

УДК 004.94

ОБЪЕКТНО-АССОЦИАТИВНЫЙ ПОДХОД К ПРЕДСТАВЛЕНИЮ ИНФОРМАЦИИ

*канд. техн. наук, доц. Д.О. ГЛУХОВ
(Полоцкий государственный университет)*

Системы знаний, лежащие в основе современных информационных систем, объединяет одно фундаментальное свойство, ставшее их самым большим ограничением – первичность абстрактного описания. Объектно-ассоциативный подход базируется на идее, что мы располагаем бесконечным разнообразием объектов в системе и нет никаких препятствий для добавления в систему нового объекта с новым набором свойств и новыми видами ассоциативных связей с другими объектами системы. Объектно-ассоциативный подход не ограничивает разнообразие объектов и связей между ними. Он позволяет не рассматривать эти ограничения не только на стадии проектирования и разработки программного обеспечения, но и на стадии эксплуатации. Это ключевое отличие объектно-ассоциативного подхода от всех концепций и парадигм информационного моделирования, практикуемых на сегодняшний день.

История информационных систем, эволюция подходов к их построению, калейдоскоп концепций, архитектур, наглядно иллюстрирует нам парадоксальность человеческой мысли. Парадоксальность в том, что, имея выбор, человечество выбирает одну из альтернатив, развивает ее, при этом остальные альтернативы пренебрегают, они игнорируются и даже стремительно уничтожаются. Уничтожаются с той же стремительностью, с какой выбранная альтернатива овладевает умами и идет к тотальному доминированию.

Стройные системы знаний, лежащие в основе современных информационных систем, объединяет одно фундаментальное свойство, ставшее их самым большим ограничением – первичность абстрактного описания. Во главу угла современных информационных технологий поставлена первичность абстракции. Вся инженерия программного обеспечения, все техники построения информационных моделей исходят из первичности абстракции и вторичности вносимых в эти неизменяемые и ограниченные абстракции данных. Данное обстоятельство приводит к следующим драматическим последствиям:

- 1) конструировать информационную модель предметной среды оказывается возможным только на стадии разработки программного обеспечения (обычно в сжатые сроки и, следовательно, допуская ошибки, неполноту по свойствам или связям);
- 2) оказывается невозможным фиксация связей и свойств объектов, не вошедших в абстрактное описание на стадии разработки;
- 3) разнообразие способов представления информации ограничено абстрактным уровнем описания и возможностями применяемого инструментария.

Возведя абстракцию в универсальный способ описания (точнее, представление виртуальной) действительности, идеологи современных информационных технологий: Эд Кодд, Дейт, Ульман, Бьярн Страустрап, Эд Йордон, Гради Буч, С. Шлеер, С. Меллор, Б. Мейер, Майкл Стоунбрейкер и др. [3 – 11], фактически предложили свой философский взгляд на онтологию. Однако еще Аристотель, рассуждая над не-общностью Сущего и Единого, Иммануил Кант в своей «Трансцендентальной эстетике», говоря об априорных формах созерцания, замечают тот факт, что ряд категорий, понятий для нас не является следствием, результатом обобщения, классификации, то есть, являясь важными с точки зрения информационной модели, такие «априорные формы созерцания» не являются абстракциями [1, 2].

С точки зрения прагматичного программиста, обученного и привыкшего к действующим сегодня парадигмам, попытки найти философскую основу принятым подходам покажутся ненужной тратой времени.

Покажем, что объектно-ассоциативный подход (ОАП) является не менее прагматичным и удобным с точки зрения информационного моделирования. Именно ОАП дает возможность получить простое решение сложных проектных задач. Объектно-ассоциативный подход базируется на идее, что мы располагаем бесконечным разнообразием объектов в системе и нет никаких препятствий для добавления в систему нового объекта с новым набором свойств и новыми видами ассоциативных связей с другими объектами системы.

