Принципы разработки в среде Домино

Разработка приложения в среде Домино отличается от разработки в универсальных системах тем, что приложение Домино разрабатывается в рамках проблемно-ориентированной модели с применением метаданных.
При обычном программировании необходимо описать целый ряд сущностей и способы их регистрации и взаимодействия. При проектировании на платформе Домино разработчик использует объекты (сущности), заранее описанные в платформе. Это товары, документы, каталоги, партнёры, регистры и т.д. Каждый из этих объектов имеет внутренние классификацию и типизацию.

Выделение объектов, типов и классов определялось следующими соображениями:

- понятны назначение и роль объекта в модели данных
- типизация и классификация должны обеспечивать унификацию разработки прикладных решений
- понятны способы применения объектов, классов и типов

Полученные наборы объектов, классов и типов описаны в платформе и все вместе составляют метаданные, которые в значительной степени определяют структуру проектирования приложения. Каждое приложение Домино основывается и опирается на структуру метаданных. При этом, непосредственно в приложении можно добавить свои объекты, типы и классы.

Описание объекта включает такую информацию, как используемая таблица базы данных, поля таблицы, типовые интерфейсные элементы, наборы прав, которые будут использоваться в системе разграничения прав доступа. На основе имеющихся объектов разработчик может создать свои собственные объекты. Эти новые объекты с момента создания будут наследовать свойства от объекта-родителя. При необходимости разработчик может указать особенности, которой должна обладать новая сущность в отличие от родителя.

В платформе уже заложена мощная функциональность, которая позволяет, не добавляя никакой новой (по сравнению с заложенной в платформе) функциональности, быстро и удобно поддерживать объекты на уровне прикладного решения.

Имеются следующие объекты:

- Каталоги

- Товары
- Партнёры (подразделения)

Это справочники, имеющие общие свойства: внутренняя идентификация объекта в БД, необходимость поддержки иерархии и группировки элементов, участие в хозяйственных операциях предприятия. Перечисленные свойства поддерживаются на уровне платформы.

Справочники - это товары, поставщики, товарные группы и категории, подразделения, банки и т.п.

В справочниках также можно хранить сведения о значениях различных величин, которые сами по себе не имеют объектной семантики. Такими значениями могут быть, например, курсы валют на определённую дату.

- Документы

Документы - это счета, накладные, заказы и т.п. Посредством документов фиксируются различные события, происходящие в хозяйственной жизни организации. Важными свойствами документа являются его привязка ко времени и месту создания/применения/хранения (подразделение). Документ может иметь состав - строки.

Обычно в бизнес-приложениях разделяют два состояния документа: черновик и подтверждённый (акцептованный) документ. Если документ не подтверждён, то считается, что документ ещё не готов, находится в процессе ввода, проверки, согласования и т.п. Только после подтверждения документ учитывается в различных системных регистрах.

- Регистры

Регистр "отвечает" за учёт движений ресурсов (финансов, товаров, материалов и т.д.). В регистрах хранится информация о поступлении и расходе тех или иных ресурсов. Заложенная в платформу функциональность этих регистров позволяет получить остатки ресурсов на определённый момент времени, рассчитать итоги и т.д.

- Проводки и сальдо

Система двойной записи бухгалтерского учёта представляет собой отдельную модель учёта со своей спецификой, поэтому для хранения данных выделены отдельные классы объектов. Причём механизмы, связанные с данными объектами, никак не навязывают разработчику собственно принципов ведения бухгалтерского учёта.

Версия #6

Демонов Сергей создал Tue, Jan 16, 2024 12:37 PM

Демонов Сергей обновил Fri, Jan 19, 2024 3:04 PM