

Ontoloji Geliştirme 101: Kendinizi Yaratmak İçin Bir Kılavuz İlk Ontoloji

Natalya F.Noy ve Deborah L. McGuinness
Stanford Üniversitesi, Stanford, CA, 94305
noy@smi.stanford.edu ve dlm@ksl.stanford.edu

1 Neden bir ontoloji geliştirmelisiniz?

Son yıllarda ontolojilerin gelişimi - içindeki terimlerin açık biçimsel özellikleri etki alanı ve aralarındaki ilişkiler (Gruber 1993) - Yapay Zeka laboratuvarlarından alan uzmanlarının masaüstlerine. Ontolojiler var World-Wide Web'de yaygınlaştı. Web üzerindeki ontolojiler, büyük Web sitelerini (Yahoo! gibi) ürün kategorilerine ayıran sınıflandırmalar satış ve özellikleri (Amazon.com'daki gibi). WWW Konsorsiyumu (W3C), için bir dil olan Kaynak Tanımlama Çerçevesini (Brickley ve Guha 1999) geliştirmek Elektronik araçlar aramada anlaşılır kılmak için Web sayfalarında bilgi kodlama bilgi için. Savunma İleri Araştırma Projeleri Ajansı (DARPA), W3C ile birlikte, DARPA Agent Markup Language (DAML) geliştiriyor. RDF'yi, ajan etkileşimini kolaylaştırmayı amaçlayan daha etkileyici yapılarla genişletmek Web (Hendler ve McGuinness 2000). Birçok disiplin artık standart ontolojiler geliştiriyor alan uzmanlarının alanlarındaki bilgileri paylaşmak ve açıklama eklemek için kullanabileceği. Tıp için örneğin, SNOMED (Fiyat ve fiyat) gibi büyük, standartlaştırılmış, yapılandırılmış sözcük dağarcığı üretti. Spackman 2000) ve Birleşik Tıp Dil Sisteminin anlamsal ağı (Humphreys ve Lindberg 1993). Geniş genel amaçlı ontolojiler de ortaya çıkıyor. İçin Örneğin, Birleşmiş Milletler Kalkınma Programı ve Dun & Bradstreet, ürünler için terminoloji sağlayan UNSPSC ontolojisini geliştirme çabaları ve hizmetler (www.unspsc.org).

Ontoloji, bilgiyi paylaşmaya ihtiyaç duyan araştırmacılar için ortak bir kelime hazinesi tanımlar. bir alan adı. Alandaki temel kavramların makine tarafından yorumlanabilir tanımlarını içerir ve aralarındaki ilişkiler.

Birisi neden bir ontoloji geliştirmek istesin? Nedenlerden bazıları şunlardır:

- Bilgi yapısına ilişkin ortak anlayışı insanlar arasında paylaşmak veya yazılım araçları
- Alan bilgisinin yeniden kullanımını etkinleştirmek için
- Etki alanı varsayımlarını açık hale getirmek için
- Alan bilgisini operasyonel bilgiden ayırmak için
- Alan bilgisini analiz etmek

İnsanlar veya yazılımlar arasında bilgi yapısının ortak anlayışını paylaşmak araçlar ontolojileri geliştirmede en yaygın hedeflerden biridir (Musen 1992; Gruber 1993). Örneğin, birkaç farklı Web sitesinin tıbbi bilgiler içerdiğini veya tıbbi e-ticaret hizmetleri sağlamak. Bu Web siteleri aynı şeyi paylaşır ve yayınlarsa Hepsi kullandıkları terimlerin temel ontolojisi, daha sonra bilgisayar ajanları ayıklayabilir ve bu farklı sitelerden toplanan bilgiler. Temsilciler bu toplu halde kullanabilir kullanıcı sorgularını yanıtlamak için veya diğer uygulamalara giriş verileri olarak bilgi.

Alan bilgisinin yeniden kullanılmasını sağlamak , son zamanlarda yaşanan artışın arkasındaki itici güçlerden biriydi. ontoloji araştırması. Örneğin, birçok farklı alan için modellerin, zaman kavramı. Bu temsil, zaman aralıkları, zaman içindeki noktalar,

göreceli zaman ölçüleri vb. Bir grup araştırmacı böyle bir ontoloji geliştirirse ayrıntılı olarak, diğerleri bunu kendi alanları için yeniden kullanabilir. Ek olarak, büyük bir

Sayfa 2

ontoloji, büyük parçaların bölümlerini tanımlayan birkaç mevcut ontolojiyi bütünleştirebiliriz. alan adı. Ayrıca UNSPSC ontolojisi gibi genel bir ontolojiyi yeniden kullanabilir ve genişletebiliriz. ilgi alanımızı tanımlamak için.

Bir uygulamanın altında yatan *açık alan varsayımları* yapmak, aşağıdakileri mümkün kılar: Etki alanı hakkındaki bilgimiz değişirse, bu varsayımları kolayca değiştirin. Sabit kodlama programlama dili kodunda dünya hakkındaki varsayımlar, bu varsayımları sadece bulması ve anlaması zor ama aynı zamanda değiştirilmesi de zor, özellikle programlama uzmanlığı. Ek olarak, alan bilgisinin açık özellikleri yararlıdır etki alanındaki terimlerin ne anlama geldiğini öğrenmesi gereken yeni kullanıcılar için.

Alan bilgisini operasyonel bilgidan ayırmak başka bir yaygın kullanımdır ontolojilerin. Bir ürünü bileşenlerine göre yapılandırma görevini tanımlayabiliriz. gerekli bir spesifikasyona göre ve bu konfigürasyonu bağımsız olarak yapan bir programı uygulamaya ürünlerin ve bileşenlerin kendileri (McGuinness ve Wright 1998). O zaman yapabiliriz PC bileşenlerinin ve özelliklerinin ontolojisini geliştirin ve algoritmayı sipariş üzerine yapılan PC'leri yapılandırın. Asansörleri yapılandırmak için de aynı algoritmayı kullanabiliriz. ona bir asansör bileşeni ontolojisini “besliyoruz” (Rothenfluh ve diğerleri, 1996).

Alan bilgisini analiz etmek, terimlerin bildirim niteliğinde bir spesifikasyonu olduğunda mümkündür. mevcut. Her ikisi de yeniden kullanmaya çalışırken terimlerin resmi analizi son derece değerlidir var olan ontolojiler ve onları genişletme (McGuinness ve ark. 2000).

Genellikle alanın ontolojisi kendi başına bir amaç değildir. Bir ontoloji geliştirmek benzerdir diğer programların kullanması için bir dizi veri ve bunların yapısını tanımlama. Problem çözme yöntemleri, alandan bağımsız uygulamalar ve yazılım araçları ontolojileri ve bilgi tabanlarını kullanır veri olarak ontolojilerden inşa edilmiştir. Örneğin, bu makalede bir şarap ontolojisi geliştiriyoruz ve yiyecekler ve yemeklerle uygun şarap kombinasyonları. Bu ontoloji daha sonra bir restoran yönetim araçları paketindeki bazı uygulamaların temeli: Bir uygulama, günün menüsü için şarap önerileri oluşturun veya garsonların ve müşterilerin sorularını yanıtlayın. Başka bir uygulama, bir şarap mahzeninin envanter listesini analiz edebilir ve hangi şarabın genişletilecek kategoriler ve yaklaşan menüler için satın alınacak belirli şaraplar veya yemek kitapları.

Bu kılavuz hakkında

Protégé-2000 (Protege 2000), Ontolingua (Ontolingua 1997) ve Ontoloji düzenleme ortamları olarak Chimaera (Chimaera 2000). Bu kılavuzda biz Örneklerimiz için Protégé-2000'i kullanın.

Bu kılavuzda kullandığımız şarap ve yemek örneği, genel olarak bir örneğe dayanmaktadır. Bir bilgi temsili olan CLASSIC'i tanımlayan bir makalede sunulan bilgi tabanı sistem tanım-mantık yaklaşımına dayanmaktadır (Brachman ve diğerleri 1991). KLASİK öğretici (McGuinness ve ark. 1994) bu örneği daha da geliştirmiştir. Protégé-2000 ve diğer çerçeve tabanlı sistemler ontolojileri bildirimli olarak tanımlar ve sınıfın ne olduğunu açıkça belirtir. hiyerarşi, bireylerin ait olduğu sınıflardır.

Bu kılavuzdaki bazı ontoloji-tasarım fikirleri, nesne yönelimli literatürden kaynaklanmıştır. tasarım (Rumbaugh ve diğerleri 1991; Booch ve diğerleri 1997). Ancak ontoloji gelişimi nesne yönelimli programlamada sınıfları ve ilişkileri tasarlamaktan farklıdır. Nesne-odaklı programlama, öncelikle sınıflardaki yöntemler etrafında merkezlenir; bir programcı Bir sınıfın *operasyonel* özelliklerine dayalı kararlar tasarlar, bir ontoloji tasarımcısı bu kararları bir sınıfın *yapısal* özelliklerine göre verir. Sonuç olarak, bir sınıf ontolojideki sınıflar arasındaki yapı ve ilişkiler, bir ontolojinin yapısından farklıdır. nesne yönelimli bir programda benzer etki alanı.

Bir ontoloji geliştiricisinin uğraşması gerekebilecek tüm sorunları ele almak imkansızdır. ve bu kılavuzda hepsine değinmeye çalışmıyoruz. Bunun yerine, bir başlangıç sağlamaya çalışıyoruz nokta; yeni bir ontoloji tasarımcısının ontolojileri geliştirmesine yardımcı olacak bir başlangıç kılavuzu. Şurada

Sayfa 3

sonunda, daha karmaşık yapıların ve tasarımların açıklamalarını aramak için yerler öneriyoruz etki alanı gerektiriyorsa mekanizmalar.

Son olarak, tek bir doğru ontoloji-tasarım metodolojisi yoktur ve biz bunu yapmaya çalışmadık. birini tanımlayın. Burada sunduğumuz fikirler, kendi başımıza yararlı bulduğumuzdur. ontoloji geliştirme deneyimi. Bu kılavuzun sonunda bir referans listesi öneriyoruz alternatif metodolojiler.

2 Bir ontolojide neler var?

Yapay Zeka literatürü bir ontolojinin birçok tanımını içerir; çoğu bunlar birbiriyle çelişir. Bu kılavuzun amaçları doğrultusunda, **ontoloji** resmi bir açıklıktır bir söylem alanındaki kavramların açıklaması (**sınıflar** (bazen **kavramlar** olarak adlandırılır)), kavramın çeşitli özelliklerini ve niteliklerini tanımlayan her kavramın özellikleri (**yuvalar** (bazen **roller** veya **özellikler** olarak adlandırılır)) ve alanlarla ilgili kısıtlamalar (**yönler** (bazen **rol kısıtlamaları**)), Sınıfların bir dizi bireysel **örneği** ile birlikte bir ontoloji bir **bilgi tabanı oluşturur** . Gerçekte, ontolojinin bittiği ince bir çizgi vardır ve bilgi tabanı başlar.

Sınıflar çoğu ontolojinin odak noktasıdır. Sınıflar, alandaki kavramları tanımlar. İçin Örneğin, bir şarap sınıfı tüm şarapları temsil eder. Belirli şaraplar bu sınıfın örnekleridir. Bu belgeyi okurken önünüzdeki bardağa bordo şarabı Bordeaux şarapları sınıfı. Bir sınıf , daha fazla kavramları temsil eden **alt sınıflara** sahip olabilir. süper sınıftan daha özel. Örneğin, tüm şarapların sınıfını kırmızı, beyaz olarak bölebiliriz, ve roze şarapları. Alternatif olarak, tüm şarapların bir sınıfını köpüklü ve köpüklü olmayan şaraplar olarak ayırabiliriz. şampanya.

