6.2 protégé本体工具

本节以一个空间目标识别本体的应用为例，说明protégé本体工具实现本体建模的主要功能。

6.2.1浏览本体

在protégé本体工具中打开本体文件后，就可以对文件进行浏览。

1. 浏览类概念

浏览类概念主要包括如下步骤。

第一步，打开classes标签界面，在左侧的Class hierarchy视窗内可以看到构建好的类概念，通过点击每个类左侧的小三角形符号，可以展开该类的子类概念。

第二步，单击类概念，可以看到该界面右侧上方的Annotations视窗内构建的对类概念的注释，包括了概念的内容、标注等信息。



图2.3 浏览OntoStar 2.01中的空间目标类

第三步，通过下拉按钮，可以看到该界面右侧下方的Description视窗内类概念的说明，包含了等价类、所有约束、匿名类、所有类概念实例、不相交公理等。



图2.4 浏览空间目标类的Description视窗

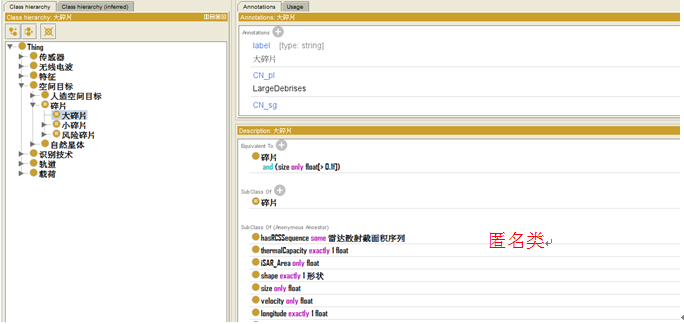


图2.5 浏览大碎片类概念的匿名类

如图2.5所示匿名类包含了大碎片类继承得到的所有约束，其中有它继承自父类碎片的所有约束，也有碎片继承自空间目标的所有约束。

2. 浏览属性

在protégé本体中，类概念的属性分为对象属性（Object properties）和数值属性（Data properties）。浏览对象属性主要包括如下步骤。

第一步，打开Object Properties标签界面，在左侧的Object properties hierarchy视窗中，点击顶层对象属性左边的小三角形标记，将对象属性概念展开，可以看到一个本体文件中的全部对象属性。

第二步，单击其中一个对象属性，例如图2.6所示的inOrbit，可以看到屏幕右侧视窗出现了该属性的注释、描述和约束。

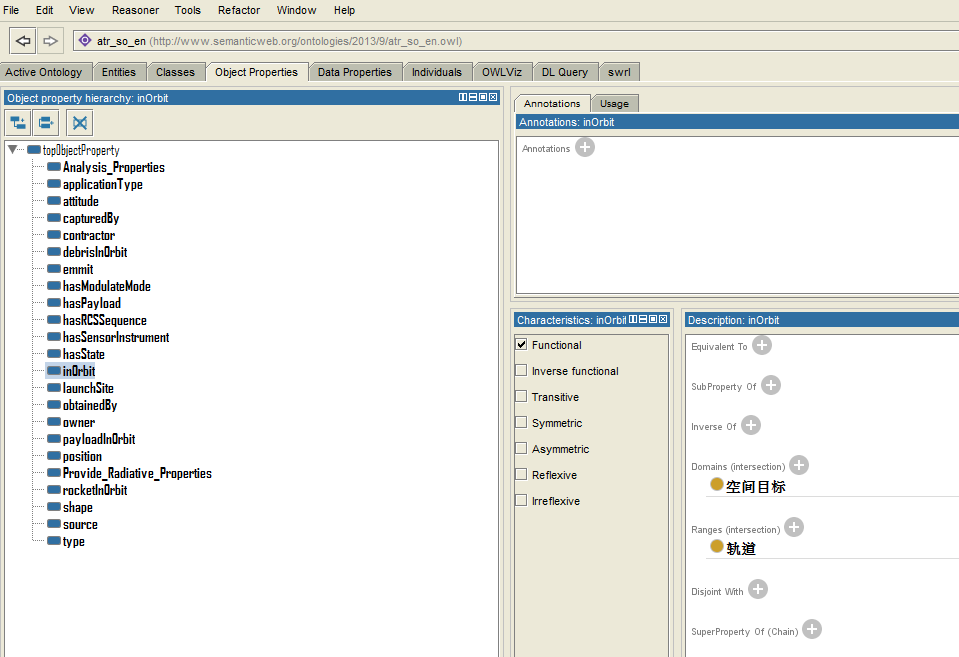


图2.6 对象属性inOrbit的描述与约束

在图2.6中，Description视窗显示了inOrbit对象属性定义的定义域和作用域，Characteristics视窗描述了inOrbit对象属性的约束，包括Functional、Inverse functional、Transitive、Symmetric、Asymmetric、Reflexive、Irreflexive等。这里标明了Functional约束，定义域为空间目标，作用域为轨道，说明一个空间目标的实例通过inOrbit所连接的轨道实例最多只有一个，即若同一个空间目标实例如果通过inOrbit连接了两个轨道实例，那么说明这两个轨道实例是同一个轨道实例，如果实际上这两个实例并不相同，那就出现了不一致情况。

浏览数值属性主要包括如下步骤。

第一步，打开Data Properties标签界面，在左侧的Data property hierarchy视窗中可以看到OntoStar中的全部数值属性。

第二步，单击其中一个数据属性，例如单击图2.7 数值属性altitude，在右侧视窗中显示了altitude属性的注释、描述和约束。其中，altitude的定义域为轨道，值域为float，具有Functional约束，表示任意一个轨道实例的高度只能有一个浮点数值，若一个轨道实例有了两个不同的altitude值，就出现不一致。

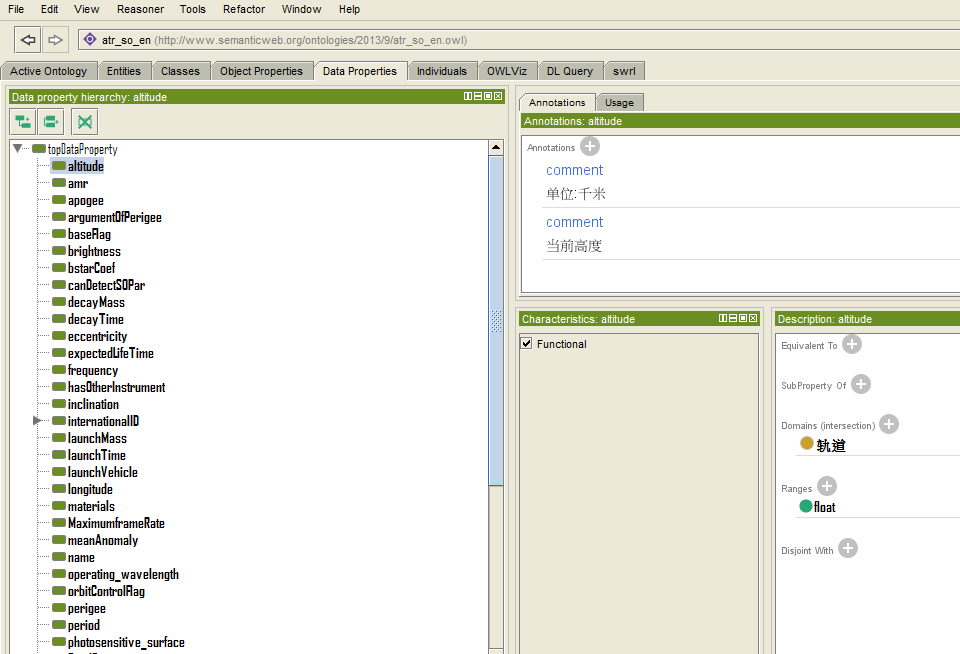


图2.7 数值属性altitude的注释、描述和约束

3.浏览SWRL规则

浏览本体中的SWRL规则主要包括如下步骤。。

第一步，点击菜单栏中的Window，在下拉的菜单项中，点击Tabs，然后选择swrl，标签栏里就会出现swrl标签，点击该标签，图2.8所示为swrl标签初始空界面。

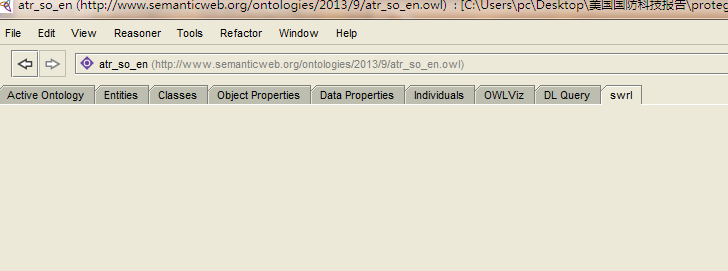


图2.8 swrl标签初始空界面

第二步，点击菜单栏中的Window，在下拉菜单项中，点击Views，然后点击Ontology views，选中最下面的Rules，如图2.9所示为本体中的swrl规则。