Данный подход не ограничивает разнообразие объектов и связей между ними. Он позволяет не рассматривать эти ограничения не только на стадии проектирования и разработки программного обеспечения, но и на стадии эксплуатации. Это ключевое отличие ОАП от всех концепций и парадигм информационного моделирования, практикуемых на сегодняшний день.

Парадоксальные информационные миры

Прежде чем рассмотреть одну из возможных формализаций предлагаемой идеи, остановимся на еще одном размышлении об информационных технологиях сегодняшнего дня. Все мировое информационное пространство в настоящее время существует благодаря выбору, сделанному на заре информаци-

ных технологий в пользу операции копирования информации. Все современные технологии построены на многократном, многоуровневом копировании информации на ее пути от источника к потребителю. Копирование начинается на аппаратном уровне, уровне контроллеров, далее следует в различных буферах, кешах, прикладных программных структурах, памяти передающих устройств, экрана, и т.д. Многократное параллельное копирование требует колоссального ресурса. Надежность такой системы благодаря огромному количеству копий должна, казалось бы, быть высочайшей. Но, по странному стечению обстоятельств, большинство производимых копий никак не участвует в процессе обеспечения надежного хранения и передачи информации. Кроме этого, появляется множество аспектов уязвимости с точки зрения несанкционированного доступа к информации (к одной из ее копий). Современные информационные технологии усиленно решают проблему защиты информации, блокировки доступа к копиям, расходуются огромные средства, но сама идея технологий, основанных на копировании, препятствует решению задачи борьбы с несанкционированным доступом.

Постепенно информация, представленная в цифровой форме, становится продуктом номер один для потребителей. Мы готовы тратить на нее все больше и больше. Однако, по нашему мнению, пришло время пересмотреть концепцию копирования. При многократном копировании информации вопрос учета, стоимости информации теряет смысл. По сути, система сама провоцирует пиратское использование излишнего копируемого материала.

Представим на мгновение иной мир, технологии, изначально работавшие с информацией как с продуктом, копирование которого рассматривалось как кража. Мы можем вообразить себе мир перемещающихся информационных объектов, бесследно скользящих по информационной инфраструктуре. Представьте публикацию в сети статьи определенным тиражом или надпись на сайте «на сегодня экземпляры этой странички закончились». Ученые решали бы и несомненно решили бы основную проблему надежной передачи, поставки информации конечному потребителю. Эти размышления наглядно показывают, что путей эволюции в области информационных технологий много и человечество сделало первую попытку построить виртуальное пространство. Наша же задача – увидеть альтернативы, их преимущества, их особенности, раскрыть скрытый потенциал фантастического мира информации.

Одна из возможных формализаций объектно-ассоциативного подхода

Вернемся к нашей идее построения информационных моделей. Рассмотрим формальную систему, в первом приближении удовлетворяющую нашим требованиям.

Фактически в рамках объектно-ассоциативного подхода мы декларируем существования универсального множества объектов O . Причем под объектом мы будем понимать не только сущности, выявляемые при информационном анализе предметной среды, но и любые единичные свойства. Объекты-свойства – это несоставные «терминальные» объекты с точки зрения ассоциации.

Мы говорим: «объект обладает свойствами», – фактически отражая тот факт, что объект в нашем представлении есть совокупность объектов-свойств, обладающих тривиальными смыслами. Объекта как такового не существует, существует совокупность свойств, в своей целостности представляющая новый смысл и, соответственно, способная участвовать в качественно новых связях-ассоциациях.

Множеству объектов сопоставляется множество смыслов (интерпретаций) E :

O – множество объектов;

E – множество смыслов (интерпретаций).

Введем еще одно формальное определение – множество представлений (множество разметки) L . Это фактически формализации неких срезов, инструментов оперирования информацией (форм, схем, карт), через которые мы видим, воспринимаем информацию;

L – множество разметки.

Эта тройка, отметим еще раз, в первом приближении может быть использована нами для иллюстрации реализации предложенной идеи в программном обеспечении.

Ранее мы предлагали формализацию объектно-ассоциативной системы ОАС, построенной на одном фундаментальном отношении агрегации. Рассмотрим ее более подробно.