Yuvalar, sınıfların ve örneklerin özelliklerini tanımlar: Château Lafite Rothschild Pauillac şarabının tam bir vücudu vardır; Château Lafite Rothschild tarafından üretildi şaraphane. Bu örnekte şarabı açıklayan iki yuvamız var: değerine sahip yuva gövdesi tam ve değerli Château Lafite Rothschild şaraphanesine sahip bir slot üreticisi. Şurada sınıf düzeyinde, Wine sınıfının örneklerinin, bunları tanımlayan yuvalara sahip olacağını söyleyebiliriz. lezzet, vücut, şeker seviyesi, şarabın yapımcısı vb. ı

Wine sınıfının tüm örnekleri ve alt sınıfı Pauillac, bir slot makinesinin değerine sahiptir. Winery sınıfının bir örneği olan (Şekil 1). Winery sınıfının tüm örneklerinde bir tüm şarapları ifade eden slot üretimleri (Wine ve alt sınıflarının örnekleri) şaraphanenin ürettiği.

Pratik anlamda, bir ontoloji geliştirmek şunları içerir:

- ontolojide sınıfları tanımlama,
- sınıfları bir taksonomik (alt sınıf-üst sınıf) hiyerarşisinde düzenlemek,
- slotları tanımlama ve bu slotlar için izin verilen değerleri tanımlama,
- örnekler için yuvalar için değerlerin doldurulması.

Daha sonra bu sınıfların bireysel örneklerini doldurarak tanımlayarak bir bilgi tabanı oluşturabiliriz. belirli yuva değeri bilgileri ve ek yuva kısıtlamaları.

ı Sınıf adlarını büyük harfle yazıyoruz ve yuva adlarını küçük harflerle başlatıyoruz. Biz de kullanmak daktilo yazı için örnek ontolojideki tüm terimler.

4. sayfa

Şekil 1. Şarap alanındaki bazı sınıflar, örnekler ve aralarındaki ilişkiler. Siyah kullandık sınıflar için ve örnekler için kırmızı. Doğrudan bağlantılar, yuvaları ve örnek-of gibi dahili bağlantıları temsil eder. ve alt sınıfı.

3 Basit Bir Bilgi Mühendisliği Metodolojisi

Daha önce de söylediğimiz gibi, ontolojileri geliştirmek için tek bir "doğru" yol veya metodoloji yoktur. Burada dikkate alınması gereken genel konuları tartışıyoruz ve bir geliştirme için olası bir süreci sunuyoruz. ontoloji. Ontoloji geliştirmeye yinelemeli bir yaklaşım tanımlıyoruz: kaba bir ontolojide ilk geçiş. Daha sonra gelişen ontolojiyi gözden geçirip iyileştiriyoruz ve detaylar. Yol boyunca, bir tasarımcının vermesi gereken modelleme kararlarını şu şekilde tartışıyoruz: artıları, eksileri ve farklı çözümlerin sonuçları.

Öncelikle, ontoloji tasarımında uygulayacağımız bazı temel kuralları vurgulamak istiyoruz. birçok kez başvurun. Bu kurallar oldukça dogmatik görünebilir. Bununla birlikte, yardımcı olabilirler birçok durumda tasarım kararları.

- 1) *Bir alanı modellemenin tek bir doğru yolu yoktur - her zaman vardır uygulanabilir alternatifler. En iyi çözüm neredeyse her zaman şunlara bağlıdır: aklınızdaki uygulama ve tahmin ettiğiniz uzantılar.*
- 2) *Ontoloji geliştirme, zorunlu olarak yinelemeli bir süreçtir.*
- 3) *Ontolojideki kavramlar nesnelere yakın olmalıdır (fiziksel veya mantıksal) ve ilgi alanınızdaki ilişkiler. Bunlar büyük olasılıkla isimler (nesneler) veya fiiller (ilişkiler) sizi tanımlayan alan adı.*

Yani, ontolojiyi ne için kullanacağımıza ve ne kadar detaylı veya genel ontoloji, yolun ilerisindeki modelleme kararlarının çoğuna rehberlik edecektir. Arasında birkaç uygulanabilir alternatif varsa, hangisinin daha iyi çalışacağını belirlememiz gerekecek. öngörülen görev, daha sezgisel, daha genişletilebilir ve daha sürdürülebilir olmalıdır. Ayrıca ihtiyacımız var ontolojinin dünyanın gerçekliğinin bir modeli olduğunu ve ontoloji bu gerçeği yansıtmalıdır. Ontolojinin ilk versiyonunu tanımladıktan sonra, Uygulamalarda veya problem çözme yöntemlerinde kullanarak veya tartışarak değerlendirip hatalarını ayıklayın alandaki uzmanlarla veya her ikisi ile. Sonuç olarak, neredeyse kesinlikle revize etmemiz gerekecek ilk ontoloji. Bu yinelemeli tasarım süreci büyük olasılıkla tüm ontolojinin yaşam döngüsü.

5. Sayfa

Adım 1. Ontolojinin alanını ve kapsamını belirleyin

Etki alanını ve kapsamını tanımlayarak bir ontolojinin gelişimine başlamayı öneriyoruz. Bu birkaç temel soruyu yanıtlamaktır:

- Ontolojinin kapsayacağı alan nedir?
- Ontolojiyi ne için kullanacağız?
- Ontolojideki bilgiler ne tür sorular için cevap vermelidir?
- Ontolojiyi kim kullanacak ve sürdürecektir?

Bu soruların cevapları ontoloji tasarım süreci sırasında değişebilir, ancak herhangi bir zamanda belirli bir süre modelin kapsamını sınırlamaya yardımcı olurlar.

Daha önce tanıttığımız şarap ve yiyecek ontolojisini düşünün. Gıdanın temsili ve şaraplar ontolojinin alanıdır. Bu ontolojiyi uygulamalar için kullanmayı planlıyoruz bu iyi şarap ve yemek kombinasyonlarını öneriyor.

Doğal olarak, farklı şarap türlerini tanımlayan kavramlar, ana yemek türleri, şarap ve yiyeceğin iyi kombinasyonu ve kötü bir kombinasyon ontolojimizde şekillenecektir. Şurada: aynı zamanda, ontolojinin envanteri yönetmek için kavramlar içermesi olası değildir bir şaraphanede veya bir restorandaki çalışanlar, bu kavramlar bir şekilde ilgili olsa da şarap ve yemek kavramları.

Tasarladığımız ontoloji, makalelerin doğal dilde işlenmesine yardımcı olmak için kullanılacaksa şarap dergilerinde, eşanlamlıları ve konuşma parçası bilgilerini eklemek önemli olabilir ontolojideki kavramlar için. Ontoloji, restoran müşterilerinin karar vermesine yardımcı olmak için kullanılacaksa hangi şarabı sipariş edeceğimizi, perakende fiyatlandırma bilgilerini eklememiz gerekir. Şarap için kullanılıyorsa bir şarap mahzeni stoklayan alıcılar, toptan satış fiyatları ve bulunabilirlik gerekli olabilir. Eğer ontolojiyi sürdürecektir kişiler alanı farklı bir dilde tanımlar ontoloji kullanıcılarının dilinden bakıldığında, arasındaki eşlemeyi sağlamamız gerekebilir. Diller.

Yetkinlik soruları.

Ontolojinin kapsamını belirlemenin yollarından biri, bir soru listesi oluşturmaktır. ontolojiye dayalı bilgi tabanı, **yeterlilik sorularını cevaplayabilmelidir.**

(Gruninger ve Fox 1995) . Bu sorular daha sonra turnusol testi görevi görecek: ontoloji bu tür soruları yanıtlamak için yeterli bilgi içeriyor mu? Cevapları yap belirli bir alan için belirli bir ayrıntı düzeyi veya temsilini mi gerektiriyor? Bu yetkinlik sorular sadece bir taslaktır ve ayrıntılı olmaları gerekmez.

Şarap ve yemek alanında, olası yeterlilik soruları şunlardır:

- Şarap seçerken hangi şarap özelliklerini dikkate almalıyım?
- Bordeaux kırmızı mı yoksa beyaz bir şarap mı?
- Cabernet Sauvignon deniz ürünleri ile iyi gider mi?
- Izgara et için en iyi şarap seçimi nedir?
- Bir şarabın hangi özellikleri bir yemeğe uygunluğunu etkiler?
- Belirli bir şarabın bir buketi veya gövdesi vintage yılına göre değişir mi?
- Napa Zinfandel için iyi şaraplar hangileriydi?

Bu sorular listesinden yola çıkarak ontoloji, çeşitli şaraplar hakkındaki bilgileri içerecektir. özellikleri ve şarap türleri, eski yıllar - iyi ve kötü olanlar - yiyeceklerin sınıflandırılması uygun bir şarap seçimi için önemlidir, önerilen şarap ve yiyecek kombinasyonları.

Adım 2. Mevcut ontolojileri yeniden kullanmayı düşünün

Neredeyse her zaman bir başkasının ne yaptığını düşünmeye ve yapıp yapamayacağımızı kontrol etmeye değer Belirli etki alanımız ve görevimiz için mevcut kaynakları iyileştirin ve genişletin. Var olanı yeniden kullanmak

Sayfa 6

Sistemimizin diğer uygulamalarla etkileşime girmesi gerekiyorsa ontolojiler bir gereklilik olabilir. belirli ontolojilere veya kontrollü kelime dağarcığına zaten bağlıydı. Birçok ontoloji halihazırda elektronik biçimde mevcuttur ve bir ontoloji geliştirmeye aktarılabilir kullandığınız ortam. Bir ontolojinin ifade edildiği biçimcilik genellikle önemli değil, çünkü birçok bilgi-temsili sistemi ontolojileri içe ve dışa aktarabilir. Bir bilgi temsili sistemi, belirli bir biçimcilikle doğrudan çalışmasa bile, Bir ontolojiyi bir formalizmden diğerine çevirme görevi genellikle zor değildir bir.

Web'de ve literatürde yeniden kullanılabilir ontolojilerin kütüphaneleri vardır. Örneğin biz Ontolingua ontoloji kitaplığını (<http://www.ksl.stanford.edu/software/ontolingua/>) veya DAML ontoloji kütüphanesi (<http://www.daml.org/ontologies/>). Ayrıca bir dizi var halka açık ticari ontolojiler (örneğin, UNSPSC (www.unspsc.org), RosettaNet (www.rosettanet.org), DMOZ (www.dmoz.org)).

Örneğin, Fransız şaraplarının bir bilgi tabanı zaten mevcut olabilir. Bunu ithal edebilirsek bilgi tabanı ve onun dayandığı ontoloji, sadece sınıflandırmaya sahip olmayacağız Fransız şaraplarının değil, aynı zamanda şarap özelliklerinin sınıflandırılmasında ilk geçiş şarapları ayırt eder ve tanımlar. Şarap özelliklerinin listeleri şu adresten edinilebilir: müşterilerin şarap satın almak için kullanmayı [düşündükleri www.wines.com](http://www.wines.com) gibi ticari Web siteleri .

Ancak bu kılavuz için, ilgili ontolojilerin halihazırda mevcut olmadığını ve başladığını varsayacağız. ontolojiyi sıfırdan geliştirmek.

