

МНОГОУРОВНЕВОЕ ОБЪЕКТНОЕ ОПИСАНИЕ ИНТЕГРИРОВАННОЙ СРЕДЫ АВТОМАТИЗИРОВАННОГО ПРОЕКТИРОВАНИЯ

Кроль Т.Я., канд. техн. наук, Чистяков П.Н., аспирант, Капитонихин А.С., асп.

В данной статье рассматриваются методы интеграции гетерогенных САПР в единое информационное пространство проектирования сложных электротехнических объектов. Основой интеграционного решения является многоуровневое метаописание различных аспектов проектирования. Приводятся математические модели метаописания и описывается архитектура информационной системы, построенной на базе предложенной концепции.

Долгие годы при разработке САПР решались задачи автоматизированного выполнения наиболее трудоемких проектных процедур. Успехи в этой области не вызывают сомнения, однако вопрос комплексной автоматизации процесса и среды проектирования в настоящее время не решен. В связи с необходимостью применения разнородных проектирующих и информационно-управляющих систем при разработке и сопровождении сложных электротехнических объектов, таких, как тепловые и атомные электрические станции, актуальной является задача создания и поддержки единого информационного пространства проектирования.

В данном пространстве присутствует информация о погруженных в него САПР, документообороте процесса проектирования, графике выполнения проектных работ, оборудовании проекта, нормативно-справочная и номенклатурная информация, а также прочие специфические сведения, доступные любой погруженной САПР в соответствии с ее полномочиями. Именно наличие целостной информации в едином хранилище позволяет обеспечить поддержку всего процесса проектирования. Структура и состав хранимой информации являются существенно переменными из-за следующих основных причин:

- 1 необходимости применения новых САПР и отказа от использования существующих;
- 1 модификации используемых САПР;
- 1 динамики рынка покупных изделий и комплектующих;
- 1 быстро изменяющегося процесса проектирования и структуры проектных организаций.

При этом изменения в структуре должны быть сделаны в минимальные сроки и с минимальными затратами и без остановки процесса проектирования. Таким образом, единое хранилище должно быть реализовано с учетом требований инвариантности по отношению к структуре и виду хранимой информации, гибкости и настраиваемости.

В данной статье предлагается технология создания и поддержки хранилища, отвечающего перечисленным требованиям. В ее основу положено многоуровневое объектное описание интегрированной среды автоматизированного проектирования.

Многоуровневое метаописание создается на базе модели объектно-ориентированной декомпозиции информационного пространства. Согласно этой модели, информационное хранилище

представляет собой систему взаимосвязанных объектов, называемых *информационными объектами*, отражающих информационные нужды погруженных САПР.

Концептуально, метаописание представляется иерархической структурой (рис. 1), при переходе с каждого нижележащего уровня которой к вышележащему повышается уровень абстракции модели информационного хранилища. Таким образом, оперируя на первом уровне структурными элементами единого хранилища в терминах СУБД, на третьем и четвертом уровнях оперирование происходит в терминах предметной области. Такое абстрагирование от конкретной реализации алгоритмов доступа к нужным данным позволяет перейти к оперированию терминами предметной области и решить проблему настраиваемости и гибкости путем применения унифицированных алгоритмов доступа к данным нижележащего уровня.

Самый нижний уровень иерархии представлен *уровнем данных*, который находится под управлением СУБД. Анализ рынка СУБД показал, что наиболее востребованными в настоящее время являются реляционные СУБД, в связи с чем единое хранилище должно быть представлено в виде реляционной базы данных. Таким образом, формально первый уровень R иерархии метаописания можно представить в виде схемы базы данных [1]. Пусть U – множество атрибутов единого хранилища, каждый из которых соотнесен со своим доменом значений. Тогда R над U представляет собой совокупность схем реляционных отношений $\{R_1, R_2, \dots, R_p\}$.

$$R_i = (K_i, S_i), 1 \leq i \leq p,$$

p – количество атрибутов в хранилище,

K_i – множество выделенных ключей,

S_i – множество атрибутов единого хранилища, таких что:

$$\bigcup_{i=1}^p S_i = U, S_i \neq S_j, \text{ при } i \neq j,$$

Второй уровень иерархии – *объектный*. Любой объект данного уровня имеет свойства, отражающие информационные нужды проектировщиков. Благодаря алгоритмам комбинирования данных и наличию схемы хранилища на нижнем уровне в свойствах информационного объекта могут отображаться данные из разных отношений хранилища. Каждое свойство объекта нотируется его смысловой нагрузкой в предметной области.