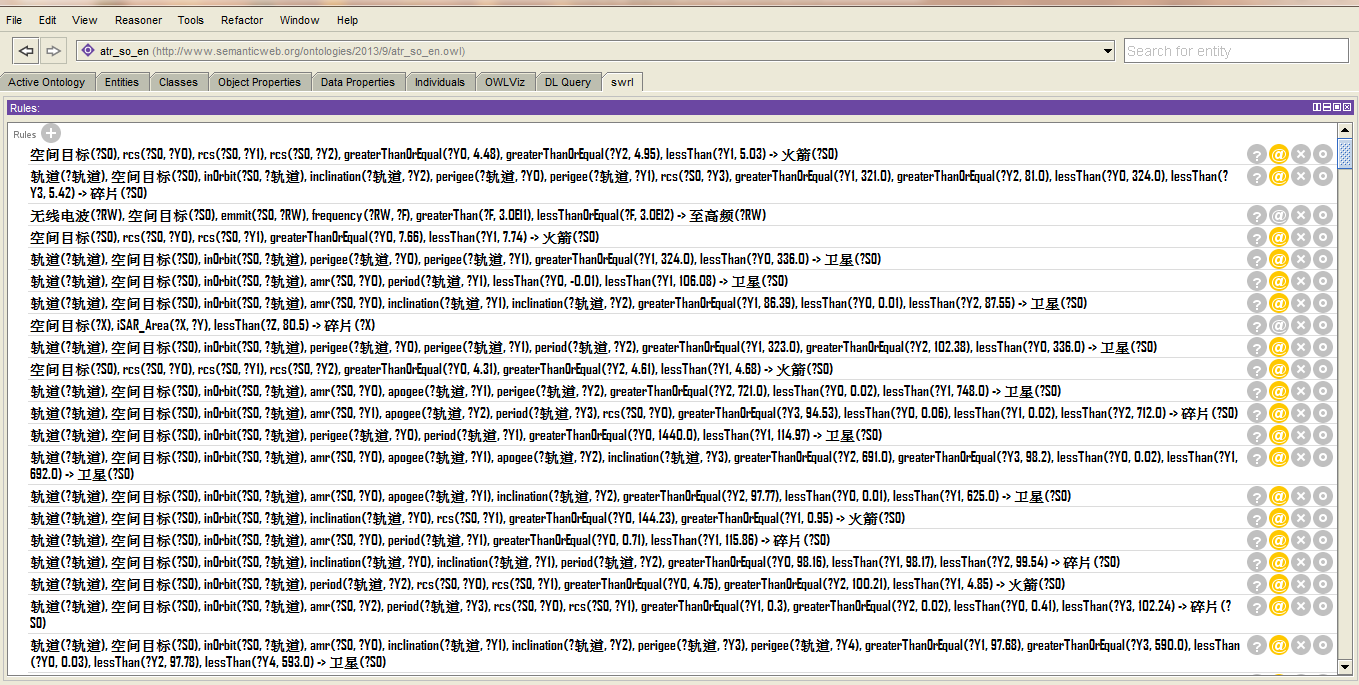


图2.9本体中的swrl规则

4.浏览实例

首先是总览所有实例，可按照如下步骤进行。

第一步，打开Individuals标签界面，在protégé中，与实例相关的视窗为紫色，在其中的Individuals视窗内，可以看到本体的全部实例。

第二步，点击任意一个紫色菱形后的空间目标实例名，如1960-013B实例，在右侧的视窗中会显示其注释、描述和断言。Description视窗是描述视窗，显示了该空间目标实例所定义的类是空间目标和火箭；Property Assertions视窗为断言视窗，显示了1960-013B实例的属性约束有来源于US，在12186565号轨道上等。

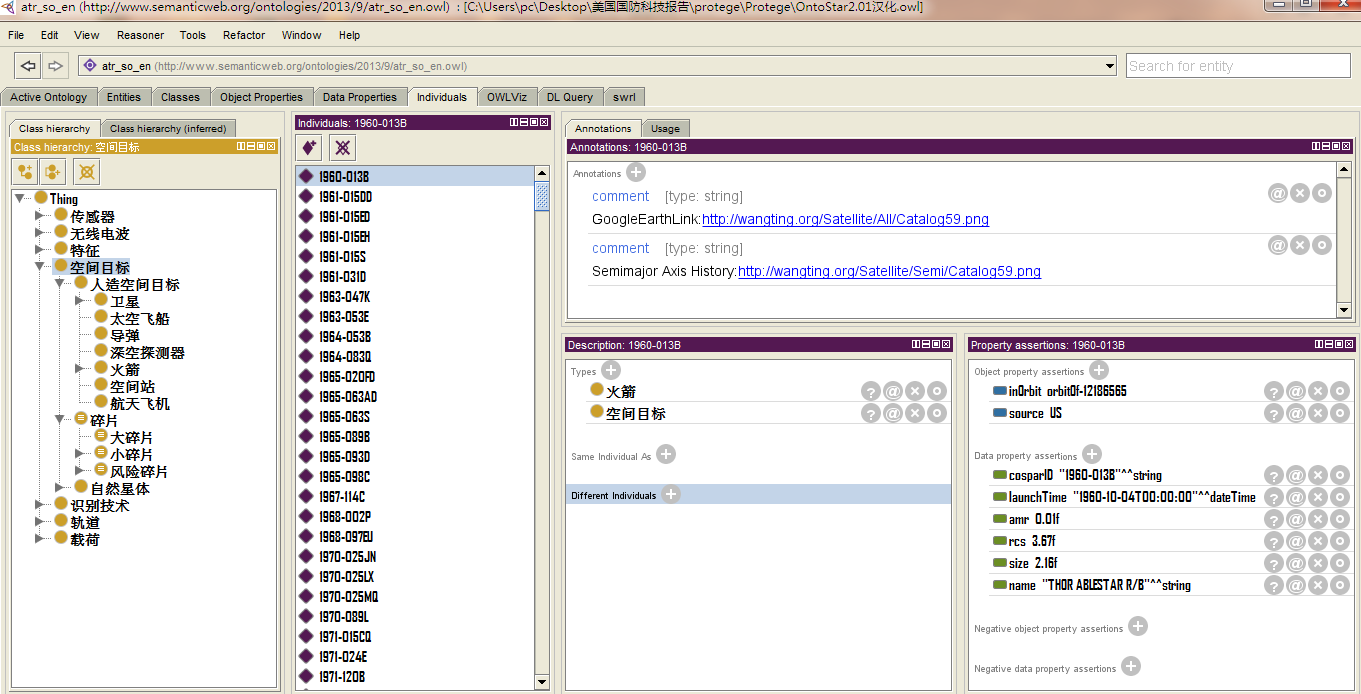


图2.10 实例总览与单个实例浏览

另外，还可以通过类概念来浏览实例，查看一类空间目标下的所有实例，具体步骤如下所示。

第一步，点击菜单栏中的Window，点击Views项，再点击Class views内的Description项，在屏幕中间建立类概念的Description视窗，如图2.11所示。

