

Особенности реализации учетных задач в условиях существенной динамики бизнес-процессов

**Андрей Гордиенко
апрель 2003 г.**

Виды учетных задач

- Бухгалтерский учет
- Налоговый учет
- Управленческий учет
- Оперативный учет
- ...

Типичные проблемы реализации учетных задач

Жесткость приложения

- Сложность настройки под конкретное место внедрения (если таких мест много)
- Сложность внесения изменений в ранее внедренное приложение при изменении условий ведения бизнеса

Недостатки инструментария (РСУБД)

- Взаимное проникновение OLTP и OLAP. Низкая оперативность OLAP-обработки.
- Сложность настройки репликации
- Мониторинг и управление уже настроенной репликацией должны производить специалисты прикладной области, а не DBA
- Неоднозначность понятия «объект». Изменение со временем характеристик, идентифицирующих объект (наименование, ИНН, номер паспорта, ФИО и т.п.)
- Контроль за вводом, модификацией и удалением информации должен осуществляться руководителями соответствующих подразделений, а не DBA

Схемы реализации учетных задач

ЖЕСТКАЯ

Факты (документы)

Приход
(накладная)

Расход (акт
списания)

Другие
факты (...)

SQL-запросы
(учитывают нюансы каждого документа)

Отчеты

ГИБКАЯ

Факты (документы)

Приход
(накладная)

Расход (акт
списания)

Другие
факты (...)

Бизнес-логика отражения фактов

Движение
товаров

Взаимо-
расчеты

Движение
денег

Аналитические регистры

SQL-запросы
(простые, только по аналитическим
регистрам)

Отчеты

Понятия

Специалист по настройке и сопровождению

Для специалиста в области ИТ

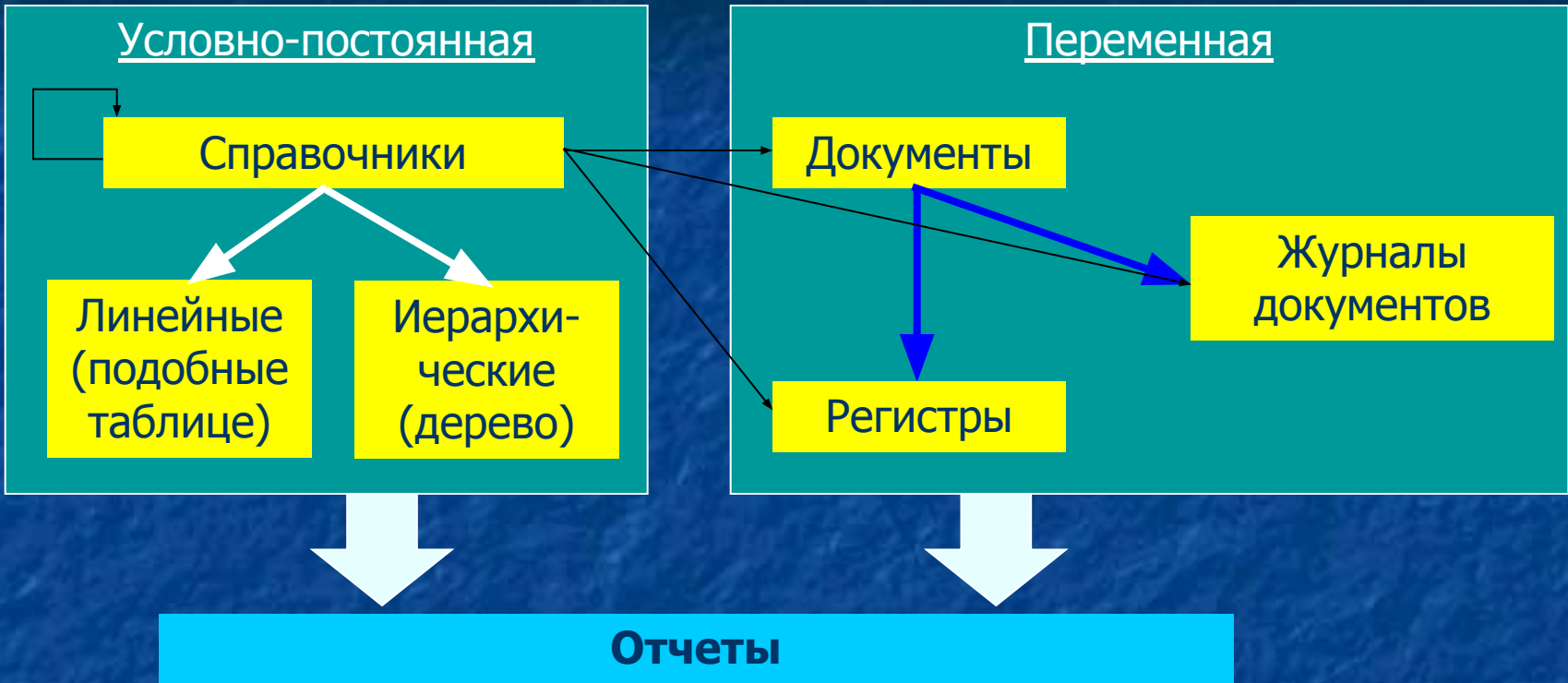
- Таблицы
- Хранимые процедуры
- Индексы
- Представления
- Триггеры
- Отчеты