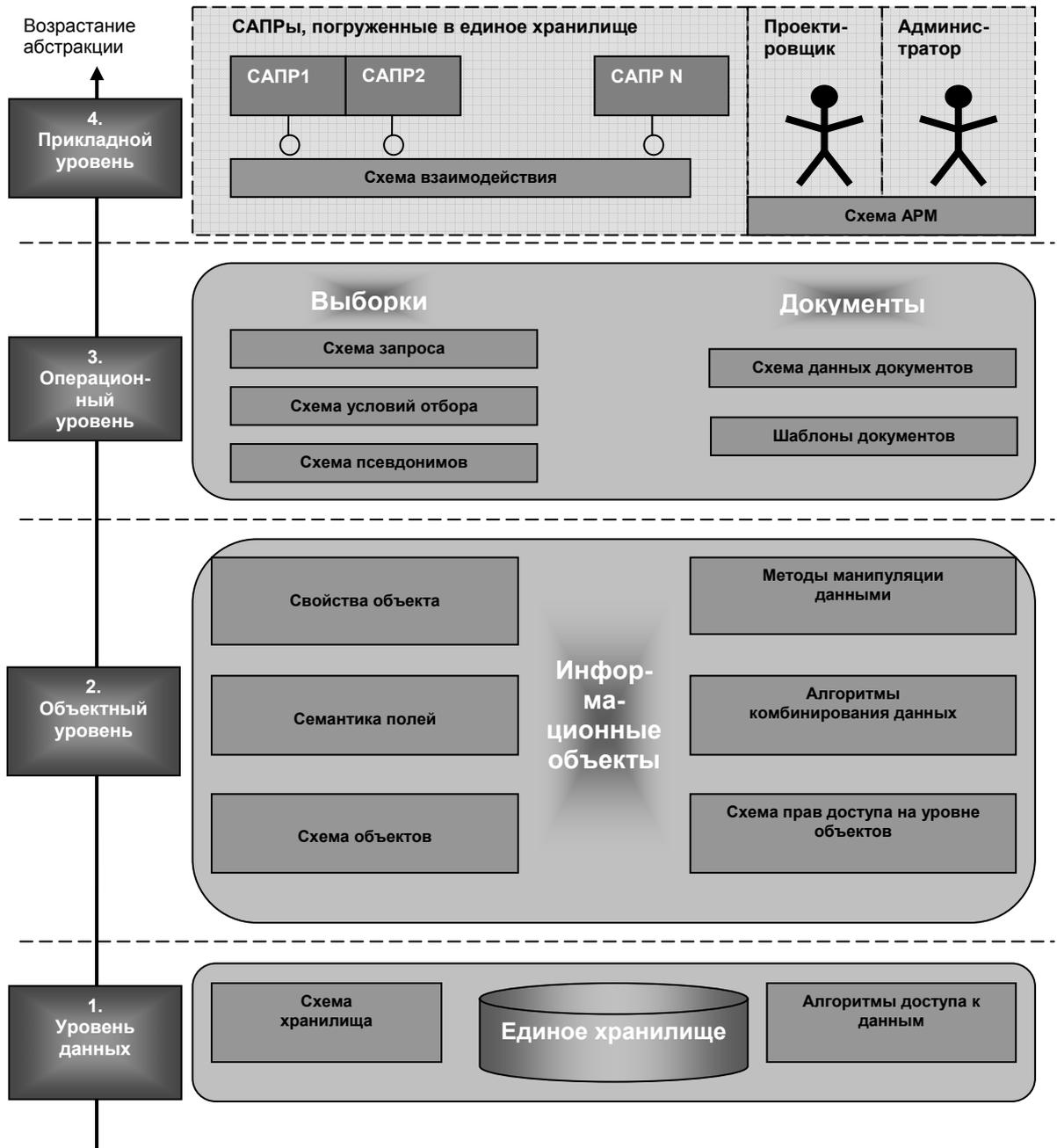


Рис. 1 Схема многоуровневого метаописания

Например, параметры оборудования размещены в доменах атрибутов $Attr_1 \dots Attr_n$, при формировании информационного объекта атрибут $Attr_1$ явно нотируется как потребляемая мощность, а $Attr_3$ – как пусковой ток.

Для манипулирования данными каждый информационный объект имеет соответствующие методы, работа с которыми происходит на основании прав доступа к информационным объектам.

Формально второй уровень R' представляет собой совокупность схем отображения R на R' :

$$\{R'_1, R'_2, \dots, R'_n\}, \text{ где:}$$

$$R'_j = \pi_{\{\sigma_{S=S'_j, CU=cu}(A)\}}$$

$$\left(\bigtriangleleft_{i=1}^m \delta_{\{S'_j \leftarrow \sigma_{S=S_j}(U)\}}(R_i) \right) \quad (1)$$

где n – количество информационных объектов, m – количество соединяемых отношений,

$$1 \leq m \leq p, 1 \leq j \leq n, R_i \subseteq R, S_i \subseteq R_i, \bigcup_j S_j = Sem,$$

π - реляционный оператор проекции,

δ - оператор переименования доменов,

σ - реляционный оператор выборки,

$\triangleright \triangleleft$ - реляционный оператор соединения,

Sem - множество, определяющее семантику полей,

A - функционал, определяющий права текущего пользователя CU:

$$A = \{A_1, A_2, \dots, A_c\}$$

$$A_i = (CU_i, S, \{Sel | Add | Upd | Del\}),$$

$$1 \leq c \leq кол - во_пользователей * card(Sem),$$

где $\{Sel | Add | Upd | Del\}$ - множество, задающее права доступа к свойству S для пользователя CU_i .

Согласно (1) ключевыми в R' являются множества Sem - семантики полей, подмножества множества R - схемы единого хранилища и A - права текущего пользователя. Таким образом, происходит абстрагирование от конкретной реализации единого хранилища, что позволяет перейти к уровню оперирования понятиями.

