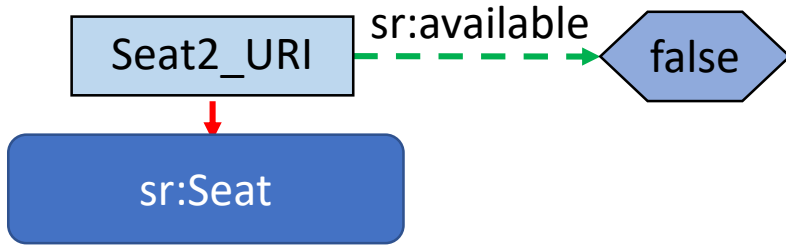
Spiegazione

Sono introdotti a livello di ontologia i vincoli tra features e «possessori di features» (Seat, Table e StudyRoom). In particolare:

- c'è una sottoclasse di Feature per ogni tipo di feature
- c'è una sottoproperty di hasFeature per ogni tipo di feature (e corrispondente «possessore di features»).

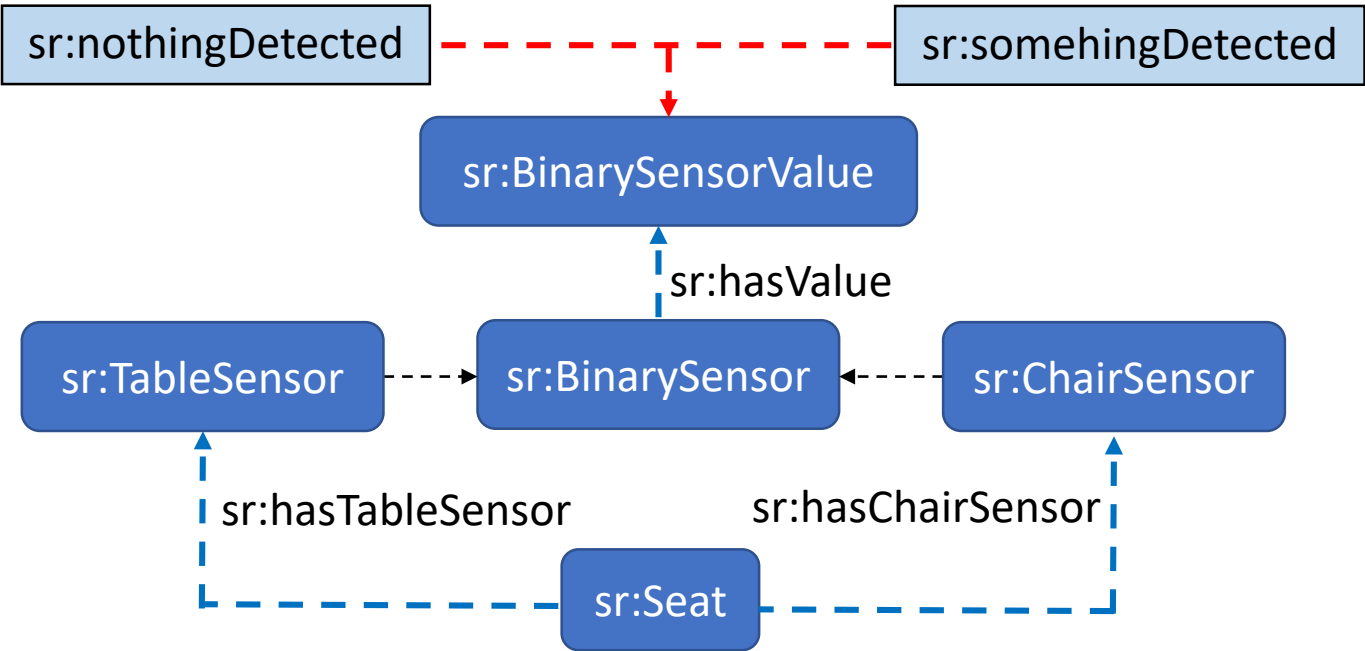
Modellazione sensori nell'ontologia

Prima

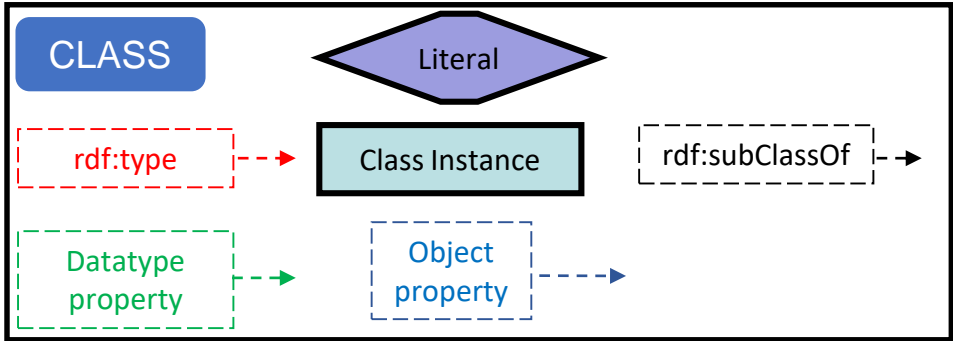


I sensori non sono modellati nell'ontologia, sono conosciuti solo dal KP in aula studio.