Универсальным способом представления значений будем рассматривать строковое представление. Для алфавита char множество допустимых строк – цепочек символов – будем обозначать как char^* . Такая нотация употребляется в теории формальных грамматик и, кроме этого, хорошо понятна программистам, работающим на языке C++.

Для определения доступа к элементам множеств будем использовать точечный оператор:

$o.a : a \in o$

Определение объекта. Каждый объект универсального множества объектов является сложным по своей структуре и для реализации сформулированных принципов объектно-ассоциативного подхода может быть представлен в следующей конфигурации:

$o = \langle \text{value}, \text{reference}, \text{interpretation}, \text{parent} \rangle$

Ведем соответствующие операторы отношений.

Уровни представления ассоциативной системы объектов

Будем рассматривать ассоциативную систему объектов на трех уровнях представления:

- 1) интерфейсное представление в виде совокупности подмножеств разметки;
- 2) объектное представление с отношением агрегации на множестве объектов;
- 3) смысловое представление или интерпретация в пространстве смыслов EXPLAIN.

Определение агрегации

Объекты могут связываться отношением агрегации.

Если объект o_1 входит в объект o_2 , то

$$o_1.parent = o_2$$

Или будем записывать

$$o_2 = parentOf(o_1),$$

$$o_1 \in childrenOf(o_2)$$

Определение интерпретации

Смысловое окрашивание объекта в рамках некоторой предметной области есть его интерпретация. Множество допустимых интерпретаций E . Одним и тем же смыслом могут обладать разные объекты. Объект не может быть двусмысленным. Интерпретация свойственна всем объектам и

$$o.interpretation \in char^*$$

Введем оператор интерпретации:

$$explain(o) = o.interpretation$$

Определение подмножества разметки

Объекты могут быть сгруппированы в подмножества разметки. Подмножеством разметки мы будем называть некоторое произвольное подмножество объектов, сформированное по некоторому смысловому признаку. Мы можем определять подмножества по своему усмотрению. Примером подмножества разметки будет являться набор объектов значений и ссылок, на наш взгляд, необходимый для определения абстракции.

В зависимости от признака выделения подмножества можно рассмотреть 3 типа подмножеств разметки:

- 1) некоторое произвольное подмножество L ;
- 2) подмножество объектов, объединенных тем, что они все являются агрегированными в один и тот же родительский объект $o - L(o)$;
- 3) подмножество объектов, ассоциированных с некоторой интерпретацией $L(e), e \in E$.

Подмножества разметки могут пересекаться, быть равными, не пересекаться, но их объединение всегда равно универсальному множеству объектов.

Иными словами, любой объект всегда принадлежит некоторому множеству разметки, но он также может принадлежать и нескольким различным подмножествам одновременно.

$$\bigcup_i L_i = O$$

Подмножества разметки выступают в качестве тех «страниц», через которые мы воспринимаем систему объектов. Они выступают интерфейсными составляющими системы.

Поскольку разметка системы объектов допускает формирование новых групп объектов на уже существующем множестве данных, то именно разметка может служить средством формулировки абстракций.

Определение подобия подмножеств разметки

Два объекта, не агрегирующие другие объекты, подобны, если

$$o_1 \sim o_2 : \begin{matrix} explain(o_1) = explain(o_2) & u \\ L(explain(o_1)) = L(explain(o_2)). \end{matrix} \quad (1)$$

Два подмножества разметки подобны в том и только в том случае, если

$$L_1 \sim L_2 : |L_1| = |L_2| \quad \text{И} \quad \forall o_i \in L_1 \quad \exists ! o_j \in L_2 \quad , o_i \sim o_j. \quad (2)$$

Два объекта, содержащие агрегированные объекты, подобны, если

$$o_1 \sim o_2 : \begin{matrix} explain(o_1) = explain(o_2) & u \\ L(explain(o_1)) = L(explain(o_2)) & u \\ L(o_1) \sim L(o_2). \end{matrix} \quad (3)$$

Данное определение рекурсивно.

