

Н. В. Бессарабов (Краснодар, КубГУ). **Классы, фреймы и концепты.**

Под концептом будем понимать некоторое расширение понятий «класс» (объектное программирование) и сущность (ER-диаграммы), позволяющее получить программно реализуемое представление достаточно представительного подмножества фреймов [1], существенно более широкого, чем допускают обычные классы и сущности. Это позволит расширить применение программных инструментальных средств на такие науки, как биология, социология, лингвистика и т. д. Разрабатываемый подход основан на более широком, чем обычно представлении концептов, их структур и взаимодействия [2]. Есть основания полагать, что он может быть использован для формализации и обобщения теории решения изобретательских задач (ТРИЗ) [3] и связанных с ней направлений.

В [2] предложено расширить спецификацию концепта введением кроме обязательных атрибутов еще ресурсов и состояний. Обратим внимание на то, что в повседневной практике проектирования объектных систем обязательно учитывают некоторый набор ограничений на сочетания атрибутов. Например, трудно представить класс «сотрудник» всего с тремя атрибутами «ИНН», «отчество» и «зарплата». Введем ограничения на сочетания атрибутов. Добавим в описание концепта еще необязательные атрибуты. Будем рассматривать атрибуты всех четырех сортов (обязательные, необязательные, состояния и ресурсы), как функции одной переменной, действующие из множества идентификаторов экземпляров в домен атрибута. Аналогии методов, определяющие поведение сущностей описываемых концептами естественно являются также функциями, но в общем случае более чем одной переменной, причем переменными могут быть атрибуты. Будем говорить об атрибутах и методах как обобщенных свойствах. Концепты–связи имеют ту же структуру, что и концепты–сущности. Остается допустить существование набора первичных концептов (категорий), не имеющих никаких обобщенных свойств, и для задания связей между ними определить некоторую систему рассуждений в заданном концептуальном пространстве.

В отличие от парадигмы объектного программирования, где атрибуты либо независимы, т. е. в некотором смысле ортогональны, либо вычислимы через другие атрибуты, в нашем случае атрибуты могут быть зависимыми, связанными, перекрывающимися.

Заметим, что даже такого обобщения не достаточно для работы с концептами, изучаемыми в биологических и общественных науках. В зависимости от предметной области может оказаться необходимым добавить: контекстные связи; историю концепта как развивающегося понятия в некоторой системе концептов; описания развивающихся объектов; подтекст.

Наибольшие трудности вызывает реализация подтекста, который может быть описан как скрытый концепт, в некоторой, известной посвященным, системе концептов. Описание контекстных связей и истории концепта может потребовать большого объема знаний и больших затрат для реализации. Описания развивающихся объектов, с небольшим числом стадий, не представляет труда.

В соответствии с [2] будем говорить, что система имеет большую онтологию, если ее модель не может быть описана небольшим числом концептов. Заметим, что классические математические и (физические системы имеют малые онтологии. Это возможно из-за высокой сворачиваемости знаний в этих областях. Системы с большими онтологиями таким свойством не обладают.

Самые слабые модели систем с большими онтологиями могут принимать характер таксономии. Более сильные модели обеспечивают сворачивание знаний для частей предметной области. Некоторые общие закономерности могут быть получены за счет повышения уровня обобщения и использования нечетких и модальных рассуждений.

Сделанные выводы подтверждаются немногочисленными известными примерами

построения моделей систем с большими онтологиями. Так, предпринятая в [4] попытка построить групповую модель системы категорий философии K_i , используя в качестве групповой единицы все множество категорий $\sum K_i$, а $\sum K_i - K_0$ в качестве элемента дополнительного к категории K_0 , неудачна, из-за не сводимости системы с большой онтологией к единственной системе с малой онтологией.

Предварительный анализ теории решения изобретательских задач [3] с точки зрения онтологического представления с концептами предлагаемого вида создает впечатление глубокой соразмерности, адекватности подхода и, видимо, типичности общей структуры теории для систем с большими онтологиями. Есть основания полагать, что к универсальным особенностям такой теории относятся: наличие, может быть нескольких, частных онтологий; в ТРИЗ это, например, справочники физических эффектов; наличие необходимого набора основных принципов, определяющих направления деятельности (в ТРИЗ это системность объектов, существование объективных законов развития технических систем, поиск и преодоление противоречия, существование идеального конечного результата, необходимость критического переосмысления постановки задачи, наличие «человеческой» и экономической компонент принятия решений); наличие классификатора решаемых задач с указанием их уровня, трудности и реальности решения; это своего рода ограничитель ожиданий; направленность на выбор создаваемых структур (как в ТРИЗ), либо на описание динамики, либо на изменение архитектуры и функций систем и их компонент.

Автор благодарен своей дипломнице О. Терещенко, реализовавшей на языке Java инструментальное средство, позволившее создавать и визуализировать онтологии, включающие концепты–категории и связи между ними, концепты–сущности и концепты–связи с четырьмя типами атрибутов (обязательные, необязательные, состояния и ресурсы) и методами.

СПИСОК ЛИТЕРАТУРЫ

1. Минский М. Фреймы для представления знаний. М.: Энергия, 1979.
2. Бессарабов И. В., Бессарабова К. Н. Онтологии и динамические системы. — Экологический вестник научных центров Черноморского экономического сотрудничества, 2005, спецвыпуск, с. 5–10.
3. Альтшуллер Р. С. Найти идею. Введение в теорию решения изобретательских задач. Петрозаводск: Скандинавия, 2003.
4. Федосин С. Р. Основы синкретики. Философия носителей. М.: Едиториал УРСС, 2003.