

Волжский Политехнический Институт (филиал) ВолгГТУ

«Информатика и вычислительная техника»

Разработка и исследование библиотеки хранимых процедур
для получения количественных метрик физических схем баз
данных.

Автор: Фатеенков М.М.

Научный руководитель: Рыбанов А.А., к.т.н., доцент

Постановка задачи

Цель работы: повышение эффективности разработки баз данных. Для достижения поставленной цели необходимо решение следующих исследовательских задач:

1. Математическое описание метрик физических схем баз данных.
2. Алгоритмы и программная реализация библиотеки хранимых процедур.
3. Анализ предлагаемого математического описания, алгоритмов и библиотеки хранимых процедур.

Метрики баз данных

- 1) Глубина дерева связей схемы БД/таблицы
- 2) Количество атрибутов и внешних связей схемы
- 3) Коэффициент внешних связей таблицы
- 4) Структурная избыточность схемы базы данных
- 5) Абсолютная глубина
- 6) Связность схемы бд
- 7) Средняя глубина схемы бд
- 8) Диаметр и радиус схемы бд
- 9) Локальный коэффициент кластеризации
- 10) Центр схемы бд
- 11) Реберная плотность схемы бд
- 12) Центр тяжести графа

Метрики баз данных

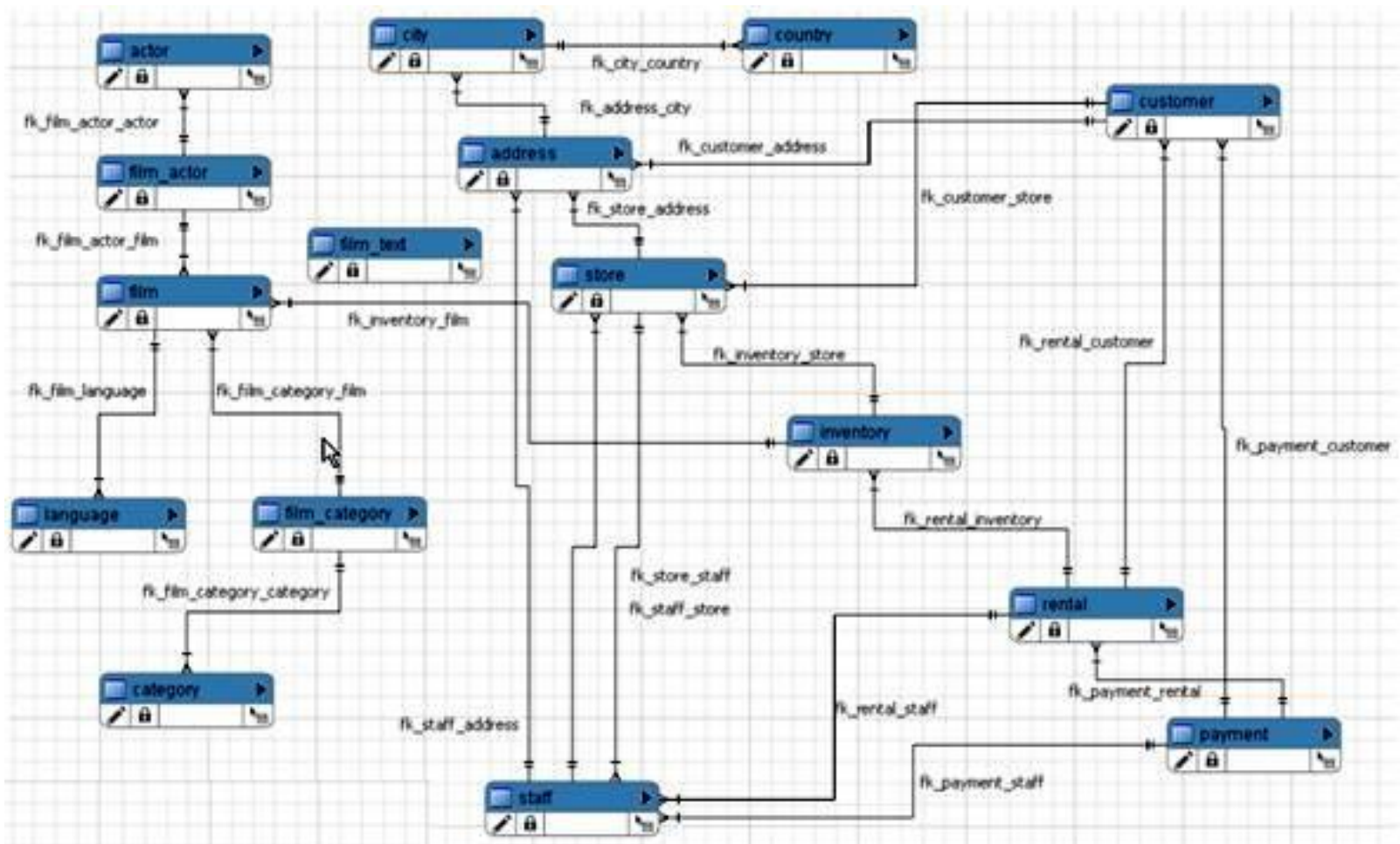
Расчет метрик физических схем баз данных рассмотрим на примере СУБД MySQL. Исходные данные для получения метрических характеристик MySQL базы данных хранятся в базе данных метаданных - information_schema.