Adım 3. Ontolojideki önemli terimleri sıralayın

Hakkında veya hakkında açıklama yapmak istediğimiz tüm terimlerin bir listesini yazmak yararlıdır. bir kullanıcıya açıklamak için. Konuşmak istediğimiz terimler nelerdir? Hangi mülkler işe yarar bu terimler var mı? Bu terimler hakkında ne söylemek isteriz? Örneğin, önemli şarapla ilgili terimler arasında şarap , üzüm , şaraphane , yer , bir şarabın rengi , vücut , tat ve şeker içeriği; balık ve kırmızı gibi farklı yiyecek türleri et; beyaz şarap gibi alt şarap türleri vb. Başlangıçta, bir kavramlar arasında örtüşme endişesi duymadan kapsamlı terimler listesi , terimler arasındaki ilişkileri veya kavramların sahip olabileceği herhangi bir özelliği temsil eder veya kavramların sınıflar mı yoksa slotlar mı olduğu.

Sonraki iki adım - sınıf hiyerarşisini geliştirme ve kavramların özelliklerini tanımlama (yuvalar) - yakından iç içe geçmiştir. Önce birini, sonra diğerini yapmak zor. Tipik olarak, hiyerarşide kavramların birkaç tanımını oluştururuz ve sonra şu şekilde devam ederiz: bu kavramların özelliklerini tanımlama vb. Bu iki adım aynı zamanda ontoloji tasarım sürecindeki önemli adımlar. Bunları burada kısaca anlatacağız ve sonra sonraki iki bölümü, olması gereken daha karmaşık konuları tartışarak geçirin düşünüldüğünde, ortak tuzaklar, alınacak kararlar vb.

Adım 4. Sınıfları ve sınıf hiyerarşisini tanımlayın

Bir sınıf hiyerarşisi geliştirmede birkaç olası yaklaşım vardır (Uschold ve Gruninger 1996) :

- Yukarıdan **aşağıya bir** geliştirme süreci, en genel olanın tanımıyla başlar. etki alanındaki kavramlar ve daha sonra kavramların uzmanlaşması. Örneğin, Şarap ve Yiyecek genel kavramları için sınıflar oluşturarak başlayabiliriz. Sonra Şarap sınıfını, alt sınıflarından bazılarını oluşturarak uzmanlaştırıyoruz: Beyaz şarap , Kırmızı şarap , Rosé şarabı. Kırmızı şarap sınıfını daha da sınıflandırabiliriz, örneğin, Syrah, Red Burgundy, Cabernet Sauvignon vb.

- **Aşağıdan yukarıya bir** geliştirme süreci, en spesifik olanın tanımlanmasıyla başlar. sınıflar, hiyerarşinin yaprakları, bu sınıfların daha sonra gruplanmasıyla daha genel kavramlar. Örneğin, Pauillac için sınıflar tanımlayarak başlıyoruz ve Margaux şarapları. Daha sonra bu ikisi için ortak bir süper sınıf oluşturuyoruz sınıflar — Medoc — bu da Bordeaux'nun bir alt sınıfıdır.
- **Kombinasyon** geliştirme süreci, yukarıdan aşağıya ve aşağıdan yukarı yaklaşımlar: Önce daha dikkat çekici kavramları tanımlarız, sonra genelleştiririz ve bunları uygun şekilde uzmanlaştırırız. Aşağıdakiler gibi birkaç üst düzey kavramla başlayabiliriz: Şarap ve Margaux gibi birkaç özel kavram. Daha sonra bunları bir Medoc gibi orta düzey konsept. O zaman tüm bunları oluşturmak isteyebiliriz Fransa'dan bölgesel şarap sınıfları, böylece bir dizi orta seviye üretiyor kavramlar.

Şekil 2, farklı genellik düzeyleri arasındaki olası bir dökümü göstermektedir.

Şekil 2. Şarap taksonomisinin farklı seviyeleri: Şarap en genel kavramdır. Kırmızı şarap, Beyaz şarap ve Rosé şarabı genel üst düzey konseptlerdir. Pauillac ve Margaux, hiyerarşideki en spesifik sınıflar (veya alt seviye kavramlar).

Bu üç yöntemden hiçbirisi doğası gereği diğerlerinden daha iyi değildir. Yaklaşım almak, büyük ölçüde alanın kişisel görüşüne bağlıdır. Bir geliştiricinin sistematikliği varsa Alanın yukarıdan aşağıya görünümü, yukarıdan aşağıya yaklaşımı kullanmak daha kolay olabilir. kombinasyon yaklaşımı çoğu ontoloji geliştiricisi için en kolay olanıdır çünkü kavramlar

"Ortada", alanda daha açıklayıcı kavramlar olma eğilimindedir (Rosch 1978) .

İlk önce en genel sınıflandırmayı ayırt ederek şarapları düşünme eğilimindeyseniz, yukarıdan aşağıya yaklaşım sizin için daha iyi sonuç verebilir. Topraklanarak başlamayı tercih edersen belirli örnekler, aşağıdan yukarıya yaklaşım daha uygun olabilir.

Hangi yaklaşımı seçersek seçelim, genellikle sınıfları tanımlayarak başlarız. Oluşturulan listeden

3. Adım, bağımsız varoluşa sahip nesneleri tanımlayan terimleri seçiyoruz bu nesneleri tanımlayan terimler. Bu terimler ontolojide sınıflar olacak ve

sınıf hiyerarşisindeki çapalar. ² Sınıfları hiyerarşik bir sınıflandırmaya göre düzenleriz. Bir sınıfın bir örneği olarak, nesnenin zorunlu olarak (yani tanım gereği) başka bir sınıfın bir örneği.

A sınıfı, B sınıfının bir süper sınıfıysa, o zaman her B örneği de bir A örneği

Başka bir deyişle, B sınıfı, bir "tür" A olan bir kavramı temsil eder.

Örneğin, her Pinot Noir şarabı mutlaka kırmızı bir şaraptır. Bu nedenle Pinot Noir sınıfı, Red Wine sınıfının bir alt sınıfıdır.

Şekil 2, Şarap ontolojisi için sınıf hiyerarşisinin bir bölümünü göstermektedir. Bölüm 4 bir Bir sınıf hiyerarşisini tanımlarken aranacak şeylerin ayrıntılı tartışması.

Şekil 3. Wine sınıfı için yuvalar ve bu yuvalar için yönler. Yapıcının yanındaki "I" simgesi slot, slotun tersi olduğunu gösterir (Bölüm 5.1)

Adım 5. Sınıfların özelliklerini tanımlayın - yuvalar

Sınıflar tek başına yeterlilik sorularını cevaplamak için yeterli bilgi sağlamayacaktır.

1. Adımdan itibaren. Sınıflardan bazılarını tanımladıktan sonra, iç sınıfları tanımlamalıyız. kavramların yapısı.

3. Adımda oluşturduğumuz terimler listesinden sınıfları zaten seçtik. geri kalan terimler muhtemelen bu sınıfların özellikleri olacaktır. Bu terimler şunları içerir: Örneğin, bir şarabın rengi, gövdesi, aroması ve şeker içeriği ve bir şarabın yeri şaraphane.

Listedeki her özellik için hangi sınıfı tanımladığını belirlemeliyiz. Bu özellikler sınıflara bağlı yuvalar haline gelir. Böylece, Wine sınıfı şu yuvalara sahip olacaktır: renk, vücut, lezzet ve şeker. Ve Winery sınıfı bir konum yuvasına sahip olacak.

Genel olarak, bir ontolojide yuva haline gelebilen birkaç nesne özelliği türü vardır:

- Bir şarabın tadı gibi “kendine özgü” özellikler;
- Bir şarabın adı ve geldiği bölge gibi “dışsal” özellikler;
- nesne yapılandırılmışsa parçalar; bunlar hem fiziksel hem de soyut "parçalar" olabilir (ör. yemek kursları)
- diğer bireylerle ilişkiler; bunlar bireyler arasındaki ilişkiler sınıfın üyeleri ve diğer öğeler (örneğin, bir şarabı temsil eden bir şarap ve bir şaraphane ile şarabın yapıldığı üzüm arasındaki ilişki.)

Bu nedenle, daha önce belirlediğimiz özelliklere ek olarak aşağıdakileri de eklememiz gerekir

² Sınıfları tekli yüklemeler olarak görebiliriz - tek argümanı olan sorular. Örneğin, "Bu mu şaraba itiraz etmek mi?" Tekli yüklemeler (veya sınıflar) ikili yüklemelerle (veya aralıklarla) zıttır — iki argüman. Örneğin, "Bu nesnenin tadı güçlü mü?" "Bu nesnenin tadı nedir?"

Wine sınıfına yuvalar: ad, alan, yapıcı, üzüm. Şekil 3, sınıf için slotları gösterir Şarap.

Bir sınıfın tüm alt sınıfları, o sınıfın yuvasını **miras alır**. Örneğin, sınıfın tüm slotları Şarap, Kırmızı Şarap ve Beyaz Şarap dahil olmak üzere tüm Şarap alt sınıflarına aktarılacaktır. Kırmızıya ek bir yuva, tanen seviyesi (düşük, orta veya yüksek) ekleyeceğiz. Şarap sınıfı. Tanen seviyesi yuva, kırmızıyı temsil eden tüm sınıflar tarafından miras alınacaktır. şaraplar (Bordeaux ve Beaujolais gibi).

Bu özelliğe sahip olabilecek en genel sınıfa bir yuva eklenmelidir. Örneğin, en genel olduğu için şarabın gövdesi ve rengi Wine sınıfına eklenmelidir. örnekleri gövde ve renge sahip olacak sınıf.

Adım 6. Yuvaların yönlerini tanımlayın

Slotlar, değer türünü, izin verilen değerleri, sayılarını açıklayan farklı yönlere sahip olabilir. değerleri (kardinalite) ve yuvanın alabileceği değerlerin diğer özellikleri. Örneğin, değer Bir ad yuvasının ("şarap adı" nda olduğu gibi) bir dizedir. Yani, ad değeri olan bir slottur String yazın. Üretilen bir slot ("bu şarapları üreten bir şaraphanede olduğu gibi") birden fazla değerler ve değerler Wine sınıfının örnekleridir. Yani üreten, değeri olan bir slottur İzin verilen sınıf olarak Wine ile Örnek yazın.

Şimdi birkaç ortak yönü açıklayacağız.

Slot kardinalitesi

Yuva önemliliği, bir yuvanın kaç değere sahip olabileceğini tanımlar. Bazı sistemler yalnızca ayırt eder tek önemlilik (en fazla bir değere izin verir) ve çoklu önem (herhangi bir değer sayısı). Bir şarap gövdesi, tek bir kardinalite yuvası olacaktır (bir şarap yalnızca bir vücut). Belirli bir şaraphane tarafından üretilen şaraplar, çoklu kardinalite yuvasını doldurur bir Winery sınıfı için üretir.

Bazı sistemler, minimum ve maksimum kardinalitenin tanımlanmasına izin verir. yuva değerlerinin sayısı daha kesin. N'nin minimum kardinalitesi, bir yuvanın sahip olması gerektiği anlamına gelir en az N değer. Örneğin, bir Şarabın üzüm yuvasının minimum kardinalitesi 1: her şarap en az bir çeşit üzümünden yapılır. M'nin maksimum kardinalitesi, a slot en fazla M değerine sahip olabilir. Tekli üzüm yuvası için maksimum kardinalite varietal şaraplar 1'dir: bu şaraplar yalnızca bir üzüm çeşidinden yapılır. Bazen olabilir maksimum kardinaliteyi 0'a ayarlamak yararlı olabilir. Bu ayar, yuvanın belirli bir alt sınıf için herhangi bir değere sahip olamaz.