Операционный уровень представляет собой следующий набор метамodelей:

- модель извлечения данных;
- модель представления выбранных данных в табличной или графической формах

Одна из основных задач хранилища - это получение консолидированной информации, владельцами которой изначально являлись разные САПР (например, САПР1 осуществляет расчет токов в ветвях цепей электроснабжения и выбор соответствующих кабелей, САПР2 осуществляет расчет длин прокладки кабелей по желобам, при составлении же лимитно-заборной карты нам необходимо получить информацию об общей длине каждого типа кабеля). Методика реализации хранилища обеспечивает доступ к данной информации не в нотации структуры хранилища, а в терминах предметной области при помощи наложения произвольного количества условий на свойства информационных объектов (2). При этом метамodelь извлечения данных позволяет формировать формальные описания произвольного количества выборок по мере возникновения необходимости в них: $Q = \{Q_1, \dots, Q_n\}$.

$$Q_j = \pi(Val_1, \dots, Val_n)$$

$$(\sigma(Cond_1, \dots, Cond_n) (\triangleright \triangleleft_{i=1}^m R'_i)) \quad (2)$$

где n - количество построенных запросов, m - количество информационных объектов, участвующих в запросе, (Val_1, \dots, Val_n) - множество псевдонимов для отобранных данных,

$(Cond_1, \dots, Cond_n)$ - множество наложенных условий отбора данных,

π - реляционный оператор проекции,

δ - оператор переименования доменов,

σ - реляционный оператор выборки,

$\triangleright \triangleleft$ - реляционный оператор соединения.