Для специалиста в прикладной области

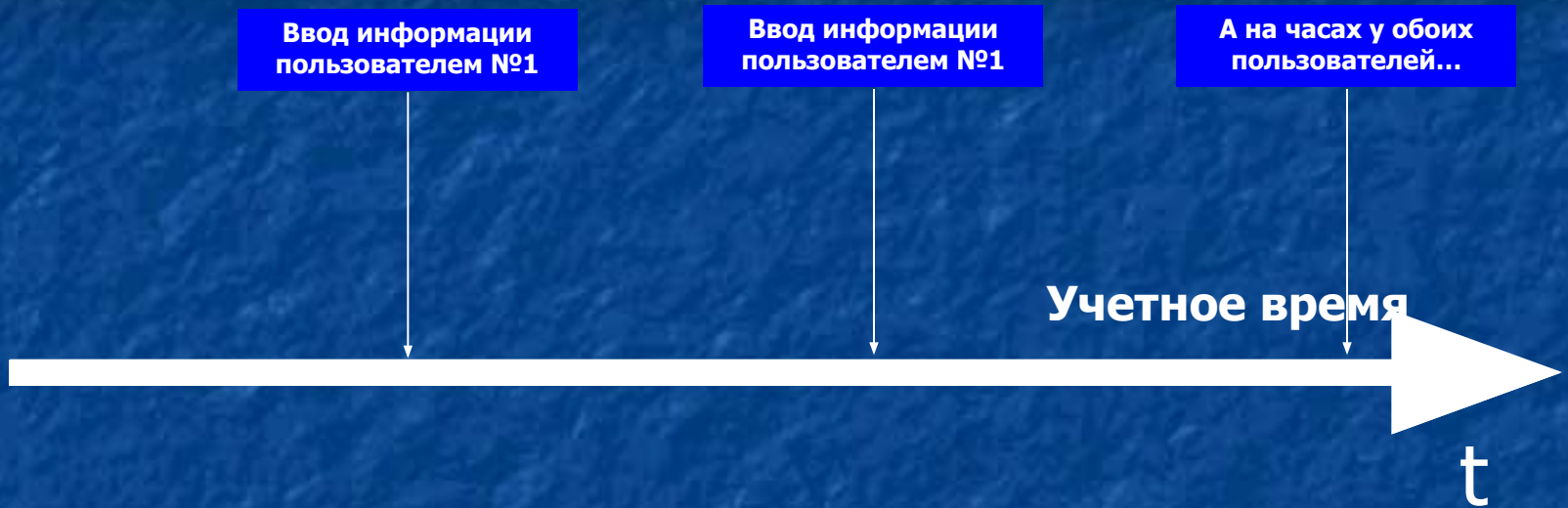
- Документы
- Журналы
- Реестры
- Регистры (бух.счета)
- Отчеты