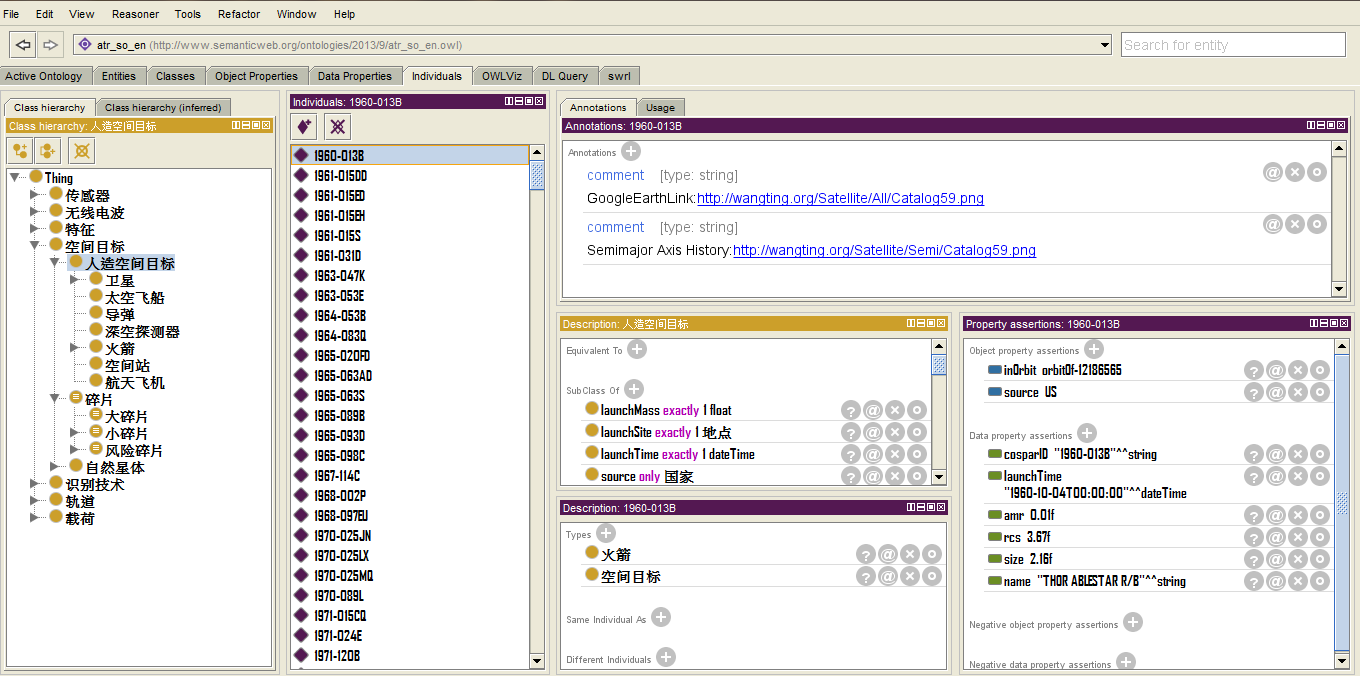


图2.11 在Individuals标签界面内增加Class的Description视窗

第二步，在Class hierarchy视窗内，选择卫星类概念，再将鼠标放到在Description视窗，下拉滑轮，可以看到卫星类下的所有实例。

第三步，点击其中一个实例，如2012-066C实例，可以看到其他视窗内显示出了该实例的分类与属性约束，如图2.12所示。

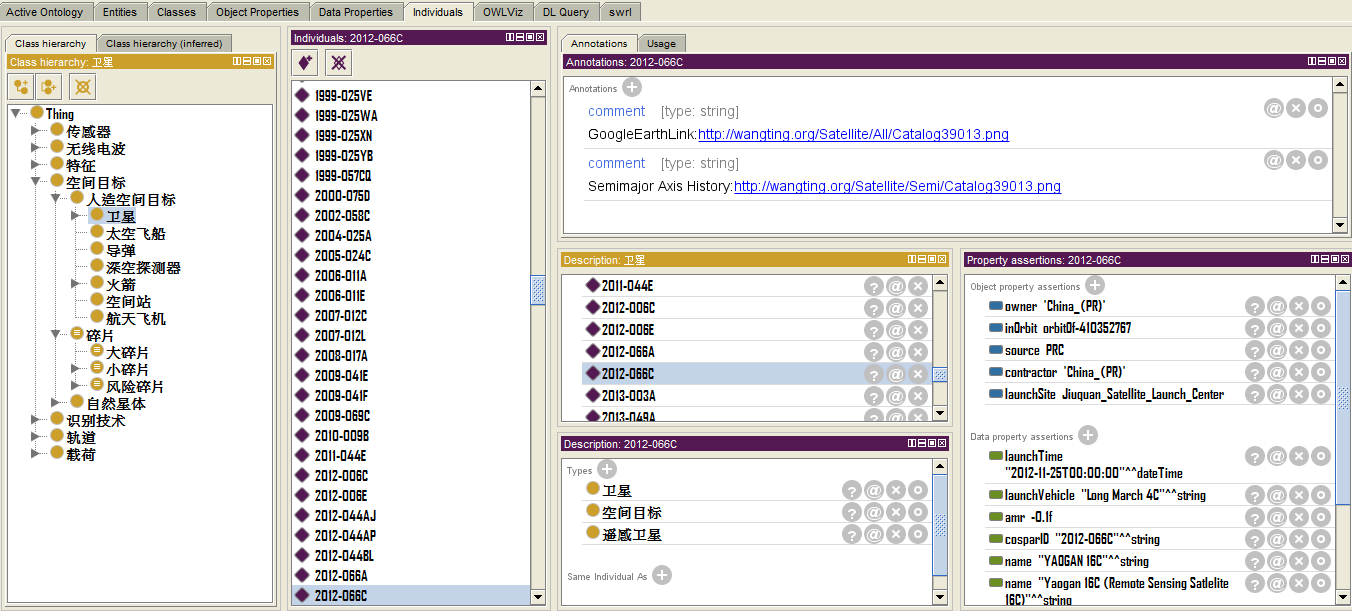


图2.12 通过类概念浏览实例

5.存储OntoStar的文件

使用protégé本体工具保存OntoStar文件，可按一下步骤进行。

第一步，点击菜单栏中的File按钮，选择Save as项，会弹出选择保存格式对话框，如图2.13所示。

第二步，选择一个本体保存格式，点击确定即可。

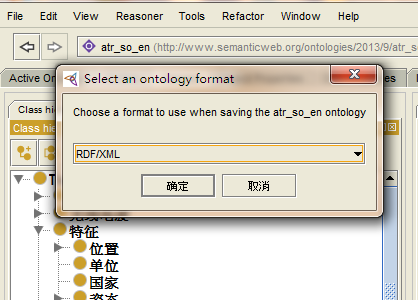


图2.13 OntoStar本体文件保存

### 6.2.2 本体建模

使用protégé工具，可以方便、快捷的构建本体知识。当需要修改本体内容时，可通过删除旧知识，增加新知识实现。

1.增加新的类概念

当应用中出现新的概念，需要添加到本体中，可通过如下步骤完成。

第一步， 选中class标签，在出现的界面内，展开原有的本体，确定要添加的类概念在本体概念体系中父类节点的位置。

第二步，点击屏幕左侧如图所示的创建子类图标，在弹出的对话框中输入要添加的新的类概念名称，注意不能使用中文且首字母必须大写。例如，我们新增一个新型中继卫星类，如图3.1所示。