Подобие множеств разметки является ограничительным отношением в силу своей симметричности. Однако на его основе мы введем менее ограничительное отношение – включение подобия.

Множество разметки L_1 включает подобие множества разметки L_2 в том и только том случае, если

$$L_1 \supseteq L_2 : \forall o_i \in L_2 \exists! o_j \in L_1, o_i \sim o_j. \tag{4}$$

Основным отличием данного определения от определения (2) является отсутствие ограничения на равенство мощностей множеств. То есть множество L_1 может содержать, кроме объектов, подобных объектам множества L_2 , объекты, не относящиеся к множеству L_2 .

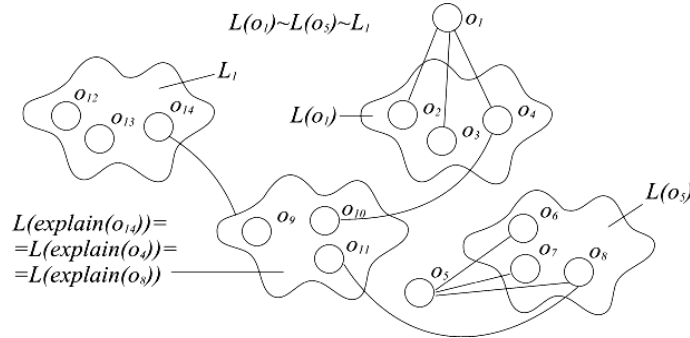


Иллюстрация отношения подобия на подмножествах разметки

Определение значения

Значением объекта является цепочка символов, в силу своей универсальности:

$o.value \in char^*$, или в операторной форме $valueOf(o) = \omega \in char^*$.

Определение ссылки

Объект o_1 ссылается на объект o_2 , если $o_1.reference = o_2$ и $o_1.value = o_2.value$, что выражается оператором

$$destinationOf(o_1) = o_2.$$

Определение клона

Клоном считается ссылка, но без ограничения по совпадению значения объектов.

$$o_1.reference = o_2$$

$$o_2 = clone(o_1)$$

Свойства объекта

Итак, объект обладает свойствами, которые выражаются агрегированными в нем значениями или ссылками. Мы будем говорить о некотором подмножестве свойств объекта, если некоторое подмножество агрегированных в нем значений и ссылок принадлежит одному подмножеству разметки. Если это подмножество разметки содержит только агрегированные объекты объекта o , то мы будем отражать это в виде $L(o)$. Если некоторое подмножество разметки содержит только агрегированные объекты объекта o , и причем все такие объекты мы будем записывать как $L_{\Sigma}(o)$.

$$\bigcup_i L_i(o) = L_{\Sigma}(o), \tag{5}$$

хотя для дальнейших рассуждений нам совсем необязательно рассматривать исключительно полные наборы свойств.

Классификация

Подмножества разметки с определенным на них отношением подобия и включения подобия являются простейшим способом формулирования классов на множестве объектов.

Если объект имеет свойства, то будем иметь в виду, что мы можем сформировать подмножество разметки, такое, что подмножество разметки свойств объекта будет обладать подобием ему или включать его подобие. Иными словами, мы выделим тип объекта в виде отдельного подмножества. Отношение подобия установит сходство объектов по наличию у них сходного состава свойств и выступит инструментом классификации.

Тогда классом C объектов системы, заданным некоторым множеством разметки L , будет множество объектов, обладающих свойством

$$C(L) = \{o_i\} | \forall o_i \in C(L) \ L(o_i) \supseteq L. \tag{6}$$

Говорить о том, что мы, обладая инструментом классификации, предложим язык манипулирования данными в такой ассоциативной среде, еще рано. И тем более рано говорить об эффективных реализациях операций над динамическими классами. Однако необходимо подчеркнуть, что классы, которые появились как вторичный продукт на множестве объектов, могут естественным образом эксплуатировать реляционную алгебру и реляционное исчисление как готовый, хорошо изученный математический аппарат манипулирования данными. Чтобы ощутить все удобства эксплуатации объектно-ассоциативной модели данных можно рассмотреть наиболее характерные области применения этого подхода и предлагаемую нами реализацию ОАС NormPro.