Информация



Понятие времени



Календарное время

Идентификация объектов на оси учетного времени



Объект, о котором необходимо хранить информацию

Род

**НЕОБХОДИМО
ИСПОЛЬЗОВАТЬ
СУРРОГАТНЫЕ
КЛЮЧИ !**

Иванов
Василий
Без па

Об этом объекте
тоже нужно хранить
информацию



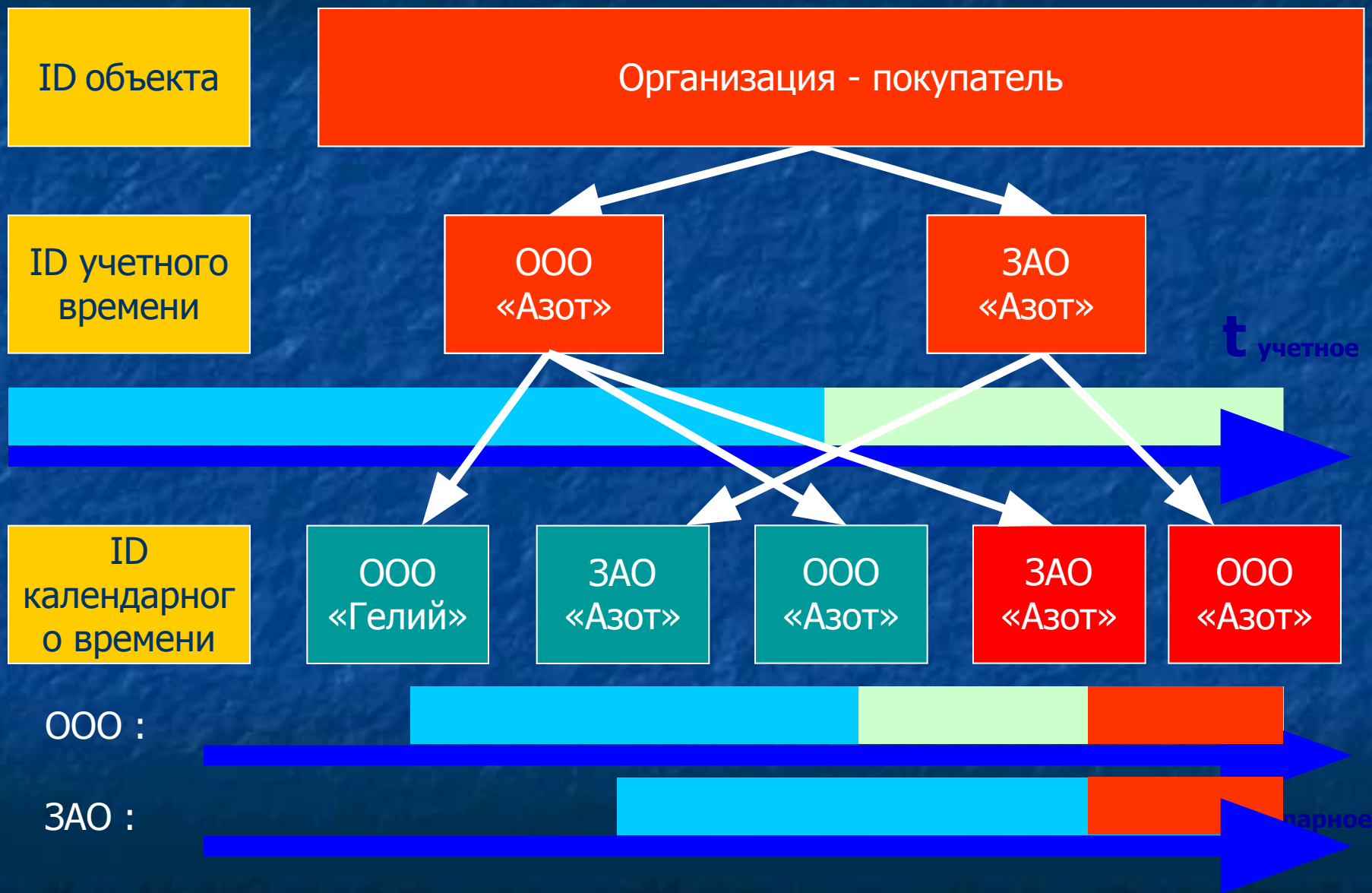
Родился



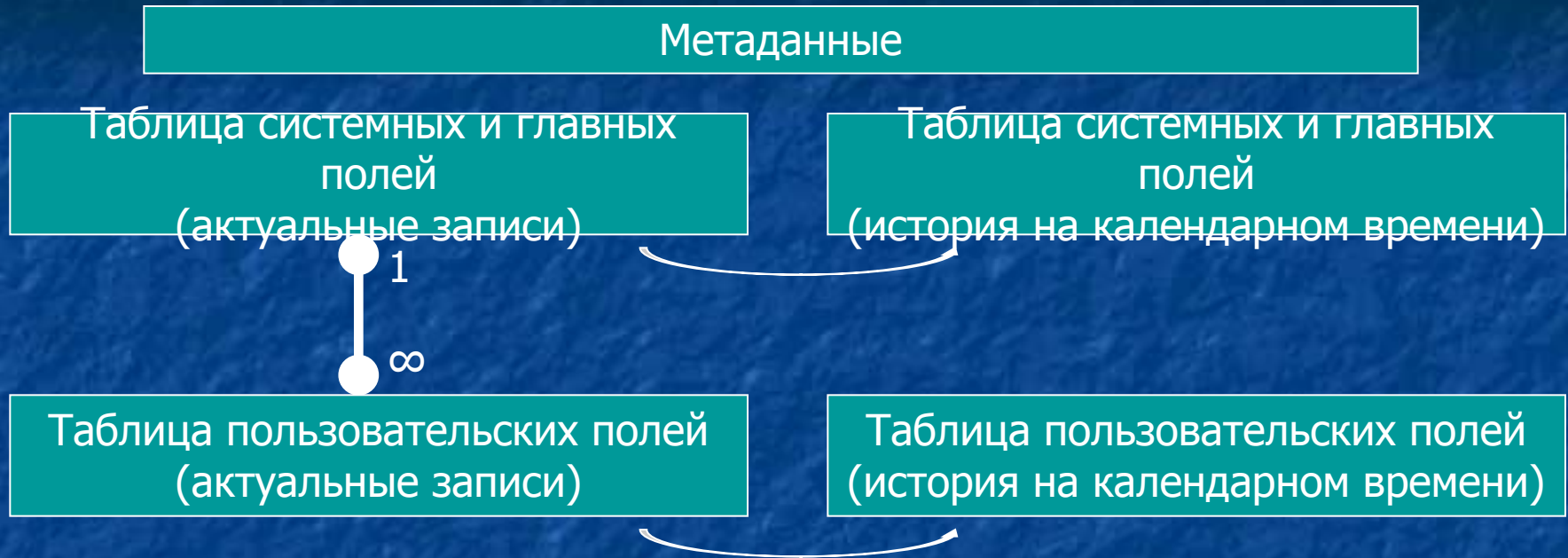