Alan değeri türü

Değer türü bir façeta, yuvayı ne tür değerlerin doldurabileceğini açıklar. İşte daha fazlasının bir listesi ortak değer türleri:

- **Dize** , ad gibi alanlar için kullanılan en basit değer türüdür: değer bir basit dize
- **Sayı** (bazen daha spesifik Kayan ve Tamsayı değer türleri kullanılır) Sayısal değerlere sahip slotları tanımlar. Örneğin, bir şarabın fiyatı bir değere sahip olabilir Float yazın
- **Boole** yuvaları basit evet-hayır işaretleridir. Örneğin, temsil etmemeyi seçersek köpüklü şaraplar ayrı bir sınıf olarak, bir şarap köpüklü olsun veya olmasın Boole slotunun bir değeri olarak temsil edilir: değer "true" ("yes") ise şarap köpüklüdür ve değer "yanlış" ("hayır") ise, şarap köpüklü değildir.
- **Numaralandırılmış** yuvalar, yuva için izin verilen belirli değerlerin bir listesini belirtir. Örneğin, lezzet aralığının üç olası değerden birini alabileceğini belirtebiliriz:

türü Sembol.

- **Örnek** tipi yuvalar, bireyler arasındaki ilişkilerin tanımlanmasına izin verir. İle slotlar değer türü Örnek, örneklerin içerdiği izin verilen sınıfların bir listesini de tanımlamalıdır. gelebilir. Örneğin, Winery sınıfı için üretilen bir yuvanın örnekleri olabilir değerleri olarak Wine sınıfının. ³

Şekil 4, Winery sınıfında üretilen slotun bir tanımını göstermektedir.

Şekil 4. Bir yuvanın tanımı, bir şaraphane tarafından üretilen şarapları tanımlayan üretir. slot, kardinalite katna, değer türüne sahiptir ve bunun için izin verilen sınıf olarak Wine sınıfına sahiptir. değerler.

Alan ve alan aralığı

Örnek türündeki yuvalar için izin verilen sınıflara genellikle bir yuva **aralığı** denir . Örnekte Şekil 4'te Wine sınıfı, üretim aralığının aralığıdır. Bazı sistemler izin verir yuva belirli bir sınıf için takıldığında bir yuvanın aralığını sınırlamak.

Bir yuvanın eklendiği sınıflar veya bir yuvanın tanımladığı sınıflar denir alanın **etki alanı** . Winery sınıfı, üretim yuvasının etki alanıdır. İçinde yuvaları sınıflara *eklediğimiz* sistemler , genellikle yuvanın eklendiği sınıflar bu alanın etki alanını oluşturur. Alan adını ayrı ayrı belirtmenize gerek yoktur.

Bir alan ve bir alan aralığını belirlemek için temel kurallar benzerdir:

Bir alan için bir alan veya aralık tanımlarken, en genel sınıfları bulun veya yuvalar için sırasıyla alan veya aralık olabilen sınıf.

Öte yandan, aşırı bir alan ve aralık tanımlamayın.

genel: bir yuvanın etki alanındaki tüm sınıflar,

bir yuva aralığındaki tüm sınıfların yuvası ve örnekleri potansiyel olmalıdır yuva için doldurucular. Aralık için aşırı genel bir sınıf seçmeyin (ör.

biri THING aralığını yapmak istemezdi) ancak

tüm dolgu maddelerini kapsayacak bir sınıf seçin

Üretilen ürün yelpazesi için Wine sınıfının tüm olası alt sınıflarını listelemek yerine

³ Bazı sistemler, özel bir örnek ifadesi gerektirmek yerine yalnızca bir sınıfla değer türü belirtir yuvaları yazın.

Daha spesifik terimlerle;

Bir yuvanın bir aralığını veya etki alanını tanımlayan bir sınıf listesi bir sınıf içeriyorsa ve alt sınıfı, alt sınıfı kaldırın.

Slotun aralığı hem Wine sınıfını hem de Red Wine sınıfını içeriyorsa,

Kırmızı Şarabı menzilden çıkarın çünkü yeni bir bilgi eklemiyor:

Red Wine, Wine'ın bir alt sınıfıdır ve bu nedenle slot aralığı zaten bunu zımnen içermektedir. ve Wine sınıfının diğer tüm alt sınıfları.

Bir yuvanın bir aralığını veya etki alanını tanımlayan bir sınıf listesi tüm A sınıfının alt sınıfları, ancak A sınıfının kendisi değil, aralık şunları içermelidir sadece A sınıfı, alt sınıflar değil.

Slot menzilini Kırmızı Şarap, Beyaz Şarap ve Güllü içecek şekilde tanımlamak yerine

Wine (Wine'ın tüm alt sınıflarını numaralandırarak), aralığı sınıfla sınırlayabiliriz

Şarabın kendisi.

Bir yuvanın bir aralığını veya etki alanını tanımlayan bir sınıf listesi, bir A sınıfının birkaç alt sınıfı, A sınıfının daha fazla uygun aralık tanımı.

Bir sınıfa bir yuva eklemenin, sınıfın etki alanına eklenmesiyle aynı olduğu sistemlerde

yuva, yuva eki için de aynı kurallar geçerlidir: Bir yandan, bunu yapmaya çalışmalıyız

mümkün olduğunca genel. Öte yandan, bağladığımız her sınıfın

yuva, yuvanın temsil ettiği özelliğe sahip olabilir. Tanen ekleyebiliriz

kırmızı şarapları temsil eden sınıfların her biri için seviye aralığı (örneğin, Bordo, Merlot,

Beaujolais, vb.). Bununla birlikte, tüm kırmızı şaraplar tanen düzeyinde özelliğe sahip olduğundan,

bunun yerine yuvayı bu daha genel Kırmızı Şaraplar sınıfına takın. Etki alanını genellemek

tanen seviyesi yuvası daha ileride (bunun yerine Wine sınıfına iliştilerilerek)

örneğin beyaz şarapları tanımlamak için tanen seviyesini kullanmadığımız için doğru.

7. Adım. Örnek oluşturun

Son adım, hiyerarşide ayrı ayrı sınıf örnekleri oluşturmaktır. Bir bireyi tanımlamak

Bir sınıfın örneği, (1) bir sınıf seçmeyi, (2) bunun bireysel bir örneğini oluşturmayı gerektirir

sınıf ve (3) yuva değerlerinin doldurulması. Örneğin, bireysel bir örnek oluşturabiliriz

Chateau-Morgon-Beaujolais, belirli bir Beaujolais şarabı türünü temsil eder.

Chateau-Morgon-Beaujolais, Beaujolais sınıfının bir örneğidir.

tüm Beaujolais şarapları. Bu örnek aşağıdaki yuva değerlerine sahiptir (Şekil 5):

- Gövde: Işık
- Kırmızı renk
- Lezzet: Narin
- Tanen seviyesi: Düşük
- Üzüm: Gamay (Şaraplık üzüm sınıfının örneği)
- Maker: Chateau-Morgon (Winery sınıfının örneği)
- Bölge: Beaujolais (Wine-Region sınıfının örneği)

- Şeker: Kuru

Şekil 5. Beaujolais sınıfının bir örneğinin tanımı. Örnek Chateaux'dur

Beaujolais bölgesinden Morgon Beaujolais, Gamay üzümünden

Chateau Morgon şaraphanesi. Hafif bir gövdeye, narin bir tada, kırmızı renge ve düşük tanen seviyesine sahiptir. Bu bir sek şarap.

4 Sınıfları ve bir sınıf hiyerarşisini tanımlama

Bu bölümde dikkat edilmesi gerekenler ve tanımlanırken yapılması kolay olan hatalar tartışılmaktadır.

sınıflar ve bir sınıf hiyerarşisi (Bölüm 3'ten Adım 4). Daha önce de bahsettiğimiz gibi var

herhangi bir alan için tek bir doğru sınıf hiyerarşisi yoktur. Hiyerarşi şunlara bağlıdır:

ontolojinin olası kullanımları, uygulama için gerekli olan detay seviyesi,

kişisel tercihler ve bazen diğer modellerle uyumluluk için gereksinimler.

Bununla birlikte, bir sınıf hiyerarşisi geliştirirken akılda tutulması gereken birkaç yönergeyi tartışıyoruz.

Önemli sayıda yeni sınıf tanımladıktan sonra, geri çekilip

ortaya çıkan hiyerarşi bu yönergelere uygundur.

4.1 Sınıf hiyerarşisinin doğru olmasını sağlamak

Bir "is-a" ilişkisi

Sınıf hiyerarşisi bir "is-a" ilişkisini temsil eder: a sınıfı, her örnek B'nin bir alt sınıfıdır

A da bir B örneğidir. Örneğin, Chardonnay Beyaz şarabın bir alt sınıfıdır.

Taksonomik ilişkiyi düşünmenin başka bir yolu da "tür" bir ilişki gibidir: Chardonnay

bir çeşit Beyaz şarap. Jetliner bir tür uçaktır. Et bir çeşit besindir.

*Bir sınıfın alt sınıfı, kavramın "türü" olan bir kavramı temsil eder
üst sınıfın temsil ettiği.*

Tek bir şarap, tüm şarapların alt sınıfı değildir

Yaygın bir modelleme hatası, aynı şeyin hem tekil hem de çoğul versiyonunu eklemektir.

hiyerarşideki kavram, birinciyi ikincisinin bir alt sınıfı haline getirir. Mesela yanlış

Şarapların bir alt sınıfı olarak bir Şarap sınıfı ve bir Şarap sınıfı tanımlamak için. Bir kez düşündüğünüzde

hiyerarşi "tür" ilişkiyi temsil ederken, modelleme hatası netleşir: a

tek Şarap bir **tür** Şarap değildir. Böyle bir hatayı önlemenin en iyi yolu her zaman kullanmaktır.

sınıfların isimlendirilmesinde tekil veya çoğul (isimlendirme tartışması için Bölüm 6'ya bakın)

kavramlar).

Hiyerarşik ilişkilerin geçişkenliği

Bir alt sınıf ilişkisi geçişlidir:

B, A'nın bir alt sınıfı ise ve C, B'nin bir alt sınıfı ise, C, A'nın bir alt sınıfıdır.

Örneğin, bir Şarap sınıfı tanımlayabiliriz ve ardından bir alt sınıf olarak bir Beyaz şarap sınıfı tanımlayabiliriz.

şarabın. Sonra bir Chardonnay sınıfını Beyaz şarabın bir alt sınıfı olarak tanımlıyoruz. Geçişlilik alt sınıf ilişkisi, Chardonnay sınıfının aynı zamanda Wine'in bir alt sınıfı olduğu anlamına gelir. Bazen doğrudan alt sınıflar ile dolaylı alt sınıflar arasında ayrım yaparız. Bir **direkt alt sınıf** sınıfın "en yakın" alt sınıfıdır: bir sınıf ile onun doğrudan alt sınıfı arasında sınıf yoktur bir hiyerarşi içinde. Yani, bir sınıf ile sınıf arasındaki hiyerarşide başka sınıf yoktur. doğrudan süper sınıf. Örneğimizde, Chardonnay, Beyaz şarabın doğrudan bir alt sınıfıdır ve Wine'in doğrudan bir alt sınıfı değil.