Метрики баз данных

Глубина дерева связей таблицы – это расстояние от ее конкретной таблицы, до элемента самого нижнего уровня.

Глубина дерева связей схемы БД – это максимальное расстояние от её корневого элемента, до элемента самого нижнего уровня.

Физическая схема базы данных



Физическая схема базы данных «Sakila»

Варианты вызова хранимой процедуры

- 1) Получение метрических характеристик для всех таблиц базы данных:

GetDBTablesDepth(<имя базы данных>);

- 2) Получение метрических характеристик для конкретной таблицы базы данных:

GetDBTablesDepth(<имя базы данных>, <имя таблицы>);

Результат работы хранимой процедуры

Имя таблица	Глубина дерева связей таблицы
payment	7
rental	5
inventory	5
staff	4
store	4
customer	3
address	2
film_category	2
city	1
film	1
film_actor	1
language	0
actor	0

Анализ кода хранимой процедуры.

Для анализа метрических характеристик было использовано CASE-средство SQLDetective 4.3. Данное инструментальное средство позволяет получать метрики только для PL/SQL кода, поэтому для анализа кода процедуры был применен онлайн конвертор из MySQL в PL/SQL – SQLLine (<http://www.sqlines.com/online>).

Метрики кода хранмой процедуры

Метрика	Значение метрики
Цикломатическая сложность (Cyclomatic Complexity)	28
Комплексный показатель качества кода (Maintainability Index)	4
Количество строк программы (LOC)	284
Процент комментариев (Comment Pct.)	13
Объем программы по Холстеду (Halstead Volume)	2118
Длина программы (Program Length)	336
Словарь программы (Program Vocabulary)	79
Сложность программы (Difficulty)	39
Усилия при разработке (Effort)	82605
Сложность интерфейса (Interface Complexity)	5
Входящие параметры (Input parameters)	5
Обязательные входящие параметры (Required Input parameters)	5
Точки возврата (Return Points)	0
Функциональная сложность (Functional Complexity)	33
Пустые строки (Blank Lines)	175
Строки комментариев (Comment Lines)	71
Эффективные строки кода (eLOC)	238
Логические операторы кода (lsLOC)	81
Строки псевдокода (Pseudocode Lines)	0

Анализ полученных метрик кода хранимой процедуры

При анализе кода хранимой процедуры, основными будут являться 3 метрики:

- 1) Цикломатическая сложность
- 2) Комплексный показатель качества кода
- 3) Объем программы по Холстеду

Анализ полученных метрик кода хранимой процедуры

Формула расчета Цикломатической сложности

$$V(G) = e - n + 2p$$

где e – количество дуг, n – количество вершин, p – число компонентов связности

Градация значений Цикломатической сложности

Значение метрики	Риск	Вероятность возникновения проблем при исправлении ошибок
1-10	Низкий	5%
11-20	Средний	10%
21-50	Высокий	30%
>50	Очень высокий	40%

Анализ полученных метрик кода хранимой процедуры

Формула расчета Комплексного показателя качества кода

$$M_I = \text{MAX}(0, (171 - 5.2 * \ln(HV) - 0.23 * CC - 16.2 * \ln(LoC)) * 100 / 171),$$

где HV – вычислительная сложность, CC – цикломатическая сложность, LoC – количество строк кода.

Градация значений Комплексного показателя качества кода

Высокая сложная поддержки	Средняя сложность поддержки	Низкая сложность поддержки
0-9	10-19	20-100

Анализ полученных метрик кода хранимой процедуры

Формула расчета Объема программы по холстеду

$$V = N * \log_2(n)$$

где N – длинна программы, n – словарь программы

Градация значений Объема программы по холстеду

Для функции	20 – 1000
Для одного файла	1000 – 8000

Анализ полученных метрик кода хранимой процедуры

Исходя из полученных данных и нормальных показателей основных метрик мы видим, что необходимо оптимизировать имеющуюся хранимую процедуру, для упрощения последующего её изменения, путем разбиение на более мелкие хранимые процедуры.

Вывод

Представленный подход получения метрических характеристик для физической схемы MySQL-базы данных, основанный на метаинформации `information_schema` и реализации хранимых процедур, с последующей оценкой их качества с помощью SQLDetective 4.3, может использоваться для создания библиотеки хранимых процедур оценки метрик физических баз данных. Которая позволит существенно облегчить разработку баз данных и дальнейшую их оптимизацию.

Спасибо за внимание.