

## ОРГАНИЗАЦИЯ ЗАПРОСОВ В РАСПРЕДЕЛЕННОЙ СИСТЕМЕ УПРАВЛЕНИЯ БЕЗОПАСНОСТЬЮ ПОЛЕТОВ

Институт авиационный университет  
Национальный авиационный университет

*Исследованы параметры информационно-вычислительных систем обработки полетной информации, в которых решается задача анализа и прогнозирования уровня безопасности полетов. Разработан алгоритм организации запросов в распределенной базе данных для цели управления безопасностью полетов. Предложена распределенная база данных многоуровневой модели ANSI-SPARC, которая обеспечивает минимальную стоимость запросов как в штатных, так и в экстремальных ситуациях управления безопасностью полетов*

### **Введение**

Концепция безопасности полетов является основным понятием общей проблемы авиационной безопасности.

В работе [1] рассмотрен вопрос о создании базы данных безопасности полетов (БД БП), с целью эффективного анализа полученной информации, в том числе по результатам расследования авиационных событий и добровольных извещений об опасных факторах, и разработках профилактических мероприятий.

### **Постановка задачи**

При построении такой базы, за основу была принята архитектура ANSI-SPARC, которая является результатом многолетнего исследования, в первую очередь того, как может поддерживаться независимость данных в системе баз данных. Система управления базами данных и ее приложения имеют длительный срок действия, в то время как средство накопления данных или внешние интерфейсы модифицируются или расширяются на протяжении времени. Целью является разработка базы данных, которая является независимой от необходимых изменений приложений.

### **Решение задачи**

Более детальная версия первичной 3-х уровневой архитектуры – это 3-х уровневая архитектура ANSI-SPARC с пятью слоями и детализацией системы

управления базами данных. В сравнении с 3-х уровневой архитектурой более конкретно описываются услуги обработки данных.

На рис. 1 представлена подробная структура пяти слоев архитектуры. В представленной модели занесенные в систему управления базой данных компоненты преобразования описываются более точно. В блоках преобразования реализуется трансформация запросов и изменения абстрактных уровней моделей базы данных вниз к доступам в среде хранения. Кроме того, между компонентами определяются интерфейсы. Можно структурировать физические слои архитектуры. При этом на более высоком слое реализуются более сложные функции с использованием данных от более низких слоев.

Обзор компонентов преобразования следующий:

*Система обработки входящих данных* – преобразует пользовательский запрос в формат внутреннего представления.

*Система доступа* – осуществляет поддержку всех механизмов логического доступа (транзакции, блокировки), а также контролирует права доступа к структурным элементам базы.

*Исполняющая система* – выполняет пользовательский запрос преобразованный системой обработки входящих данных в формат внутреннего

представления, под контролем системы доступа.

*Управление буфером* – сбор и хранение результатов работы исполняющей системы.

*Операционная система* – поддержка системных вызовов (имеется в виду операции с памятью и различными устройствами).

Обзор интерфейсов между компонентами:

*Массово ориентированный интерфейс* – декларативный механизм манипулирования данными (специализированный программный продукт или язык запросов, например *SQL*). Специализированный программный продукт – набор приложений, который предоставляет пользователю возможность оперировать понятиями конкретной прикладной задачи.



Рис. 1. Пять слоев архитектуры

*Интерфейс, ориентированный на приложение* – ведущий доступ к приложениям, логическим файлам и логическим путям доступа

*Внутренний интерфейс приложения* – манипуляция приложениями и путей доступа.

*Инструментальный интерфейс* – передача результатов работы исполняющей системы для сбора и хранения.

*Интерфейс системы* – передача системных вызовов в операционную систему.

*Интерфейс устройств* – манипуляция устройствами, то есть, управляет аппаратными средствами через программы драйвера операционной системы.

Для оценки и сравнительного анализа различных методов организации доступа к данным, целесообразно

$$t_3 \leq T_{\max} \text{ при } V_{\text{инф}} \leq V_{\max}, \text{ при } t_3 > T_{\max} \quad C = a_2 V_{\text{инф}},$$

где  $a_1$  и  $a_2$  – весовые коэффициенты, подбираемые экспериментально, по результатам анализа состояния каналов передачи.

Если  $a_1 > a_0$ , где коэффициент  $a_i$  определяется методом экспертных оценок на основе результатов обработки статистических данных, тогда в этом случае, в качестве I приближения предлагается линейно-ломанная зависимость (рис. 2).