Bir sınıf hiyerarşisinin evrimi

Etki alanları geliştikçe tutarlı bir sınıf hiyerarşisini sürdürmek zor hale gelebilir. İçin Örneğin, uzun yıllar boyunca tüm Zinfandel şarapları kırmızıydı. Bu nedenle, bir sınıf tanımlıyoruz Zinfandel, Kırmızı şarap sınıfının bir alt sınıfı olarak şaraplar. Ancak bazen şarap üreticileri üzümeleri sıkıştırmaya ve üzümelerin renk veren yönlerini ortadan kaldırmaya başladı hemen, böylece ortaya çıkan şarabın rengini değiştirir. Böylece "beyaz" oluruz zinfandel "rengi gül olan. Şimdi Zinfandel sınıfını iki sınıfa ayırmamız gerekiyor Beyaz zinfandel ve Red zinfandel olarak sınıflandırılır ve bunları Sırasıyla Rose şarabı ve Kırmızı şarabın alt sınıfları.

Sınıflar ve isimleri

Bir sınıf ve adını ayırt etmek önemlidir:

Sınıflar, alandaki kavramları temsil eder, ifade eden kelimeleri değil bu kavramlar.

Farklı bir terminoloji seçersek sınıfın adı değişebilir, ancak terimin kendisi dünyadaki nesnel gerçekliği temsil eder. Örneğin, bir Karides sınıfı oluşturabiliriz, ve sonra onu Karides olarak yeniden adlandırın — sınıf hala aynı kavramı temsil ediyor. Uygun Karides yemeklerine atıfta bulunan şarap kombinasyonları, karides yemeklerine atıfta bulunmalıdır. Daha fazlası pratik terimler, aşağıdaki kurala her zaman uyulmalıdır:

Aynı kavram için eş anlamlılar farklı sınıfları temsil etmez

Eş anlamlılar, bir kavram veya terim için yalnızca farklı adlardır. Bu nedenle, gereken **değil** bir var Karides adlı bir sınıf ve Karides adlı bir sınıf ve muhtemelen Crevette adlı bir sınıf. Aksine, Karides veya Karides adında bir sınıf vardır. Birçok sistem ilişkilendirmeye izin verir bir sınıfla eş anlamlıların, çevirilerin veya sunum adlarının listesi. Bir sistem yoksa Bu ilişkilendirmelere izin verin, eş anlamlılar her zaman sınıf belgelerinde listelenebilir.

Sınıf döngülerinden kaçınmak

Sınıf hiyerarşisindeki **döngülerden** kaçınmalıyız . Bir hiyerarşide bir döngü olduğunu söylüyoruz bazı A sınıfının bir B alt sınıfı olduğu ve aynı zamanda B, A'nın bir üst sınıfı olduğunda Hiyerarşideki bir döngü, A ve B sınıflarının eşdeğer olduğunu bildirmek anlamına gelir: tümü A örnekleri, B örnekleridir ve tüm B örnekleri aynı zamanda A örnekleridir. Aslında, çünkü B, A'nın bir alt sınıfıdır, tüm B'nin örnekleri A sınıfının örnekleri olmalıdır. A, A'nın bir alt sınıfı olduğu için B, tüm A örneklerinin aynı zamanda B sınıfının örnekleri olması gerekir.

4.2 Bir sınıf hiyerarşisinde kardeşleri analiz etmek

Bir sınıf hiyerarşisindeki kardeşler

Hiyerarşideki **kardeşler** , aynı sınıfın doğrudan alt sınıfları olan sınıflardır (bkz.Bölüm 4.1).

Hiyerarşideki tüm kardeşler (kökte olanlar hariç) şu konumda olmalıdır: aynı düzeyde genellik.

Örneğin, White wine ve Chardonnay aynı sınıftaki alt sınıflar olmamalıdır. (örneğin, Şarap). Beyaz şarap, Chardonnay'den daha genel bir kavramdır. Kardeşler aynı seviyedeki bölümlerle aynı şekilde "aynı çizgi boyunca" düşen kavramları temsil eder bir kitapta aynı genellik düzeyindedir. Bu anlamda, bir sınıf hiyerarşisi için gereksinimler kitap taslağı gereksinimlerine benzer.

Ancak hiyerarşinin kökündeki kavramlar (bunlar genellikle doğrudan Thing gibi çok genel bir sınıfın alt sınıfları, sınıfın ana bölümlerini temsil eder. etki alanı ve benzer kavramlar olması gerekmez.

Kaçı çok fazla ve kaç çok az?

Bir sınıfın sahip olması gereken doğrudan alt sınıfların sayısı için kesin kurallar yoktur. Ancak, birçok iyi yapılandırılmış ontolojinin iki ila bir düzine arasında doğrudan alt sınıfı vardır. Bu nedenle, aşağıdaki iki yönergeye sahibiz:

Bir sınıfın yalnızca bir doğrudan alt sınıfı varsa, bir modelleme sorunu olabilir veya ontoloji tamamlanmadı.

Belirli bir sınıf için bir düzineden fazla alt sınıf varsa, o zaman ek ara kategoriler gerekli olabilir.

İki kuraldan ilki, madde işaretli listelerin asla sahip olmaması gereken bir dizgi kuralına benzer. sadece bir madde işareti. Örneğin, kırmızı Burgundy şaraplarının çoğu Côtes d'Or şaraplarıdır. Farz edelim ki, sadece bu tür Burgundy şaraplarını temsil etmek istiyoruz. Bir yaratabiliriz Kırmızı Bordo sınıfı ve ardından tek bir alt sınıf Cotes d'Or (Şekil 6a). Ancak, eğer temsilimiz kırmızı Burgundy ve Côtes d'Or şarapları esasen eşdeğerdir (tümü kırmızı Bordo şarapları Côtes d'Or şaraplarıdır ve tüm Côtes d'Or şarapları kırmızı Burgundy şaraplarıdır), Cotes d'Or sınıfının oluşturulması gerekli değildir ve herhangi bir yeni bilgi eklemeyiz. sunum. Daha ucuz olan Côtes Chalonnaise şaraplarını dahil edersek Côtes d'Or'un hemen güneyindeki bölgeden bordo şaraplar, sonra iki alt sınıf oluşturacağız Burgundy sınıfının: Cotes d'Or ve Cotes Chalonnaise (Şekil 6b).

Şekil 6. Red Burgundy sınıfının alt sınıfları. Genellikle bir sınıfın tek bir alt sınıfına sahip olmak modellemede bir soruna işaret eder.

Şimdi tüm şarap türlerini Wine sınıfının doğrudan alt sınıfları olarak listelediğimizi varsayalım. Bu liste daha sonra Beaujolais ve Bordeaux gibi daha genel şarap türlerini ve Paulliac ve Margaux gibi daha spesifik tipler (Şekil 6a). Wine sınıfında çok fazla var doğrudan alt sınıflar ve aslında ontolojinin farklı şarap türlerini daha fazla yansıtmaları için Organize bir şekilde Medoc, Bordeaux'nun bir alt sınıfı olmalı ve Cotes d'Or,

Burgundy'nin alt sınıfı. Ayrıca Kırmızı şarap ve Beyaz gibi ara kategorilere sahip şarap aynı zamanda birçok insanın sahip olduğu şarap alanının kavramsal modelini de yansıtmaktadır. (Şekil 6b).

Bununla birlikte, uzun kardeşler listesindeki kavramları gruplamak için doğal sınıflar yoksa, yapay sınıflar yaratmanız gerekiyor - sadece sınıfları olduğu gibi bırakın. Sonuçta, ontoloji gerçek dünyanın bir yansımasıdır ve gerçek dünyada herhangi bir sınıflandırma yoksa, o zaman ontoloji bunu yansıtmalıdır.

Şekil 7. Şarapların sınıflandırılması. Tüm şaraplara ve şarap türlerine sahip olmak ve çeşitli düzeylerde şaraplara sahip olmak kategorizasyon.

4.3 Çoklu miras

Çoğu bilgi temsili sistemi , sınıf hiyerarşisinde **çoklu kalıtıma** izin verir :
bir sınıf, birkaç sınıfın bir alt sınıfı olabilir. Aynı bir sınıf oluşturmak istediğimizi varsayalım.
tatlı şaraplar, Tatlı şarap sınıfı. Porto şarabı hem kırmızı şarap hem de tatlıdır

şarap. ⁴ Bu nedenle, iki süper sınıfa sahip bir Port sınıfı tanımlıyoruz: Kırmızı şarap ve Tatlı şarap. Port sınıfının tüm örnekleri, hem Kırmızı şarabın örnekleri olacaktır. sınıfı ve Tatlı şarap sınıfı. Port sınıfı, yuvalarını ve yönlerini devralacak her iki ebeveyninden. Böylece, Şeker yuvası için TATLI değerini Tatlı şarap sınıfı ve tanen seviyesi yuvası ve renk yuvasının değeri Kırmızı şarap sınıfı.

4.4 Ne zaman yeni bir sınıf tanıtılmalı (veya verilmemeli)

Modelleme sırasında verilmesi en zor kararlardan biri, yeni bir sınıfın ne zaman tanıtılacağı veya farklı özellik değerleri aracılığıyla bir ayrımın ne zaman temsil edileceği. İkisinde de gezinmek zor birçok yabancı sınıf ve çok düz bir hiyerarşi ile son derece iç içe geçmiş bir hiyerarşi

yuvalarda kodlanmış çok fazla bilgi içeren çok az sınıfa sahip. Uygun olanı bulmak dengelemek kolay değil.

Yeni sınıfları ne zaman tanıtacağınıza karar vermenize yardımcı olan birkaç pratik kural vardır. hiyerarşi.

*Bir sınıfın alt sınıfları genellikle (1),
süper sınıfın (2) kısıtlamaları yoktur veya (2)
üst sınıf veya (3) farklı ilişkilere katılır
üst sınıflar*

Kırmızı şaraplar farklı seviyelerde tanenlere sahip olabilir, oysa bu özellik açıklamak için kullanılmaz. genel olarak şaraplar. Tatlı şarabının şeker yuvasının değeri TATLI iken, Tatlı Şarap sınıfının süper sınıfı için geçerlidir. Pinot Noir şarapları deniz ürünleri ile iyi gidebilir diğer kırmızı şaraplar ise yok. Başka bir deyişle, hiyerarşide yeni bir sınıf tanıtıyoruz genellikle sadece bu sınıf hakkında söyleyemeyeceğimiz bir şey olduğunda süper sınıf hakkında.

Pratik açıdan, her alt sınıfın ya yeni yuvalar eklemesi ya da yeni yuvaya sahip olması gerekir. değerler tanımlanır veya devralınan alanlar için bazı özellikleri geçersiz kılar.

Bununla birlikte, bazen yeni sınıflar oluşturmasalar bile yeni sınıflar oluşturmak faydalı olabilir. yeni özellikler.

*Terminolojik hiyerarşilerdeki sınıfların yeni
özellikleri*

Örneğin, bazı ontolojiler, kullanılan ortak terimlerin geniş referans hiyerarşilerini içerir. alan adı. Örneğin, bir elektronik tıbbi kayıt sisteminin altında yatan bir ontoloji, çeşitli hastalıkların sınıflandırılmasını içerir. Bu sınıflandırma sadece bir hiyerarşi olabilir özellikleri olmayan (veya aynı özellik kümesiyle) terimler. Bu durumda, yine de faydalıdır terimleri düz bir liste yerine bir hiyerarşide düzenlemek, çünkü (1) daha kolay keşif ve seyrüsefer ve (2) bir doktorun kolaylıkla bir genellik düzeyi duruma uygun terim.