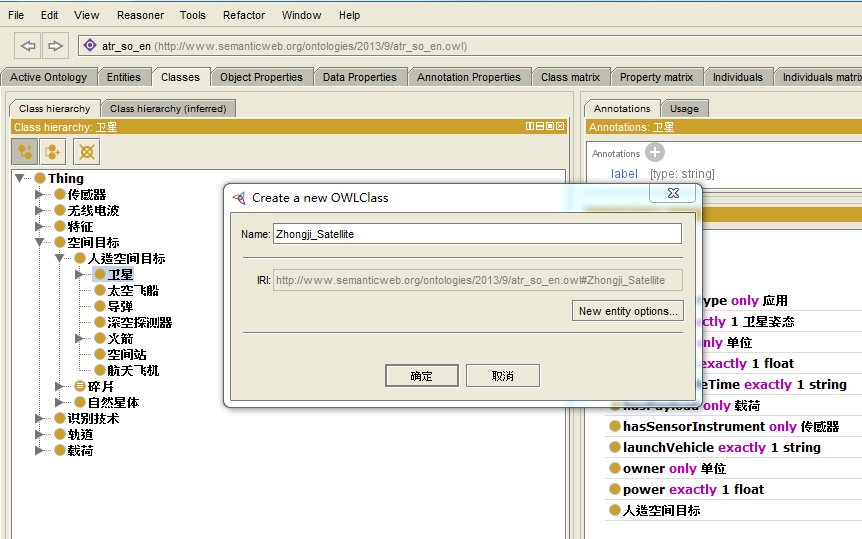


图3.1 在卫星类下添加中继卫星类

第三步，在屏幕右侧上方的Annotations视窗里可以输入新类的注释，注释文本可以用中文表示，包括版本号、内容、标注等说明。

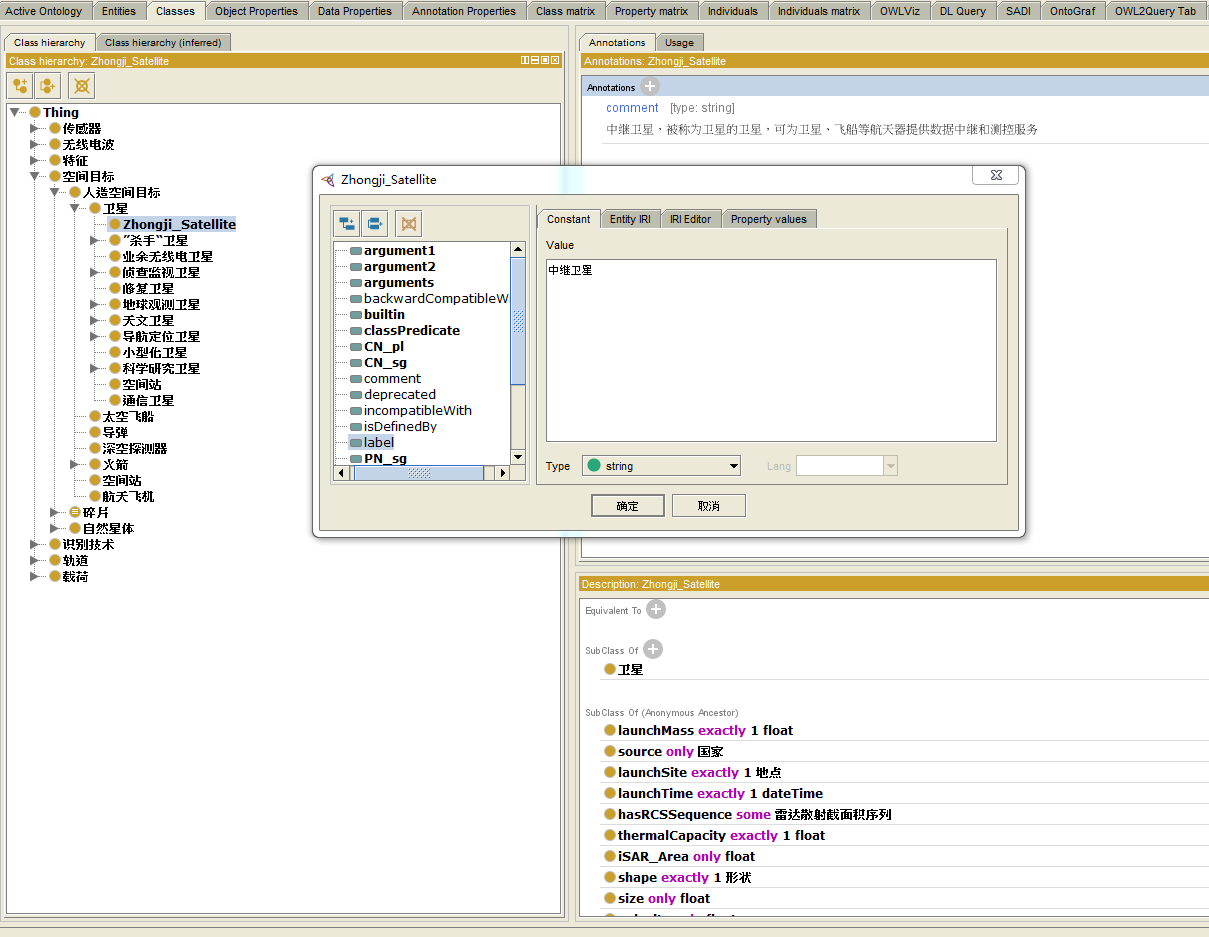


图3.2 增加类概念中继卫星的注释

第四步，在屏幕右侧下方的Description视窗里可以输入对新类的描述，这些描述主要包括类的定义（即等价类）、该类应满足的必要约束条件（即超类与类属性约束条件、该类的实例（即成员定义）、与哪些概念不相交（即不相交类的定义）等说明。

要修改类的层次时，可以直接按住一个类，将它拖动到其他类中，成为其他类的子类，拖动过程会将整个类包括它的子类一起，类似于文件夹的移动。在protégé工具中，一个类的子类会自动继承父类所具备的属性约束。

2.增加新的属性

对象的属性包括数值属性和对象属性。

数值属性（Data properties）描述了类概念内在固有的属性，能够量化表示为数值，如长度、面积、周期、功率等。增加数值属性的主要步骤如下所示。

第一步，打开Data Properties标签界面，增加新的数值属性需要在Data Properties界面里进行，如图2.7所示。

第二步，在屏幕最左侧的DataPropertyHierarchy视窗里，点击增加属性按钮，即可通过对话框输入新数据属性的名称。

第三步，在屏幕右侧上方的Annotations视窗里可以输入新属性的描述，描述文本可以用中文表示，包括版本号、内容、标注等说明。可以进行数值属性的创建与约束。

第四步，在屏幕右侧左下方的Characteristics视窗里可以对新属性进行约束，这里的约束主要说明该属性是否为函数属性。函数属性是指对象的该属性值是唯一的。

第五步，在屏幕右侧右下方的Descriptions视窗里可以对新属性进行约束，这里的描述主要包括新属性的形式定义、子属性、该属性的定义域和值域、与该属性不相交的属性等。

第六步，打开class标签界面，在屏幕右侧下方的Description视窗里修改该类针对新属性应满足的必要约束条件，即超类与类属性约束条件。

（2）对象属性（Object properties）描述类与类之间的关系，如空间目标类与特征类之间可以用has关联起来，表示一个二元关系。要注意，protégé里不能直接表示三元关系，可以通过另外构建一个概念，分别与三个类两两关联。如已有卫星、传感器、亮度特征的类概念，想表示传感器获取卫星的亮度特征时，可以新建为亮度特征类建造两个新的对象属性，capturedby和hasfeature，用capturedby建立亮度特征和传感器之间的关联关系，用hasfeature建立卫星和亮度特征之间的关联关系，由此建立这三者之间的三元关系。

新增对象属性的构建过程与新增数据属性相似，首先选择Object properties界面，步骤如下。

第一步，打开Object properties标签界面，增加新的对象属性需要在Object properties界面里进行，如图3.3所示。

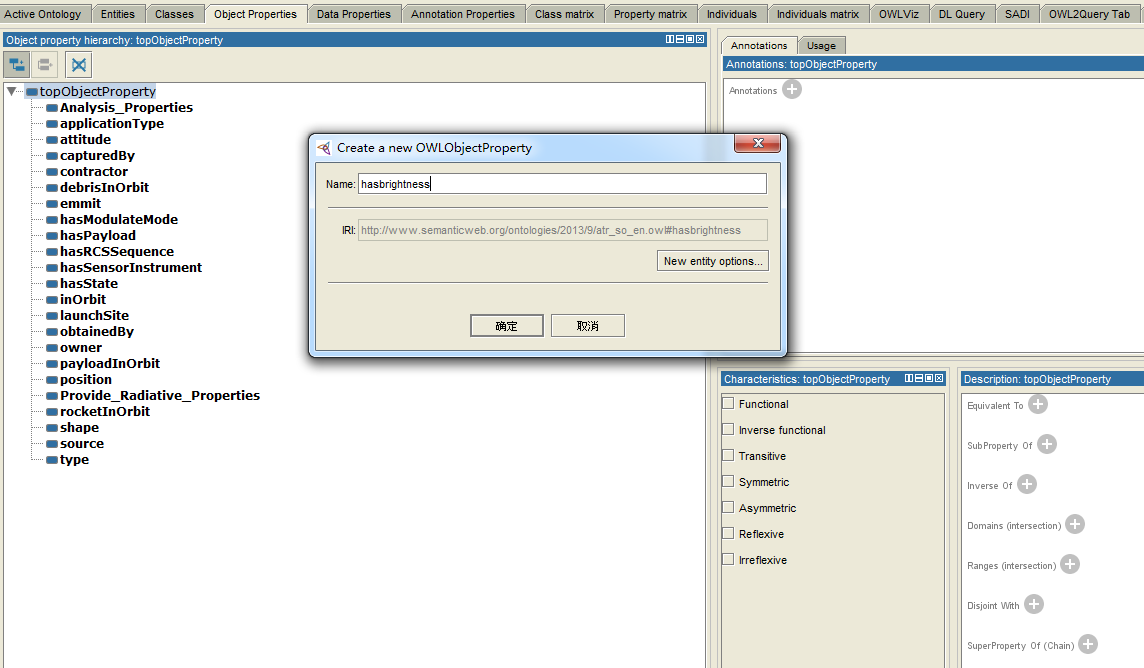


图3.3 增加hasfeature对象属性

第二步，在屏幕最左侧的ObjectPropertyHierarchy视窗里，点击创建对象属性按钮，在对话框里输入新增对象属性名称即可，这里对大小写没有限制，但是不能为中文，否则易出现乱码。

第三步，在屏幕右侧上方的Annotations视窗里可以输入新属性的描述，描述文本可以用中文表示，包括版本号、内容、标注等说明。可以进行数值属性的创建与约束。

第四步，在屏幕右侧左下方的Characteristics视窗里可以对新属性进行约束，这里的约束主要说明该属性应满足的性质，这些性质包括函数属性、逆函数属性、传递性、对称性、反对称性、自反性、非自反性。函数属性是指对象的该属性值是唯一的；逆函数属性是说明属性是否存在逆关系；传递性说明该对象属性关系是否为传递关系；对称性是指该对象属性关系存在对称性，例如结婚关系，A与B结婚，则B也一定与A结婚；反对称性是指该对象属性关系不存在对称性，例如父子关系，A与B满足父子关系，则B与A一定不存在父子关系；自反性是指该对象属性关系允许出现个体与自己产生关系；非自反性是指该对象属性关系不允许出现个体与自己产生关系。

第五步，在屏幕右侧右下方的Descriptions视窗里可以对新属性进行约束，这里的描述主要包括新属性的形式定义、逆属性、子属性、该属性的定义域和值域、与该属性不相交的属性、父属性等。

第六步，若新属性的值域是当前本体中没有的新概念，则需要增加新的类概念（参见6.2.2 1.增加新的类概念）。

第七步，打开class标签界面，在屏幕右侧下方的Description视窗里修改该类针对新属性应满足的必要约束条件，即超类与类属性约束条件。

3.增加新的基本公理与约束

在protégé工具中，除了定义类、属性所涉及的相关公理之外，本体还需要针对如下公理和约束进行定义，这些公理和约束可能会影响推理机求解问题的能力或效果。

（1）等价类公理。

在类概念定义过程中，class界面中屏幕右侧下方的Description视窗里，可定义等价类描述。这里的等价类，并非仅仅是与该类等价的其他类，也可以通过表达式定义一个类（即等价类公理）。通过等价类公理定义的类称作定义类（defined class）。等价类定义形成该类对象要满足的充分必要条件。