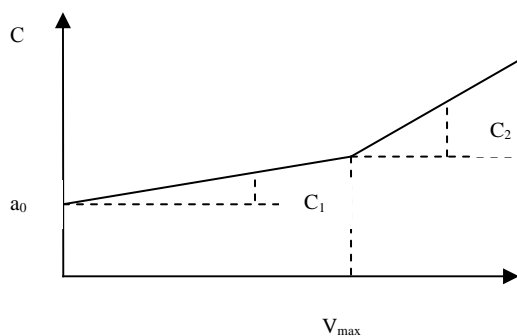


Рис. 2. Линейно-ломанная зависимость

использовать критерий "эффективность/стоимость" как наиболее универсальный [2].

Под эффективностью  $E$  в данной задаче подразумевается некоторый функционал скорости передачи запросов  $V_q$  и достоверности передачи, которая выражается через ошибки первого и второго рода с соответствующими вероятностями  $P_I$  и  $P_{II}$  [3]:

$$E = \psi(V_g, P_I, P_{II})$$

Стоимость определяется по формуле:

$$C = a_1 V_{\text{инф}} + a_0,$$

1. При ограничениях на время занятия канала при гарантированном качестве обслуживания  $QoS$ :

Альтернатива – качество сервиса не гарантируется (обслуживание по соглашению *Best Effort*) [4].

$$C = a_3 V_{\text{инф}} + a_{03}$$

Обеспечение передачи данных – проблема оператора (провайдера сети).

В этом случае также остановимся на линейно-ломанной зависимости в качестве I приближения, но с другим коэффициентом и точкой излома (рис. 3).

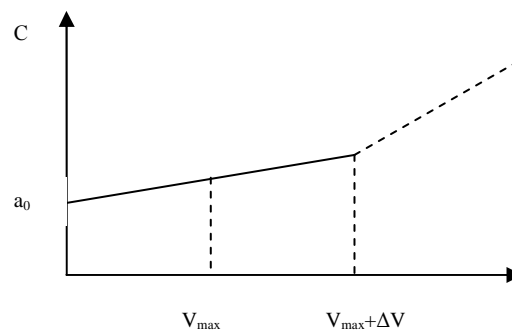


Рис. 3. Линейно-ломанная зависимость

3. Форс – мажорная ситуация (летное происшествие, предпосылки и т.д.), качество обслуживания  $QoS$  максимальное, при этом стоимость является вторым приоритетом.

Необходимо обеспечить гарантированный объем канала –  $V_{канал} \geq V_{инф}$

Задача: определить требуемый объем канала.

В штатной ситуации объем перерабатываемой информации ~ в 10 – 100 раз меньше, чем при возникновении нештатной ситуации. Тогда, стоимость растет по закону  $C_{max} = (1 - \exp^{-a_4 V_{info}})$  и график выглядит следующим образом (рис. 4)

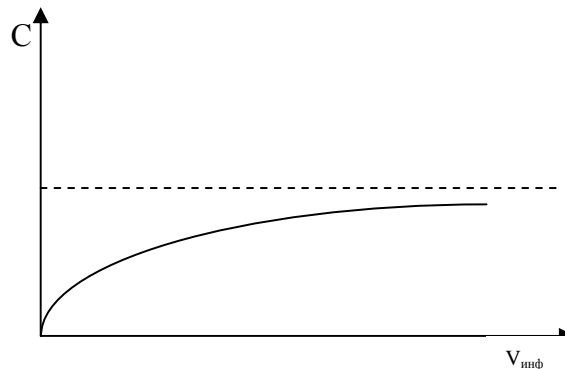


Рис.4. Экспоненциальная зависимость

### Выводы

Для управления безопасностью полетов и обработки нештатных ситуаций необходимо применять систему управления с переменными параметрами и метод многоуровневой оптимизации запросов в распределенной базе данных.

В дальнейшем планируется исследовать потенциальную точность оценки весовых коэффициентов в предложенных моделях, используя набранную статистику летных происшествий.

### Список литературы

1. Холявкина Т.В. Интегрированная система сбора и обработки информации в распределенной системе анализа безопасности полетов / Т.В. Холявкина // Наукові записки Українського Науководослідного інституту зв'язку. Зб. наук. пр. – 2009. – №1(9). – С. 51–57.

2. Маусеев Н.Н. Математические задачи системного анализа. – М.: "Наука", 1981. – 487 с.

3. Леман Э. Проверка статистических гипотез. – М.: "Наука", 1964. – 498 с.

4. Олифер В.Г., Олифер Н.А. Компьютерные сети. Принципы, технологии, протоколы: Учебник для вузов. 3-е изд. – СПб.: Питер, 2006. – 958 с.