Технология выполнения проектных работ требует представления выбранных данных в виде проектных документов как свободной, так и утвержденной стандартом формы. Для обеспечения возможности создания произвольного количества подобных документов разработана метамodelь представления данных.

$$Rp = f(\{Q_1, \dots, Q_n\}, Sh), \quad (3)$$

где n - количество запросов, используемых в документе, Sh - шаблон документа.

Прикладной уровень представляет собой следующий набор метамodelей:

- модель рабочего места пользователя системы (проектировщика);
- модель рабочего места администратора;
- модели взаимодействия хранилища и САПР.

В зависимости от объема должностных полномочий и потребностей в получении информации и формировании документов пользователи системы, построенной на базе единого хранилища, должны иметь различные интерфейсы взаимодействия с системой.

Для того, чтобы избежать традиционного программирования клиентских приложений, предусматривается формальное описание специфики рабочих мест проектировщиков.

$$A_{Pi} = f(F, R', Q, R_p, I) \quad (4)$$

где F - функции, которые может выполнять пользователь, R' - информационные объекты, доступные для данного пользователя, Q - запросы, доступные для данного пользователя, R_p - документы, доступные для данного пользователя, I - множество интерфейсных элементов (меню, панели инструментов и т.п.).

Модель рабочего места администратора аналогична модели рабочего места пользователя, но для администратора предусматривается полный доступ к объектам хранилища.

Очень существенным компонентом системы является модель взаимодействия САПР и единого хранилища.

$$M_{int} = M_{inp} \cup M_{out} \cup M_{reg}. \quad (5)$$

Эта модель предусматривает наличие формального описания данных, получаемых от данной САПР и схемы размещения их в хранилище, с одной стороны, (M_{inp}) и наличие формального описания данных (M_{out}), передаваемых из хранилища в данную САПР, с другой стороны, а также описание регламентов процедур взаимодействия (M_{reg}).