当需要增加等价类公理时，在class界面内，选中我们要进行约束的一个类名，在屏幕右侧下方的Description视窗里，选中第一个Equivalent项，则可以添加该类的等价类定义。

如图3.4所示，类SHF\_Space\_Research\_RW表示超高频的空间研究业务无线电波，右侧的等价类表示它所描述的概念首先是空间研究业务无线电波，并且频率范围确定为浮点数的(8.4E9f，8.5E9f]，…，(1.275E10f，1.43E10f]，......等频段中（其中，8.5E9f表示科学计数法，即）。我们可以通过或关系来定义SHF\_Space\_Research\_RW类，从而表示了SHF\_Space\_Research\_RW类可能具有的所有频率范围。

除了等价类公理之外，类似的还有等价属性公理可以定义，这里不重复介绍了。

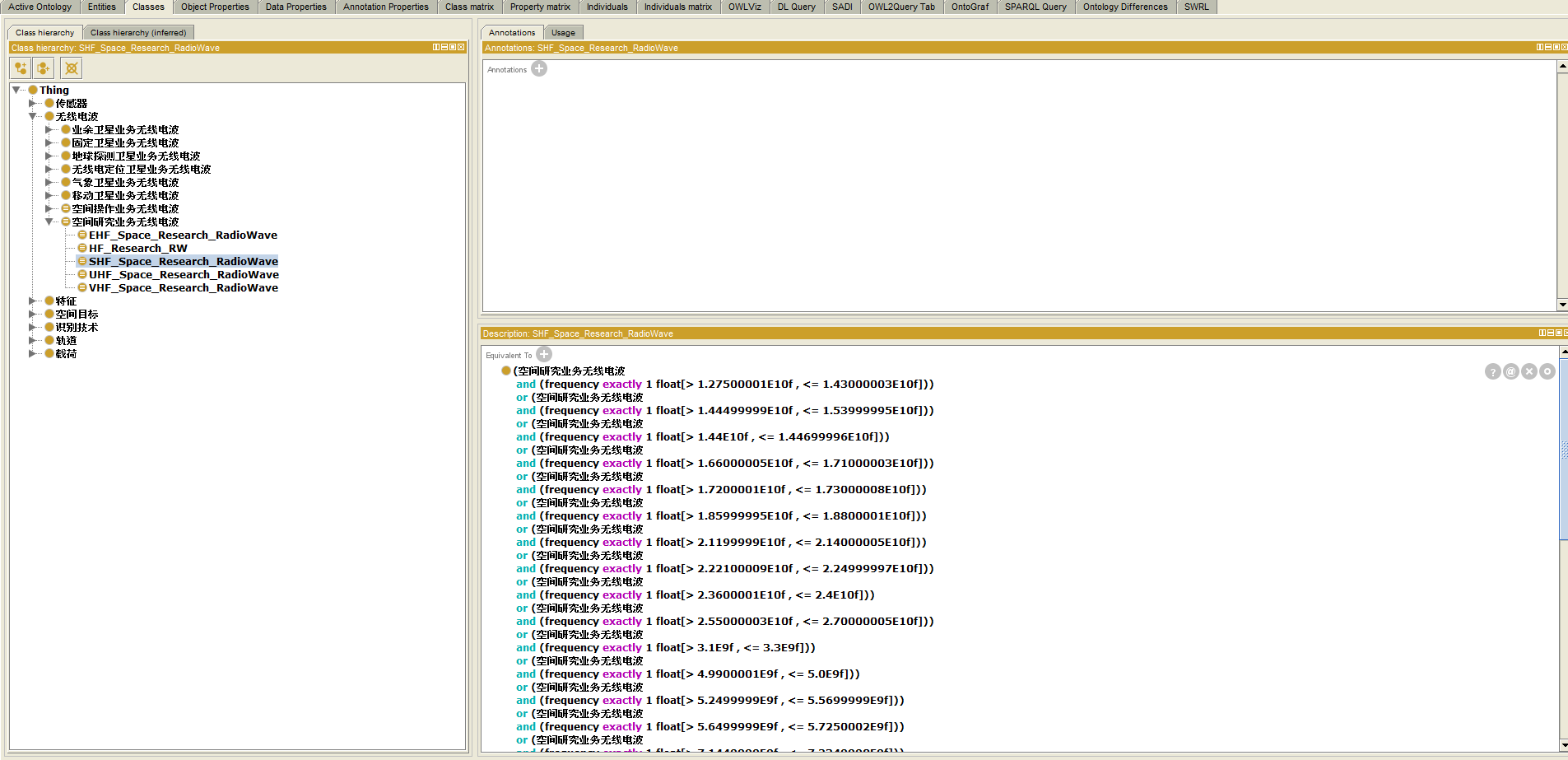


图3.4 类SHF\_Space\_Research\_RW的等价公理描述

（2）类覆盖公理。

类覆盖公理是指一个类概念内包含的子类是完备的（即全覆盖，不存在其他子类的可能）。也就是说，只要实例分类到该类，那么结果一定是其子类中的一个，不可能存在其他情况。增加类覆盖公理可通过定义等价类实现，即定义该类的等价类是其子类的并集。

如图3.5所示为碎片按尺寸的分类情况，用等价类表示一个碎片只可能是小碎片、风险碎片或者大碎片。

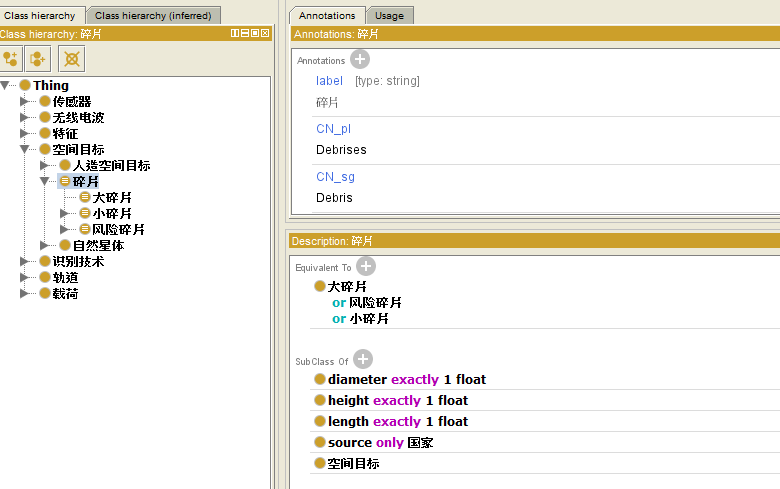


图3.5 碎片的覆盖公理

（3）类对象的属性描述公理。

在class标签界面右侧下方的Description视窗中，存在一个Superclasses项，可用于定义一个类所具有的属性及其约束。这里定义的属性及其约束形成一个类对象要满足的必要条件。当等价类定义空缺时，该类称作原始类（primitive class）

在protégé工具中，本体缺省要求一个类的所有属性都必须在Superclasses项中给予定义，不能遗漏。例如，人造空间目标所应具有的所有属性，都在如图3.6所示的Superclasses项中，给出了完整定义，其中Superclasses下方的匿名类中，包含了人造空间目标继承自父类空间目标的所有属性。