Herhangi bir yeni özellik olmadan yeni sınıflar sunmanın bir başka nedeni de konseptleri sahip olsak bile hangi alan uzmanları genellikle bir ayırım yapar? ayırımın kendisini modellememeye karar verdi. Kolaylaştırmak için ontolojiler kullandığımızdan alan uzmanları arasında ve alan uzmanları ile bilgiye dayalı iletişim sistemlere uzmanların ontolojideki alan hakkındaki görüşlerini yansıtmak istiyoruz.

Son olarak, her ek kısıtlama için bir sınıfın alt sınıflarını oluşturmamalıyız. İçin Örneğin, Kırmızı şarap, Beyaz şarap ve Gül şarabı sınıflarını tanıttık çünkü bu ayırım şarap dünyasında doğal bir ayırımdır. Hassas sınıflar uygulamadık

⁴ Ontolojimizde yalnızca kırmızı Portları temsil etmeyi seçtik: beyaz Limanlar var ama bunlar son derece nadir.

şarap, ılımlı şarap vb. Bir sınıf hiyerarşisi tanımlarken, amacımız bir sınıf organizasyonu için yararlı yeni sınıflar oluşturmakla çok fazla oluşturmak arasında denge sınıflar.

4.5 Yeni bir sınıf mı yoksa özellik değeri mi?

Bir alanı modellerken, genellikle belirli bir ayırımı modelleyip modellememeye karar vermemiz gerekir. (beyaz, kırmızı veya roze şarabı gibi) bir özellik değeri olarak veya bir sınıf seti olarak yine şunlara bağlıdır: etki alanının kapsamı ve eldeki görev.

Sınıf bir Beyaz şarap mı yaratıyoruz yoksa sadece bir sınıf Şarap mı oluşturup farklı yuva rengi için değerler? Cevap genellikle için tanımladığımız kapsamdadır. ontoloji. Beyaz şarap kavramı bizim alanımızda ne kadar önemli? Şaraplar varsa etki alanında sadece marjinal önemi ve şarabın beyaz olup olmaması diğer nesnelerle ilişkileri için herhangi bir özel çıkarım varsa, o zaman bir

beyaz şaraplar için ayrı sınıf. Şarap etiketleri üreten bir fabrikada kullanılan bir alan modeli için, herhangi bir renkteki şarap etiketleri için kurallar aynıdır ve ayırım çok önemli değildir. Alternatif olarak, şarap, yemek ve bunların uygun kombinasyonlarının temsili için kırmızı şarap beyaz şaraptan çok farklıdır: farklı yiyeceklerle eşleştirilir, farklı özellikler vb. Benzer şekilde, şarap rengi, şarap bilgi tabanı için önemlidir. şarap tadım sırasını belirlemek için kullanabiliriz. Böylece Beyaz için ayrı bir sınıf oluşturuyoruz şarap.

Farklı yuva değerlerine sahip kavramlar, farklı diğer sınıflardaki yuvalar; o zaman ayırım için yeni bir sınıf oluşturmalıyız. Aksi takdirde, bir slot değerindeki farkı temsil ederiz.

Benzer şekilde, şarap ontolojimiz, Kırmızı Merlot ve White Merlot gibi sınıflara sahiptir. tüm Merlot şarapları için tek bir sınıftan daha: kırmızı Merlotlar ve beyaz Merlotlar gerçekten farklı şaraplar (aynı üzümden yapılmış) ve şarap için ayrıntılı bir ontoloji geliştiriyorsak, bu ayırım önemlidir.

Etki alanında bir ayırım önemliyse ve aşağıdaki nesneleri düşünürsek farklı nesne türleri olarak ayırım için farklı değerler, ayırım için yeni bir sınıf oluşturmalıdır.

Bir sınıfın potansiyel bireysel örneklerini göz önünde bulundurmak, aynı zamanda ya da yeni bir sınıf başlatmamak.

Tek bir örneğin ait olduğu bir sınıf sık sık değişmemelidir.

Genellikle kavramların içsel özelliklerinden ziyade dışsal özellikleri kullandığımızda sınıflar arasında, bu sınıfların örnekleri genellikle bir sınıftan diğerine geçmek zorunda kalacaktır. bir diğeri. Örneğin, soğutulmuş şarap, şarabı tanımlayan bir ontoloji sınıfı olmamalıdır. bir restoranda şişeler. Soğutulmuş mülk, sadece bir şarapta şarabın bir niteliği olmalıdır. Soğutulmuş şarap bu sınıfın bir örneği olmaktan kolayca çıkabildiğinden şişe ve sonra tekrar bu sınıfın bir örneği haline gelir.

Genellikle sayılar, renkler, konumlar yuva değerleridir ve yeni sınıflar. Bununla birlikte, şarap dikkate değer bir istisnadır çünkü şarabın rengi şarap için çok önemlidir. şarabın tanımı.

Başka bir örnek olarak, insan anatomisi ontolojisini düşünün. Kaburgaları temsil ettiğimizde, “1 her biri için bir sınıf oluşturmak st kaburga sol”, “2 nd sol kaburga” ve benzeri? Veya bir sınıf kaburga var mı sipariş ve yanal konum için yuvaları olan (sol-sağ)? 5 Her biri hakkında bilgi ontolojide temsil ettiğimiz kaburgalar önemli ölçüde farklıdır, o zaman gerçekten de

5 Burada her anatomik organ sınıf olduğunu varsayalım biz de John'un 1” hakkında konuşmak istiyorum çünkü st sol kaburga. " Var olan insanların bireysel organları ontolojimizde bireyler olarak temsil edilecektir.

her bir kaburga için bir sınıf oluşturun. Yani, ayrıntıları bitişik olarak temsil etmek istiyorsak ve konum bilgisi (her bir nervür için farklıdır) ve her birinin rib playa ve koruduğu organları, sınıfları istiyoruz. Anatomiye biraz modelliyorsak daha az genellik seviyesi ve potansiyel uygulamalarımız kadar tüm nervürler çok benzer endişeli (sadece röntgende hangi kaburganın kırıldığından bahsediyoruz. vücudun diğer bölümleri), hiyerarşimizi basitleştirmek ve sadece Rib sınıfına sahip olmak isteyebiliriz, iki yuvalı: yanal konum, sıra.

4.6 Bir örnek mi, sınıf mı?

Belirli bir kavramın ontolojideki bir sınıf mı yoksa bireysel bir örnek mi olduğuna karar vermek ontolojinin potansiyel uygulamalarının ne olduğuna bağlıdır. Derslerin nerede biteceğine karar vermek ve tek tek örnekler, en düşük ayrıntı düzeyinin ne olduğuna karar vermekle başlar. temsil. Parçacık düzeyi, sırayla, potansiyel bir uygulama tarafından belirlenir. ontoloji. Başka bir deyişle, en spesifik öğeler nelerdir? bilgi tabanında temsil ediliyor mu? İçinde belirlediğimiz yetkinlik sorularına geri dönersek

Bölüm 3'teki 1. Adım, bu sorulara yanıt oluşturacak en spesifik kavramlar bilgi tabanındaki bireyler için çok iyi adaylardır.

Bireysel örnekler, bir bilgi tabanı.

Örneğin, sadece şarabı yiyeceklerle eşleştirmekten bahsedeceksek, belirli fiziksel şarap şişeleriyle ilgileniyor. Bu nedenle, Sterling gibi terimler Merlot Üzüm Bağları, muhtemelen kullandığımız en spesifik terimler olacaktır. Diğer Şarap sınıfı, tek tek şarap şişelerinden değil, daha çok şaraplardan oluşan bir koleksiyondur. belirli şarap imalathaneleri tarafından üretilen özel şaraplar. Bu nedenle, Sterling Bağları Merlot, bilgi tabanında bir örnek olacaktır.

Öte yandan, restoranda bir şarap envanterini tutmak istersek, İyi şarap-yemek eşleştirmelerinin bilgi tabanına ek olarak, her şarabın ayrı şişeleri bilgi tabanımızda bireysel örnekler haline gelebilir.

Benzer şekilde, her bir belirli ürün için farklı özellikler kaydetmek istiyorsak Sterling Vineyards Merlot, o zaman şarabın belirli bir mahsulü, bir bilgi tabanı ve Sterling Vineyards Merlot, aşağıdakiler için örnekler içeren bir sınıftır: tüm vintages.

Başka bir kural, bazı bireysel örnekleri sınıf kümesine "taşıyabilir":

Kavramlar doğal bir hiyerarşi oluşturuyorsa, onları şu şekilde temsil etmeliyiz:
sınıflar

Şarap bölgelerini düşünün. Başlangıçta, Fransa, Birleşik Krallık gibi ana şarap bölgelerini tanımlayabiliriz. Eyaletler, Almanya vb. Sınıflar ve bu büyük bölgelerdeki belirli şarap bölgeleri olarak örnekler. Örneğin, Bourgogne bölgesi, Fransız bölgesinin bir örneğidir sınıf. Ancak Cotes d'Or bölgesinin bir Bourgogne bölgesi. Bu nedenle, Bourgogne bölgesi bir sınıf olmalıdır (alt sınıflara veya örneklere sahip). Ancak, Bourgogne bölgesini bir sınıf ve Cotes yapmak d'Or bölgesi, Bourgogne bölgesinin bir örneği gelişigüzel görünüyor: hangi bölgelerin sınıf ve hangilerinin örnek olduğunu açıkça ayırt edin. Bu nedenle hepsini tanımlıyoruz sınıf olarak şarap bölgeleri. Protégé-2000, kullanıcıların bazı sınıfları Özet olarak belirlemesine izin verir, sınıfın herhangi bir doğrudan örneğe sahip olamayacağını belirtir. Bizim durumumuzda, tüm bölge sınıfları özet (Şekil 8).

Sekil 8. Şarap bölgelerinin hiyerarşisi. Sınıf adlarının yanındaki "A" simgeleri, sınıfların soyut ve doğrudan örnekleri olamaz.

Sınıftan "bölge" kelimesini çıkarırsak aynı sınıf hiyerarşisi yanlış olur. isimler. Alsace sınıfının Fransa sınıfının bir alt sınıfı olduğunu söyleyemeyiz: Alsace bir tür Fransa değil. Ancak Alsas bölgesi bir tür Fransız bölgesidir.

Yalnızca sınıflar bir hiyerarşi içinde düzenlenebilir - bilgi temsil sistemlerinin bir alt örnek kavramı. Bu nedenle, terimler arasında doğal bir hiyerarşi varsa, örneğin Bölüm 4.2'deki terminolojik hiyerarşiler, bu terimleri sınıflar olarak tanımlamalıyız ancak kendilerine ait herhangi bir örneğe sahip olmayabilirler.

4.7 Kapsamı sınırlandırma

Bir sınıf hiyerarşisinin tanımlanmasına ilişkin son bir not olarak, aşağıdaki kurallar dizisi her zaman ontoloji tanımının ne zaman tamamlandığına karar vermek:

Ontoloji, hakkında olası tüm bilgileri içermemelidir.

etki alanı: ihtiyacınız olandan daha fazla uzmanlaşmanıza (veya genellenenize) gerek yok başvurunuz için (her şekilde en fazla bir ekstra seviye).