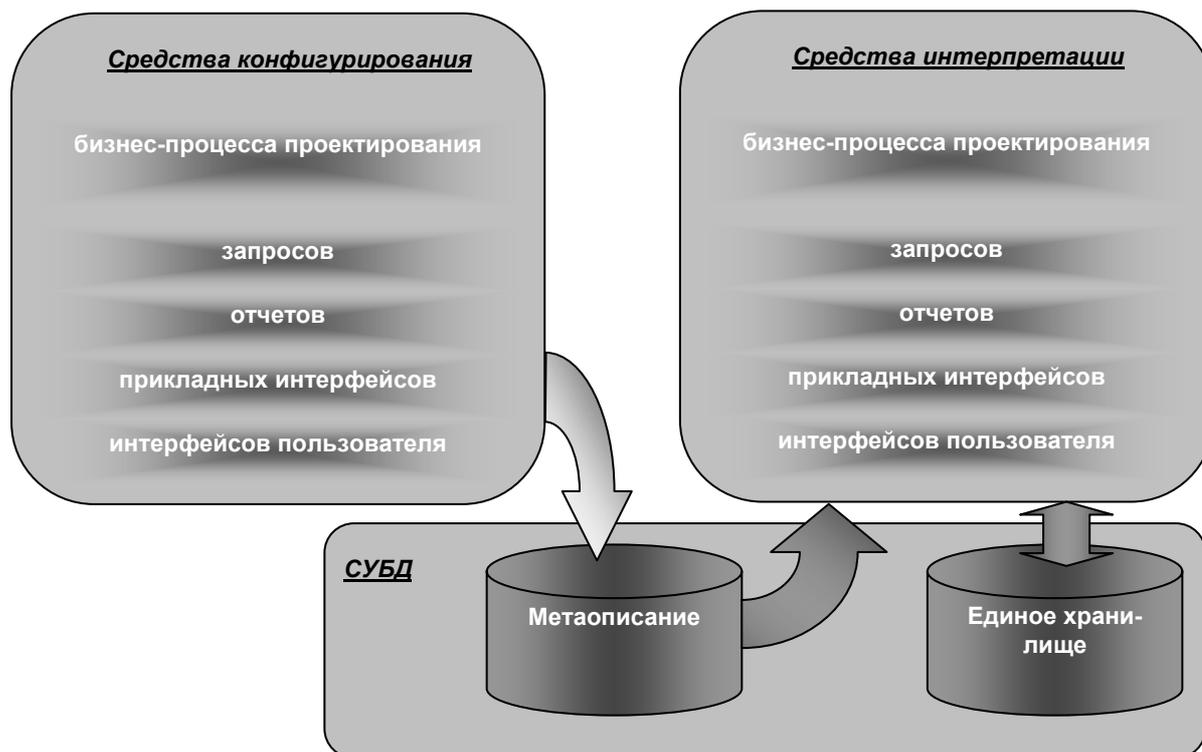


Рис. 2 Архитектура единой информационной среды проектирования

Таким образом, разработана модель метаописания всей совокупности аспектов, рассматриваемых при разработке информационной среды проектирования, включая в себя:

- структуру и организацию данных;
- методы представления данных и методы доступа к ним;
- алгоритмы формирования выборок и генерации документов;
- интерфейсы пользователей системы;
- интерфейсы взаимодействия проектируемых систем.

Предложенная концепция обеспечивает необходимую гибкость построенного таким образом хранилища за счет того, что при изменении информационных нужд погруженных САПР достаточно отобразить данные изменения в метаописании. Иными словами, если произойдет изменение бизнес-процесса проектирования, повлекшее за собой смену информационных потоков на его входе и выходе, то этот факт достаточно отобразить в метаописании, на основании которого произойдет автоматическая настройка средств взаимодействия единого хранилища с программным обеспечением, реализующим бизнес-процесс.

Для обеспечения формирования и изменения метаописания разработан набор специализированных инструментальных средств – *конфигураторов*. Необходимо отметить, что удобный пользовательский интерфейс открывает проектировщику возможность непосредственного участия в создании и поддержке жизненного цикла среды единого хранилища, за счет чего модель бизнес-процессов проектирования приближается к существующей в проектной организации, а этим достигается более эффективная эксплуатация единого хранилища.

Кроме конфигураторов предусмотрены также *интерпретаторы* – специализированные программные средства, предназначенные для получения информации из метаописания и формирования рабочей среды проектирования в соответствии с настройками, сделанными при конфигурировании.

Архитектура предлагаемой системы приведена на Рис. 2.

Использование данной системы позволяет существенно сократить сроки разработки и внедрения единой среды проектирования сложных электротехнических объектов, обеспечить высокую гибкость и открытость системы, гарантировать сохранность инвестиций при изменении технологии проектирования и внешних условий существования проектных организаций.

Литература

1. Мейер Д. Теория реляционных баз данных: Пер. с англ. – М.: Мир, 1987 – 608 с.
2. Чумаков В. Анализ + проектирование + разработка прототипа прикладной системы с помощью Designer/2000 // Русское издание Oracle Magazine. – 1997. – № 3(5).
3. Кайт Т. Oracle для профессионалов. Кн. 1. Архитектура и основные особенности / Пер. с англ. Т. Кайт. – СПб.: ООО «ДиаСофтЮП», 2004. – 672 с.
4. Разработка информационно-управляющих систем WWW-WorkFlow в инструментальной среде RADIUS / Т.Я. Кроль, Е.Р. Пантелеев, Е.А. Бабасин и др. // Образовательные технологии: Межвуз. сб. научных трудов. Вып.7. – Воронеж: ВГПУ, 2001. – С. 172–178.
5. Кроль Т.Я., Пантелеев Е.Р. Технология разработки и эксплуатации корпоративных информационно-управляющих систем в среде RADIUS. Конструкторско-технологическая информатика – 2000: Труды IV международного конгресса. – Москва: СТАНКИН, 2000. – Т.1. – С. 307–308.