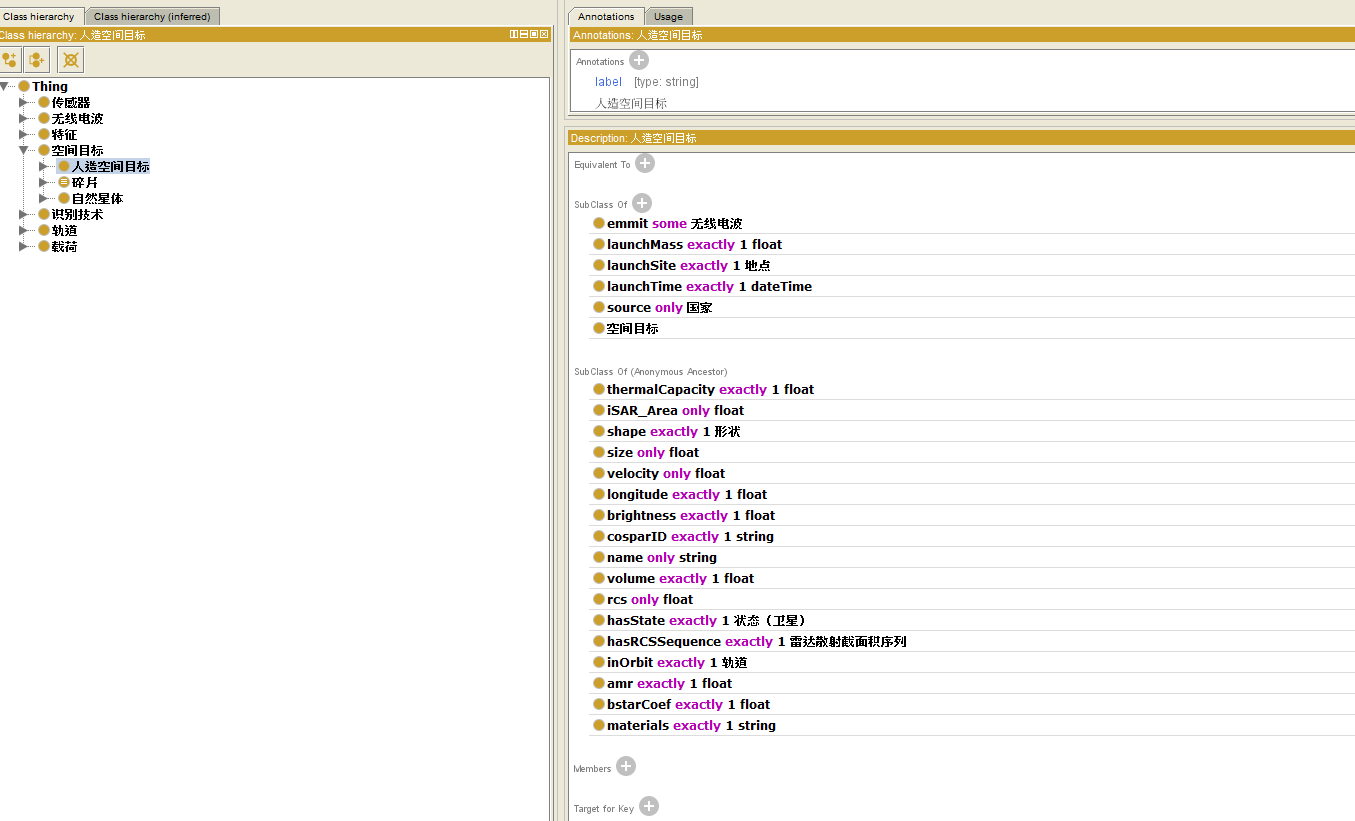


图3.6 人造空间目标约束

在定义属性描述公理时， 需要严格区分“some”“only”“exactly”三个限制术语的定义。some是存在量词，其语义相当于一阶逻辑语言中的“ ”，例如，类SpaceObject的属性描述公理“emmit some RadioWave”表示一个人造空间目标具有发射无线电波的属性；only是全称量词，其语义相当于一阶逻辑语言中的“ ”，匿名类中的属性描述公理“rcs only float”表示一个人造空间目标具有雷达散射截面积属性，且属性值只能为浮点数类型；匿名类中的属性描述公理“inOrbit exactly 1 Orbit”表示一个人造空间目标在轨道上，且有且仅有一个轨道。

对象属性的定义表示了类的对象之间具体的约束关系，例如，类SpaceObject的属性描述公理“hasState exactly 1 State”约束中，通过对象属性hasState 将SpaceObject类对象与State类对象关联起来，同时，进一步约束空间目标有且仅有一个状态等。更多的约束关系还包括其他的基数约束，例如“min”和“max”，其用法类似于“exactly”，这里的“min”表示该类对象必须参与的最少关系数量，例如，某类空间目标的属性描述公理为“hasPayload min 3 Payload”，意指该类空间目标至少具有3项载荷；“max”表示该类对象能够参与的最多关系数量，例如，某类空间目标的属性描述公理为“hasPayload max 3 Payload”，意指该类空间目标最多只能有3项载荷。

数值属性的关联与对象属性类似，如上图所示的类SpaceObject的属性描述公理“rcs only float”，表示空间目标对象具有rcs属性，且属性值只能为浮点数类型。与对象属性关联方法的区别在于，“float”是protégé工具中的浮点数类型，而非用户自定义的类概念。

（4）闭公理。

定义在属性上的闭公理用于约束关系的全称量词的取值范围。例如，某类空间目标在Description视窗中针对对象属性“hasPayload”上定义了如下所示的闭公理：

hasPayload some Multi\_Beam\_Antenna

hasPayload some Spot\_Beam\_Antenna

hasPayload only (Multi\_Beam\_Antenna or Spot\_Beam\_Antenna)

则表明该类空间目标必须带有天线，且所带天线只能是多波束天线或点波束天线。其语义相当于:

。

（5）取值约束。

取值约束是指属性取值的值域约束。例如，图2.3所示的空间目标本体中，由于特征、轨道、载荷等概念均为空间目标的属性取值的值域，而传感器、识别技术等概念又是特征、轨道、载荷等概念的属性取值的值域。因此，这些概念的属性约束需满足protégé工具中关于Value Partitions的设计要求。

4.增加新的SWRL规则

在protégé工具中，OWL语言表示的本体，可定义SWRL规则以增强本体的知识表达能力。SWRL规则的基本形式为：

antecedent ⇒ consequent

其中，规则前件antecedent和规则后件consequent均可表示为原子的合取，即的形式，变量则需要在标识符前加“？”。例如，一个空间目标识别规则可用如下的SWRL规则表示。



上述SWRL规则的含义为：如果是一个空间目标；的形状属性为；是一个不规则形状；的体积是；小于0.001个单位，那么，是一个碎片。

SWRL规则中的所用的词汇必须是在SWRL中系统中内置的一些词汇，如lessThan、greaterThan、Equal等，或采用OWL DL定义的类或属性概念，不能使用未定义的名词。SWRL中的内置词汇需要查询相关手册。

如图3.7所示，点击Rules右边的“+”按钮，即可在弹出的创建规则对话框中增加新的目标识别规则。

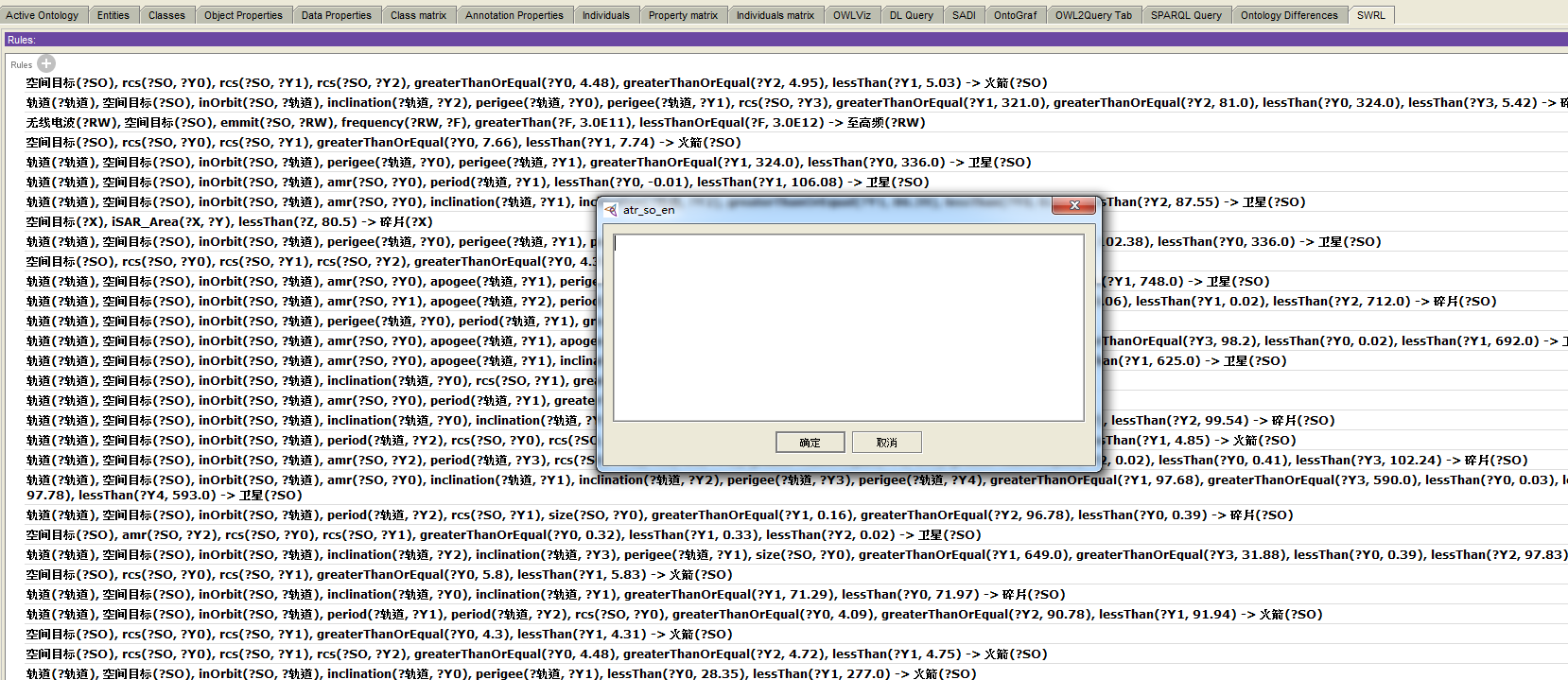


图3.7增加新的目标识别规则

5.增加新的个体

在protégé工具中增加个体有很多种方法。建立实例的方法与创建类、属性等相似，首先选中Individual标签，在该界面内，左侧选中新建实例，然后输入实例名，再输入实例的其他属性信息。