Şarap ve yiyecek örneğimiz için, etiketler için hangi kağıdın kullanıldığını bilmemize gerek yok. karides yemekleri nasıl pişirilir.

Benzer şekilde,

Ontoloji, tüm olası özelliklerini içermemelidir ve

hiyerarşideki sınıflar arasındaki farklar.

Ontolojimizde, bir şarabın veya yiyeceğin sahip olabileceği tüm özellikleri kesinlikle dahil etmiyoruz.

Ontolojimizdeki öge sınıflarının en göze çarpan özelliklerini temsil ettik. Hatta Şarap kitapları bize üzümlerin büyüklüğünü söylese de bu bilgiye yer vermedik. Benzer şekilde, aklınıza gelebilecek tüm ilişkileri tüm terimler arasına eklemedik. bizim sistemimiz. Örneğin, en sevdiğimiz şarap gibi ilişkileri dahil etmiyoruz ve ontolojide en sevilen yiyecek sadece tümünün daha eksiksiz bir temsiline izin vermek için tanımladığımız terimler arasındaki bağlantılar.

Son kural, halihazırda sahip olduğumuz kavramlar arasında ilişkiler kurmak için de geçerlidir. ontolojiye dahildir. Biyoloji deneylerini açıklayan bir ontoloji düşünün. Ontoloji Muhtemelen bir Biyolojik organizma kavramı içerecektir. Ayrıca bir konsept içerecek bir deney yapan bir Deneycinin (adı, bağlılığı vb.). Bu doğru

bir kişi olarak bir deneyci de biyolojik bir organizma olur. Ama, biz muhtemelen bu ayrımı ontolojiye dahil etmemelidir: bunun amaçları için bir deneycinin temsili biyolojik bir organizma değildir ve muhtemelen asla deneycilerin kendileri üzerinde deneyler yapın. Her şeyi temsil ediyor olsaydık ontolojideki sınıflar hakkında söyleyebiliriz, bir Deneyci bir alt sınıf olur Biyolojik Organizma. Ancak, bu bilgiyi dahil etmemize gerek yoktur. öngörülebilir uygulamalar. Aslında, mevcut olanlar için bu tür ek sınıflandırma dahil sınıflar gerçekten acı veriyor: Artık bir Experimenter örneğinde ağırlık, yaş, türler ve biyolojik bir organizmaya ait diğer veriler, ancak kesinlikle ilgisiz bir deneyi açıklama bağlamı. Ancak, böyle bir tasarım kararını bu ontolojiye bakacak olan ve bunu yapabilecek kullanıcıların yararına dokümantasyon aklımızdaki uygulamadan haberdar olmayın. Aksi takdirde, yeniden kullanmak isteyenler diğer uygulamalar için ontoloji deneyciyi bir kişinin alt sınıfı olarak kullanmaya çalışabilir. orijinal modellemenin bu gerçeği içermediğini bilmek.

4.8 Ayrık alt sınıflar

Birçok sistem, birkaç sınıfın **ayrık** olduğunu açıkça belirtmemize izin verir . Sınıflar ortak herhangi bir örneğe sahip olamazlarsa ayrık. Örneğin, Tatlı şarap

ve ontolojimizdeki Beyaz şarap sınıfları ayırık *değildir* : birçok şarap vardır. her ikisinin örnekleridir. Rothermel Trochenbierenauslese Riesling Sweet Riesling sınıfı örneği böyle bir örnektir. Aynı zamanda Kızıl şarap ve Beyaz şarap sınıfları birbirlerinden ayrılmaz: hiçbir şarap aynı anda kırmızı ve beyaz. Sınıfların ayırık olduğunu belirtmek, sistemin ontolojiyi daha iyi doğrulamasını sağlar. Kırmızı şarap ve Beyaz şarap sınıflarının ayırık olduğunu ilan edip daha sonra bir Hem Riesling'in (Beyaz şarabın bir alt sınıfı) hem de Port'un (a) alt sınıfı olan sınıf Kırmızı şarap alt sınıfı), bir sistem modelleme hatası olduğunu gösterebilir.

5 Özellikleri tanımlama - daha fazla ayrıntı

Bu bölümde, içindeki yuvaları tanımlarken akılda tutulması gereken birkaç ayrıntı daha tartışacağız. ontoloji (Bölüm 3, Adım 5 ve Adım 6). Temel olarak, ters yuvaları ve varsayılan değerleri tartışıyoruz bir yuva için.

5.1 Ters yuvalar

Bir yuvanın değeri, başka bir yuvanın değerine bağlı olabilir. Örneğin, bir şarap bir şaraphane tarafından üretilirse, o şarabı imal eder. Bu iki ilişki, yapan ve üreten, **ters ilişkiler** olarak adlandırılır . Bilginin "her ikisinde de saklanması "yönler" gereksizdir. Bir şarabın bir şaraphane tarafından üretildiğini bildiğimizde, bilgi tabanını kullanmak, şaraphanenin ters ilişkisinin değerini her zaman çıkarabilir. şarabı üretir. Bununla birlikte, bilgi edinme açısından bakıldığında, her iki bilginin de açıkça mevcut olması. Bu yaklaşım, kullanıcıların bir durumda şarap, diğerinde şaraphane. Bilgi edinme sistemi daha sonra bilginin tutarlılığını garanti altına alarak ters ilişki için değeri otomatik olarak doldurun taban.

Örneğimizde bir çift ters yuva var: Wine sınıfının maker yuvası ve Winery sınıfının yuvasını üretir. Bir kullanıcı Wine sınıfının bir örneğini oluşturduğunda ve üretici yuvasının değerini doldurur, sistem otomatik olarak yeni oluşturulan ilgili Winery örneğinin üretim yuvasına örnek. Örneğin, ne zaman Sterling Merlot'un Sterling Vineyard şaraphanesi tarafından üretildiğini söylüyoruz, sistem

Sterling Vineyard şaraphanesinin şarap listesine otomatik olarak Sterling Merlot'u ekler üretir. (Şekil 9).

Şekil 9. Ters yuvalara sahip örnekler. Slot, Winery sınıfı için üretilir.

Wine sınıfı için slot makinesi. Yuvalardan birini doldurmak, diğerinin otomatik olarak güncellenmesini tetikler.

5.2 Varsayılan değerler

Çerçeve tabanlı sistemlerin çoğu, yuvalar için varsayılan değerlerin belirtilmesine izin verir. Belirli bir yuva değer bir sınıfın çoğu örneği için aynıdır, bu değeri **varsayılan bir değer** olarak tanımlayabiliriz yuva için. Ardından, bu yuvayı içeren bir sınıfın her yeni örneği oluşturulduğunda, sistem varsayılan değeri otomatik olarak doldurur. Daha sonra değeri başka herhangi bir şekilde değiştirebiliriz yönlerin izin vereceği değer. Yani, varsayılan değerler kolaylık sağlamak içindir: model üzerinde herhangi bir yeni kısıtlama uygulamak veya modeli herhangi bir şekilde değiştirmek.

Örneğin, tartışacağımız şarapların çoğu tam gövdeli şaraplar ise, şarabın gövdesi için varsayılan değer olarak "dolu" olmalıdır. Sonra, aksini söylemedikçe, hepsi Tanımladığımız şaraplar tam gövdeli olacaktır.

Bunun **yuva değerlerinden** farklı olduğunu unutmayın . Slot değerleri değiştirilemez. Örneğin biz slot şekerin Tatlı şarap sınıfı için TATLI değerine sahip olduğunu söyleyebiliriz. Sonra hepsi Tatlı şarap sınıfının alt sınıfları ve örnekleri, TATLI değerine sahip olacaktır. yuva şekeri. Bu değer, sınıfın herhangi bir alt sınıfında veya örneğinde değiştirilemez.

6 Bir isimde ne var?

Bir ontolojideki kavramlar için adlandırma kurallarının tanımlanması ve ardından bunlara kesinlikle bağlı kalınması gelenekler yalnızca ontolojinin anlaşılmasını kolaylaştırmakla kalmaz, aynı zamanda bazı yaygın modelleme hataları. Kavramları adlandırmada birçok alternatif var. Genellikle vardır bir veya başka bir alternatif seçmek için özel bir neden yok. Ancak, ihtiyacımız var

Sınıflar ve yuvalar için bir adlandırma kuralı tanımlayın ve buna uyun.

Bir bilgi temsil sisteminin aşağıdaki özellikleri, adlandırma seçimini etkiler

sözleşmeler:

- Sistem, sınıflar, yuvalar ve örnekler için aynı ad alanına sahip mi? Yani, sistem aynı ada sahip bir sınıfa ve yuvaya (sınıf gibi) izin veriyor mu? şaraphane ve kumarhane)?
- Sistem büyük / küçük harfe duyarlı mı? Yani, sistem yalnızca farklı olan isimleri ele alıyor mu? farklı isimler olması durumunda (Şaraphane ve şaraphane gibi)?
- Sistem adlarda hangi sınırlayıcılara izin veriyor? Yani isimler içerebilir boşluklar, virgüller, yıldız işaretleri vb.

Örneğin Protégé-2000, tüm çerçeveleri için tek bir isim alanı tutar. Durum-hassas. Bu nedenle, bir sınıf şaraphanesi ve bir kumarhanemiz olamaz. Bununla birlikte, sahip olabiliriz bir sınıf Şaraphane (büyük harflere dikkat edin) ve bir slot şaraphanesi. CLASSIC ise büyük / küçük harfe duyarlı değildir ve sınıflar, yuvalar ve bireyler için farklı ad alanları korur. Böylece, sistem açısından bakıldığında, hem bir sınıfı hem de bir yuvayı adlandırmada sorun yoktur. Şaraphane.

6.1 Büyük harf kullanımı ve sınırlayıcılar

İlk olarak, tutarlı kullanırsak bir ontolojinin okunabilirliğini büyük ölçüde artırabiliriz. kavram adları için büyük harf kullanımı. Örneğin, sınıf adlarını büyük harfle yazmak yaygındır ve yuva adları için küçük harf kullanın (sistemin büyük / küçük harfe duyarlı olduğu varsayılarak).

Bir kavram adı birden fazla kelime içerdiğinde (Yemek dersi gibi) kelimeleri sınırlandırın. İşte bazı olası seçenekler.

- Boşluğu Kullanın: Yemek kursu (Protégé dahil birçok sistem, kavram isimleri).
- Kelimeleri birlikte çalıştırın ve her yeni kelimeyi büyük harfle yazın: MealCourse
- Adında bir alt çizgi veya kısa çizgi ya da başka bir sınırlayıcı kullanın: Yemek_Dersi , Meal_course , Yemek Kursu , Yemek Kursu. (Sınırlayıcı kullanırsanız, her yeni kelimenin büyük harfle yazılıp yazılmayacağına da karar vermeniz gerekir)

Bilgi temsil sistemi adlarda boşluklara izin veriyorsa, bunları kullanmak en çok birçok ontoloji geliştiricisi için sezgisel çözüm. Bununla birlikte, diğerlerini düşünmek önemlidir. sisteminizin etkileşime girebileceği sistemler. Bu sistemler boşluk kullanmıyorsa veya sizin sunum ortamı boşlukları iyi ele almıyor, başka bir yöntem kullanmak faydalı olabilir.