Сидоров Сидор
Сидорович
Без паспорта

А это уже другой объект

Идентификация учетных объектов на осях времен



Структура таблиц справочников

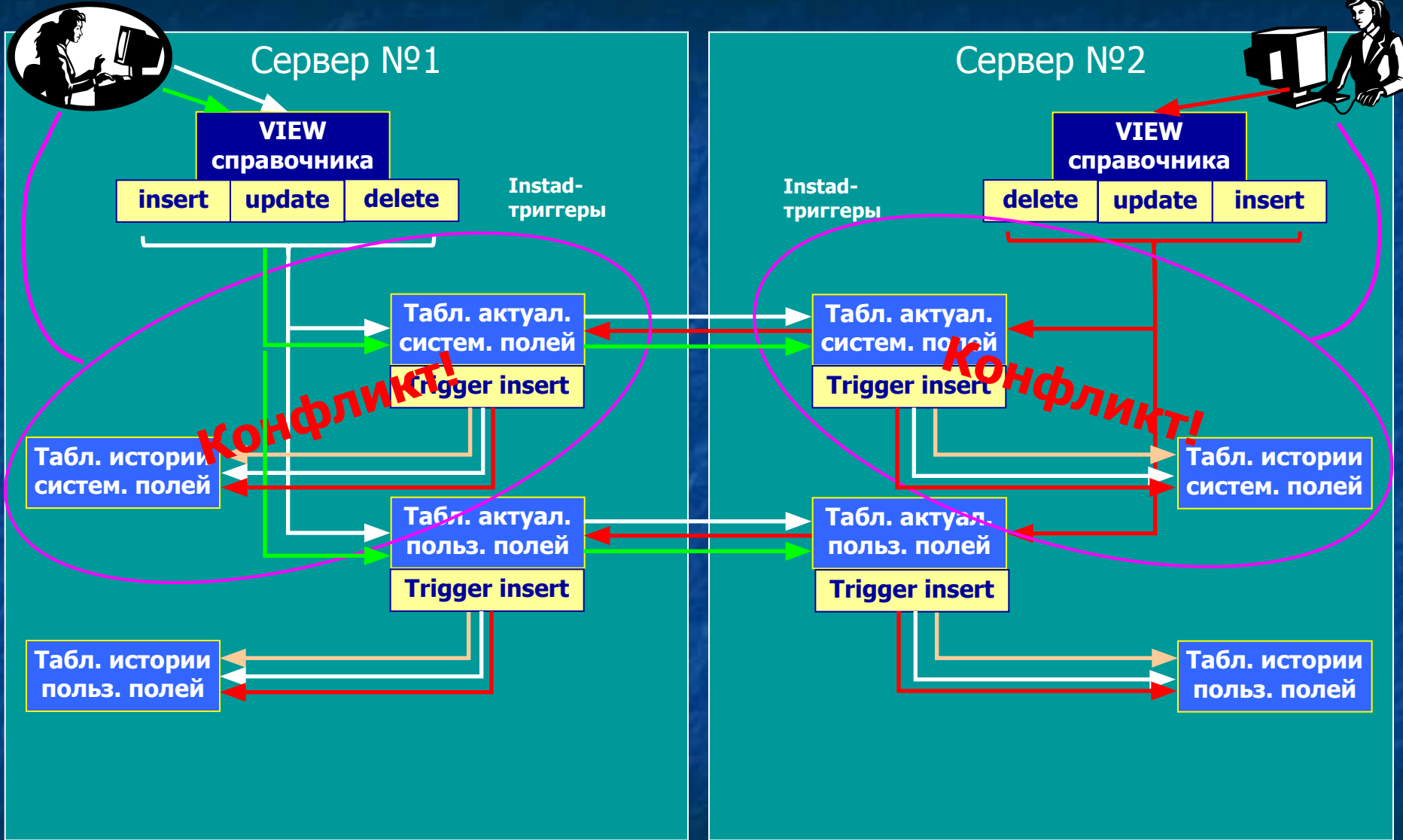


- Справочник создается после ввода в таблицы метаданных его описания и установки флага готовности метаданных.
- Справочник представляется тремя VIEW с instead-триггерами, скрипт создания которых генерится автоматически.