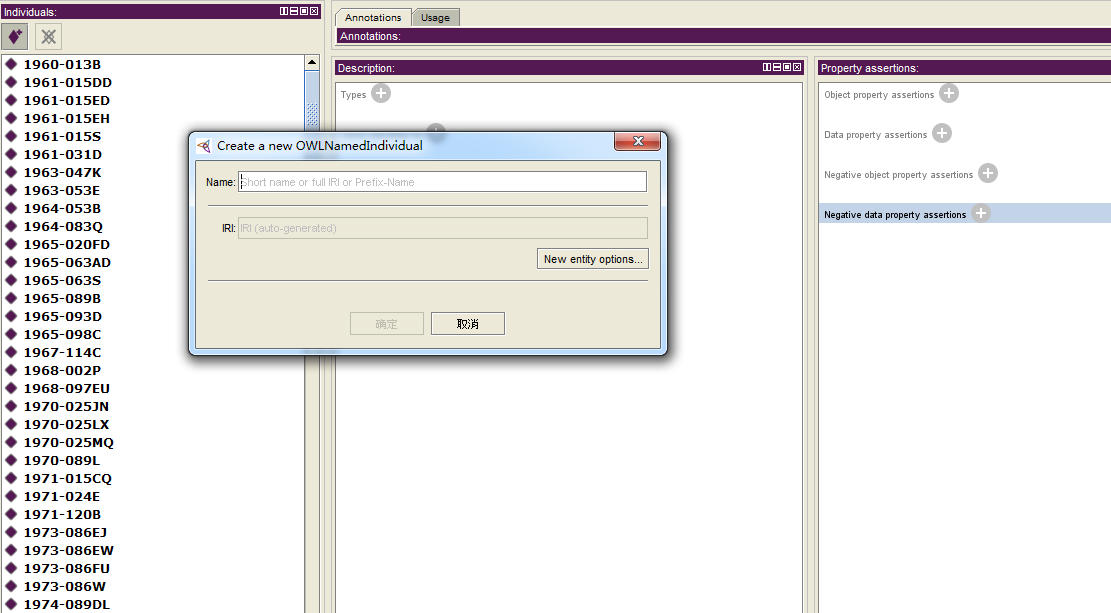


图3.8 增加新个体

### 6.2.3本体的应用

使用protégé工具，可以方便、快捷地应用一个构建好的知识模型。例如，对本体进行一致性检测和实例类型推理；使用推理机插件对本体进行推理；通过本体查询语言对本体进行信息查询等等。

1.本体的一致性检测

本体的一致性检验就是保证本体中不包含任何矛盾的事实，该功能可以由推理机推理进行验证。

在protégé工具中，建立完成一个本体后首先要进行一致性检验，并对类的划分进行完善，以保证知识库的正确性。目前的OWL推理算法中有一套成熟的Tableau算法，可以实现其概念的一致性检查且计算复杂度较低。Tableau算法的主要思想就是把概念集合通过一系列推理规则不断细化、扩展，直到发现冲突。如果直到概念被完全扩展仍然没有发现不可避免的冲突，就证明概念集合是一致的。

以下以Pellet推理机插件为例，说明本体的一致性检验。Pellet是以描述逻辑为基础，基于tableau算法开发的推理机，可作为Protégé的推理机插件使用，以验证本体的一致性。

在Protégé工具上打开一个已建立的本体，例如，OntoStar2.01.OWL本体文件，选择Reasoner菜单栏下“Pellet”选项并点击“Start reasoner”进行推理，如图4.1所示：

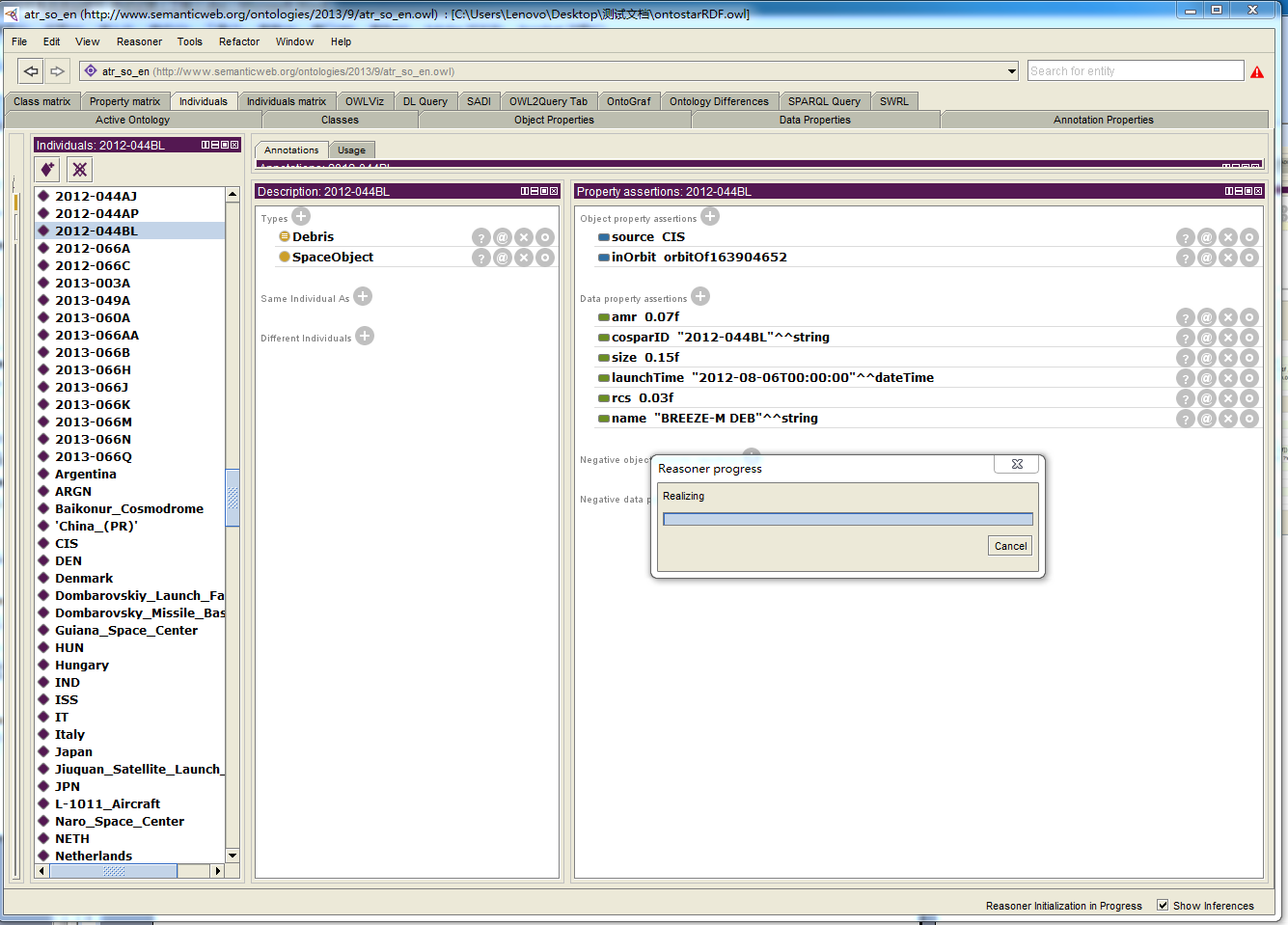


图4.1 对OntoStar2.01使用Pellet推理机进行推理

如果本体中存在不一致的知识，将不能通过通过一致性检验。在这里将OntoStar2.01中定义空间目标“2012-044BL” 的类型由“Debris”改为“Satellite”作为演示例子，进行一致性检验，由于存在矛盾的知识（OntoStar2.01中定义了“Debris”和“Satellite”不相交，即不存在一个空间目标既为“Debris”又为“Satellite”， 而推导出来“2012-044BL”为“Debris”，定义却为“Satellite”），Protégé工具弹出一个警告框，告知用户OntoStar2.01没有通过一致性检验，如图4.2所示：