Dopo



Legenda

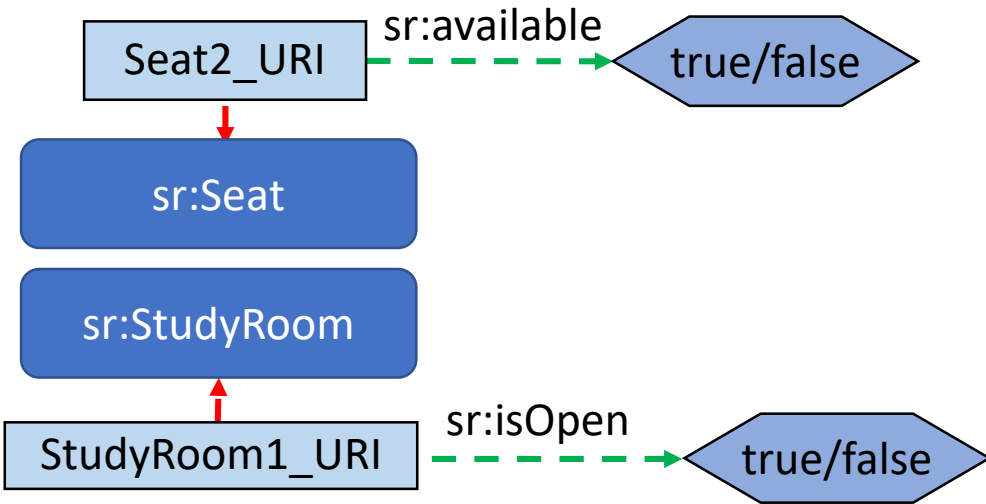


Spiegazione

Ogni Seat è legato ad un TableSensor ed un ChairSensor, sottoclassi di BinarySensor. Un BinarySensor ha sempre un BinarySensorValue, che può avere due sole istanze: somethingDetected (il sensore rileva la presenza di qualcosa) e nothingDetected (il sensore non rileva niente).

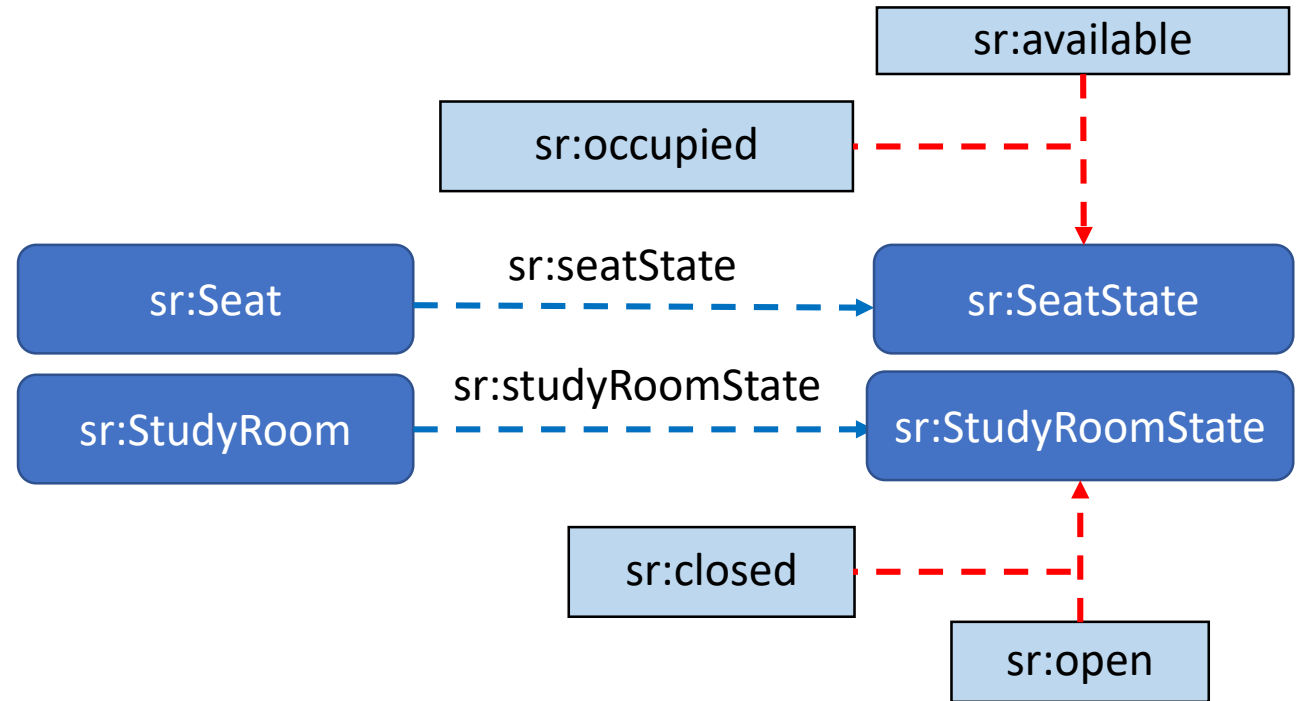
Modellazione stati di Seat e StudyRoom
(open/closed e occupied/available)

Prima

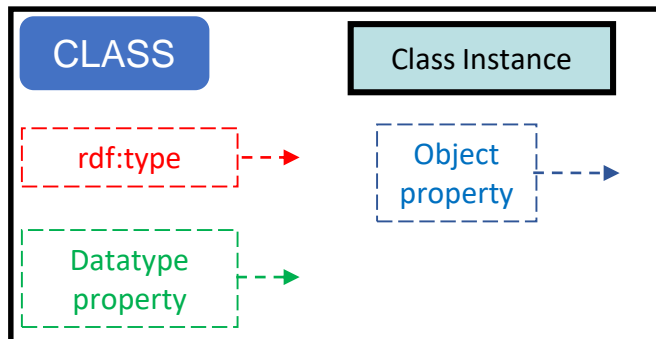


Lo stato di apertura di una StudyRoom e di disponibilità di un Seat sono modellati come dei xsd:boolean.

Dopo



Legenda



Spiegazione

Seat e StudyRoom sono caratterizzati da un seatState (available/occupied) ed un studyRoomState (open/closed) rispettivamente. Tali entità ne modellano lo stato corrente.