Заключение. Полученное описание достаточно универсально. Все отношения сведены к одному базовому отношению включенности. В силу своей универсальности оно отражает ту концептуальную особенность, что все объекты равноправны, равновесомы, в некотором смысле обезличены. Нет никакой абстракции. Есть только однородная ассоциативная сеть. Абстракция же является продуктом вторичным. Такая модель данных идеально подходит для создания базы данных в условиях начальной неопределенности относительно многообразия объектов в системе, полных спецификаций объектов и возможных связей между объектами. В связи с тем, что именно такая неопределенность стала ключевым мотивом разработки объектно-ассоциативной базы данных, эта модель позволяет расширять и обогащать систему во время эксплуатации без ущерба для хранимых в ней данных.

Основным направлением дальнейшей работы нам видится разработка языка управления данными в объектно-ассоциативной среде и создание эффективных реализаций теоретико-множественных и специальных операций в ассоциативной объектной системе.

ЛИТЕРАТУРА

1. Кант, И. Критика чистого разума / И. Кант. – М.: Мысль, 1994. – (Серия «Философское наследие»).
2. Аристотель Метафизика / Аристотель // Сочинения. – М.: Мысль, 1975. – Т. 1.
3. Williams, S. The Associative Model of Data / S. Williams. – Second edition. – Published in Great Britain: Lazy Software Ltd, 2002. – 272 p.
4. Booch, G. Object-oriented analysis and design with applications / G. Booch. – Second edition. – The Benjamin/Cummings Publishing Company, Inc. 1994. – 589 p.; Объектно-ориентированный анализ и проектирование: с примерами приложений на C++ / рус. пер. Г. Буч. – М.: Изд-во «Бином», «Невский диалект», 1998. – 560 с.
5. Codd, E.F. Relation Model of Data for Large Shared Data Banks / E.F. Codd // Comm. ACM. – 1970. – V. 13, № 6. – P. 377 – 383.; Реляционная модель данных для больших совместно используемых банков данных / пер. Е.Ф. Кодд // СУБД. – 1995. – № 1. – С. 145 – 160.
6. Страуструп, Б. Язык программирования C++ / Б. Страуструп; пер. с англ. – 3-е изд. – СПб.; М.: «Невский диалект» – «Изд-во БИНОМ», 1999. – 991 с.
7. Шлеер, С. Объектно-ориентированный анализ: моделирование мира в состояниях / С. Шлеер, С. Меллор; пер. с англ. – Киев: Диалектика, 1993. – 204 с.
8. A critical view of inheritance and reusability in object-oriented programming: Acad. Diss. Univ. of Jyraskyla / Antero Taivalsaar. – Univ. of Jyraskyla, 1993. – 276 p.
9. Дейт, К. Введение в базы данных / К. Дейт. – 6-е изд. – Киев: Диалектика, 1998.
10. Пржиялковский, В.В. Абстракции в проектировании баз данных / В.В. Пржиялковский // СУБД. – 1998. – № 1 – 2. – С. 90 – 97.
11. Стоунбрейкер, М. Объектно-реляционные системы баз данных / М. Стоунбрейкер // Открытые Системы. – 1994. – № 4.

OBJECT-ASSOCIATIVE APPROACH TO REPRESENTATION OF INFORMATION

D. GLUKHOV

Knowledge that underlie modern information systems, are united by one fundamental property that has become their biggest limitation - the primacy of the abstract description. Object-associative approach is based on the idea that we have an endless variety of objects in the system and there are no impediments to add a new object with a new set of properties and new kinds of associative links with other objects of the system. CAP does not restrict the variety of objects and relations between them. It allows you to not consider these restrictions not only on the stage of designing and developing software, but also during the using. This is a key difference between a PDA from all the concepts and paradigms of information modeling, practiced today.