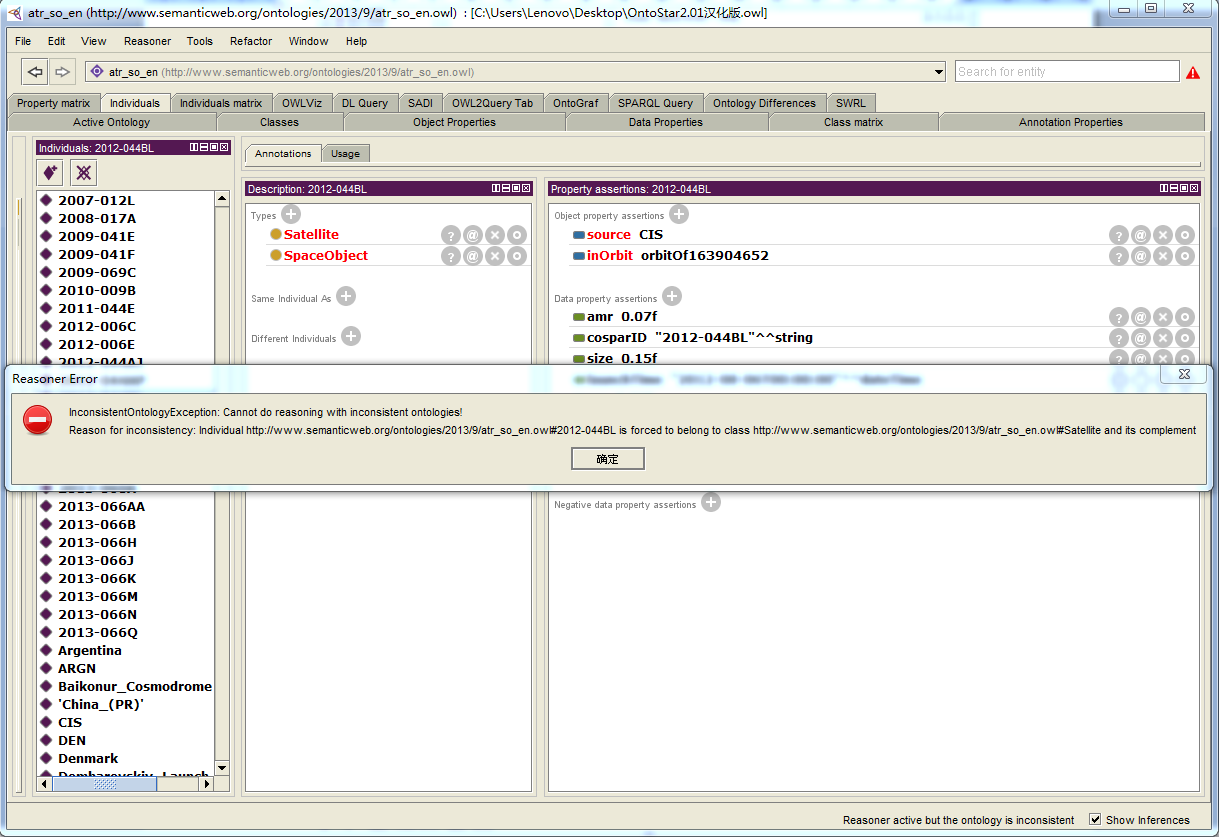


图4.2 未通过一致性检测情况

2.本体的非平凡性检验

当本体的所有概念体系和公理约束定义完成后，可通过输入所有的实例，检验公理的正确性和非平凡性。该步骤可在个体定义环节进行，也可在整个本体构建完成后进行，目的在于检测公理定义的正确性。

3.推理机推导

在protégé工具中，由于通过SWRL规则可定义推理的规则集，因而，推理机插件即可依据本体的知识库进行推理。

例如，导入一个实例后可基于本体对未分类的实例进行推理识别。

点击Reasoner菜单栏内，选择要使用的推理机，如pellet推理机，机器会根据已有公理和规则进行推理，如图4.3的一个实例，就推导出该实例是一个卫星。

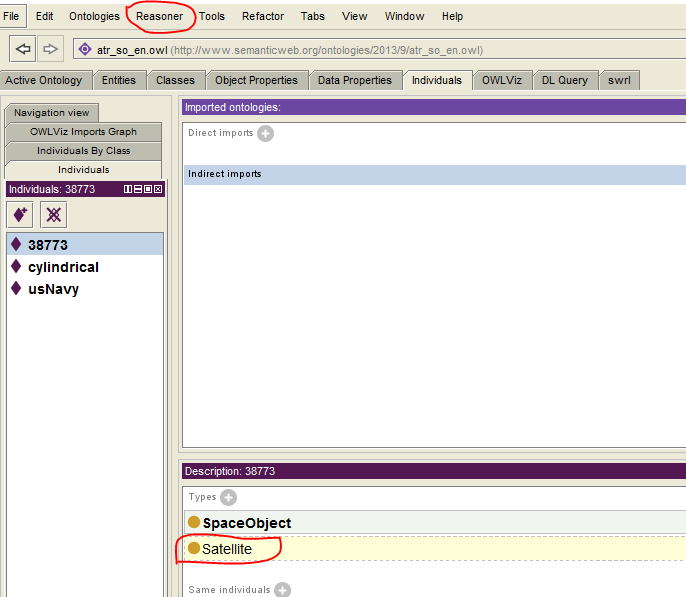


图4.3 空间目标实例推导为卫星

新版本的protégé工具还可以直接显示推理的详细过程，包括了推理中所使用的概念和公理。

4.信息查询

Protégé工具提供了两种方式对本体进行查询，即DL查询和SPARQL查询。

在DL Query标签中可以使用DL语句查询满足条件的类和实例，例如，查询本体OntoStar2.01 “源”为“美国”的空间目标实例，在Query对话框中输入DL语句''source value U''，勾选左侧的Instance多选框，并点击“Execute”按钮，Query Results对话框中将显示查询结果，如图4.4所示。

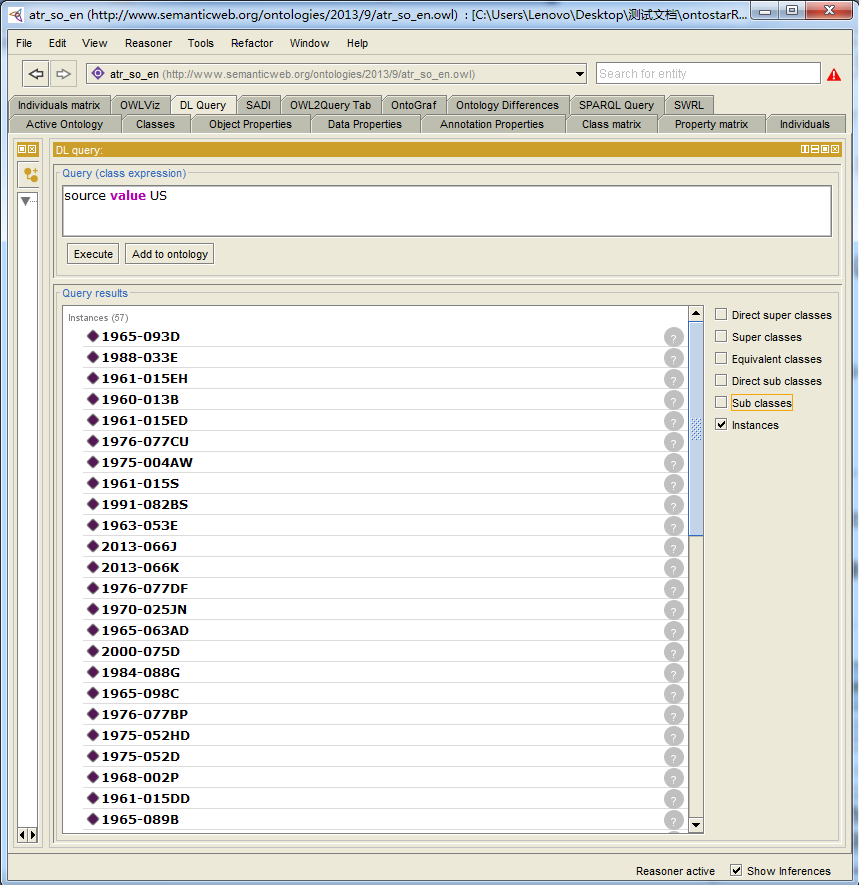


图4.4 对OntoStar2.01使用DL语句查询

在SPARQL Query标签中也可以查询满足条件的类和实例（统称为实体），例如，查询本体OntoStar2.01 “源”为“美国”的实体，在SPARQL query对话框中输入以下SPARQL语句：

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

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

PREFIX so: <http://www.semanticweb.org/ontologies/2013/9/atr\_so\_en.owl#>

SELECT ?subject WHERE {?subject so:source so:US.}

点击“Execute”按钮，视窗下半部分的对话框中将显示查询结果，如图4.5所示：

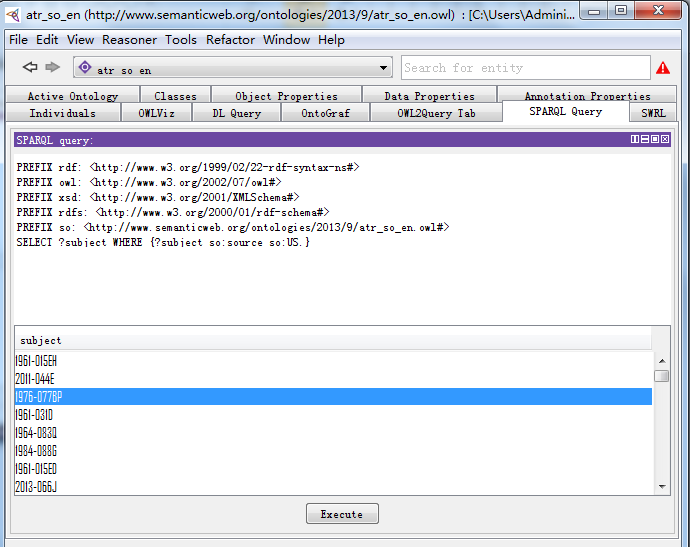
****

图4.5 对OntoStar2.01使用SPARQL语句查询