6.2 Tekil veya çoğul

Sınıf adı, bir nesne koleksiyonunu temsil eder. Örneğin, bir Wine sınıfı aslında tüm şarapları temsil eder. Bu nedenle, bazı tasarımcıların sınıfı çağırması daha doğal olabilir. Şarap yerine şaraplar. Hiçbir alternatif diğerinden daha iyi veya daha kötü değildir (tekil olmasına rağmen sınıf isimleri için pratikte daha sık kullanılır). Ancak, seçim ne olursa olsun, tüm ontoloji boyunca tutarlı. Hatta bazı sistemler, kullanıcılarının şunları beyan etmesini gerektirir: kavram isimleri için tekil veya çoğul kullanıp kullanmayacaklarını ve bu seçimden sapmalarına izin vermeyin.

Her zaman aynı formu kullanmak, bir tasarımcının böyle modelleme yapmasını da engeller. Şaraplar sınıfı oluştururken ve ardından alt sınıfı olarak bir Şarap sınıfı oluştururken yapılan hatalar (bkz. 4.1).

6.3 Önek ve sonek kuralları

Bazı bilgi tabanı metodolojileri, adlarda önek ve sonek kurallarının kullanılmasını önerir. sınıfları ve yuvaları ayırt etmek için. İki yaygın uygulama bir has- veya bir sonek eklemektir –Of yuva adlarına. Böylece, slotlarımız has-maker ve has-winery haline gelir.

has- kongre. Slotlar yapıcı ve şarap imalathanesi haline gelir. ortak düşünce. Bu yaklaşım, bir terime bakan herkesin, terim bir sınıf veya yuvadır. Bununla birlikte, terim isimleri biraz daha uzar.

6.4 Adlandırma ile ilgili diğer hususlar

Adlandırma kurallarını tanımlarken göz önünde bulundurmanız gereken birkaç nokta daha:

- Kavram adlarına “sınıf”, “özellik”, “yuva” vb. Gibi dizeler eklemeyin. Örneğin, kavramın bir sınıf mı yoksa bir yuva mı olduğu bağlamdan her zaman açıktır. İçinde Ayrıca sınıflar ve yuvalar için farklı adlandırma kuralları kullanıyorsunuz (örneğin, büyük harf kullanımı ve büyük harf kullanılmaması), adın kendisi kavramın ne olduğunun göstergesi olacaktır.
- Kavram adlarında kısaltmalardan kaçınmak genellikle iyi bir fikirdir (yani, Cabernet Sauvignon yerine Cab)
- Bir sınıfın doğrudan alt sınıflarının adlarının tümü adı içermeli veya içermemelidir süper sınıfın. Örneğin, Wine sınıfının iki alt sınıfını oluşturuyorsak, kırmızı ve beyaz şarapları temsil eder, iki alt sınıf adı Kırmızı Şarap olmalıdır ve Beyaz Şarap veya Kırmızı ve Beyaz, ancak Kırmızı Şarap ve Beyaz değil.

7 Diğer Kaynaklar

Örneklerimiz için bir ontoloji geliştirme ortamı olarak Protégé-2000 kullandık.

Duineveld ve meslektaşları (Duineveld ve diğerleri, 2000) bir dizi başka ontoloji geliştirme ortamları.

Ontoloji gelişiminin temellerine değinmeye çalıştık ve tartışmadık

ontoloji gelişimi için ileri konuların veya alternatif metodolojilerin çoğu.

Gómez-Pérez (Gómez-Pérez 1998) ve Uschold (Uschold ve Gruninger 1996) mevcut

alternatif ontoloji geliştirme metodolojileri. Ontolingua öğreticisi (Farquhar 1997)

bilgi modellemesinin bazı biçimsel yönlerini tartışır.

Şu anda, araştırmacılar yalnızca ontoloji gelişimini değil, aynı zamanda ontoloji analizini de vurgulamaktadır.

Daha fazla ontoloji üretilip yeniden kullanıldıkça, analiz etmek için daha fazla araç mevcut olacaktır.

ontolojiler. Örneğin, Chimaera (McGuinness ve diğerleri 2000) ,

ontolojileri analiz etmek. Chimaera'nın gerçekleştirdiği analiz, hem mantıksal hem de

bir ontolojinin doğruluğu ve yaygın ontoloji-tasarım hatalarının teşhisi. Bir ontoloji

tasarımcı, Chimaera teşhisini gelişen ontoloji üzerinde çalıştırmak isteyebilir.

ortak ontoloji modelleme uygulamalarına uygunluk.

8 Sonuç

Bu kılavuzda, bildirimsel için bir ontoloji geliştirme metodolojisi tanımladık.

çerçeve tabanlı sistemler. Ontoloji geliştirme sürecindeki adımları listeledik ve ele aldık.

sınıf hiyerarşilerini ve sınıfların ve örneklerin özelliklerini tanımlamanın karmaşık sorunları.

Ancak tüm kuralları ve önerileri uyguladıktan sonra yapılması gereken en önemli şeylerden biri

Şunu unutmayın: *herhangi bir alan için tek bir doğru ontoloji yoktur* . Ontoloji

tasarım yaratıcı bir süreçtir ve farklı insanlar tarafından tasarlanan hiçbir ontoloji,

aynı. Ontolojinin potansiyel uygulamaları ve tasarımcının anlayışı ve görüşü

alan adı kuşkusuz ontoloji tasarım seçimlerini etkileyecektir. "Kanıt,

puding ”— ontolojimizin kalitesini yalnızca

biz tasarladık.

Teşekkür

Protégé-2000 (<http://protege.stanford.edu>) Mark Musen'in Stanford'daki grubu tarafından geliştirilmiştir.

Tıp Bilişimi. Bazı rakamları Protégé'ye OntoViz eklentisiyle oluşturduk.

2000. Şarap ontolojisinin ilk versiyonunu Ontolingua ontolojisinden ithal ettik.

kütüphane (<http://www.ksl.stanford.edu/software/ontolingua/>) bir sürüm kullandı

Brachman ve arkadaşları (Brachman ve ark. 1991) tarafından yayınlanmış ve

KLASİK bilgi temsil sistemi. Daha sonra ontolojiyi şimdiye kadar değiştirdik

bildirimsel çerçeve tabanlı ontolojiler için kavramsal modelleme ilkeleri. Ray Fergerson ve

Mor Peleg'in önceki taslaklar hakkındaki kapsamlı yorumları bu makaleyi büyük ölçüde geliştirdi.

Referanslar

Booch, G., Rumbaugh, J. ve Jacobson, I. (1997). *Birleşik Modelleme Dili kullanıcı rehberi* : Addison-Wesley.

Brachman, RJ, McGuinness, DL, Patel-Schneider, PF, Resnick, LA ve Borgida, A.

(1991). CLASSIC ile yaşamak: KL-ONE benzeri bir dil ne zaman ve nasıl kullanılır. *İlkeleri*

Anlamsal Ağlar . JF Sowa, editör, Morgan Kaufmann : 401-456.

Brickley, D. ve Guha, RV (1999). Kaynak Açıklama Çerçevesi (RDF) Şeması

Şartname. Önerilen Öneri, World Wide Web Konsorsiyumu :

<http://www.w3.org/TR/PR-rdf-schema>.

Chimaera (2000). Chimaera Ontology Environment. www.ksl.stanford.edu/software/chimaera

Duineveld, AJ, Stoter, R., Weiden, MR, Kenepa, B. ve Benjamins, VR (2000).

WonderTools? Ontolojik mühendislik araçlarının karşılaştırmalı bir çalışması. *Uluslararası Dergi of Human-Computer Studies* **52** (6): 1111-1133.

- Farquhar, A. (1997). Ontolingua öğreticisi. <http://ksl-web.stanford.edu/people/axf/tutorial.pdf>
- Gómez-Pérez, A. (1998). Bilgi paylaşımı ve yeniden kullanım. *Uygulamalı Uzman Sistemler El Kitabı*. Liebowitz, editör, CRC Press.
- Gruber, TR (1993). Taşınabilir Ontoloji Spesifikasyonuna Çeviri Yaklaşımı. *Bilgi Edinme* **5** : 199-220.
- Gruninger, M. ve Fox, MS (1995). Tasarım ve Değerlendirme Metodolojisi Ontolojiler. İçinde: *Bilgide Temel Ontolojik Sorunlar Çalıştayı Bildirileri Paylaşım, IJCAI-95*, Montreal.
- Hendler, J. ve McGuinness, DL (2000). DARPA Aracı Biçimlendirme Dili. *IEEE Akıllı Sistemler* **16** (6): 67-73.
- Humphreys, BL ve Lindberg, DAB (1993). UMLS projesi: kavramsal yapmak Kullanıcılar ve ihtiyaç duydukları bilgiler arasındaki bağlantı. *Tıp Kütüphanesi Bülteni Dernek* **81** (2): 170.
- McGuinness, DL, Abrahams, MK, Resnick, LA, Patel-Schneider, PF, Thomason, RH, Cavalli-Sforza, V. ve Conati, C. (1994). Klasik Bilgi Temsil Sistemi Eğitimi. <http://www.bell-labs.com/project/classic/papers/ClassTut/ClassTut.html>
- McGuinness, DL, Fikes, R., Rice, J. ve Wilder, S. (2000). Birleşme ve Büyük Ontolojileri Test Etmek. *Bilgi Temsili ve Akıl Yürütme İlkeleri: Yedinci Uluslararası Konferans Bildirileri (KR2000)*. AG Cohn, F. Giunchiglia ve B. Selman, editörler. San Francisco, CA, Morgan Kaufmann Publishers.

- McGuinness, DL ve Wright, J. (1998). Yapılandırma için Kavramsal Modelleme: A Tanım Mantık Tabanlı Yaklaşım. *Mühendislik Tasarımı için Yapay Zeka, Analiz, ve Üretim - Yapılandırma ile ilgili özel sayı*.
- Musen, MA (1992). Bilgi paylaşımı ve yeniden kullanımın boyutları. *Bilgisayarlar ve Biyomedikal Araştırma* **25** : 435-467.
- Ontolingua (1997). Ontolingua Sistemi Referans Kılavuzu. <http://www-ksl-svc.stanford.edu:5915/doc/frame-editor/index.html>
- Price, C. ve Spackman, K. (2000). SNOMED klinik terimler. *BJHC & IM-British Journal of Sağlık Bilişim ve Bilgi Yönetimi* **17** (3): 27-31.
- Protege (2000). Protege Projesi. <http://protege.stanford.edu>
- Rosch, E. (1978). Sınıflandırma İlkeleri. *Biliş ve Sınıflandırma*. RE ve B. B. Lloyd, editörler. Hillside, NJ, Lawrence Erlbaum Yayıncılar : 27-48.
- Rothenfluh, TR, Gennari, JH, Eriksson, H., Puerta, AR, Tu, SW ve Musen, MA (1996). Yeniden kullanılabilir ontolojiler, bilgi edinme araçları ve performans sistemleri: Sisifos-2'ye PROTÉGÉ-II çözümleri. *Uluslararası İnsan-Bilgisayar Araştırmaları Dergisi* **44** : 303-332.
- Rumbaugh, J., Blaha, M., Premerlani, W., Eddy, F. ve Lorensen, W. (1991). *Nesne odaklı modelleme ve tasarım*. Englewood Kayalıkları, New Jersey: Prentice Hall.
- Uschold, M. ve Gruninger, M. (1996). Ontolojiler: İlkeler, Yöntemler ve Uygulamalar. *Bilgi Mühendisliği İncelemesi* **11** (2).