- Для хранения значений пользовательских полей используется тип `sql_variant`. Null значения НЕ хранятся.
- В отличие от ALTER TABLE можно удалить столбец, а затем вернуть обратно вместе с данными.

Репликация справочников

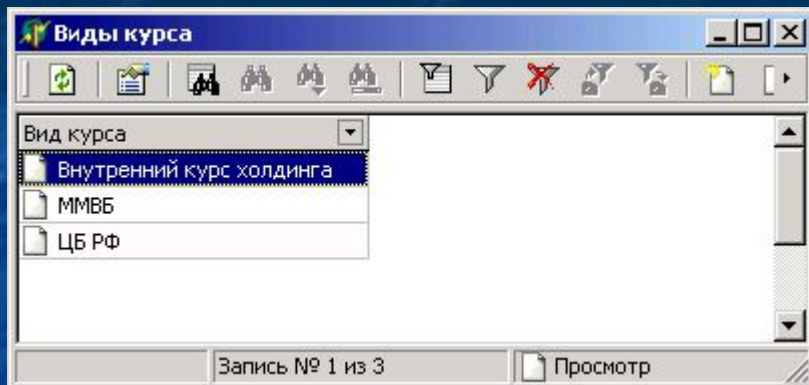
- Все три идентификатора типа uniqueidentifier (GUID), поэтому отсутствуют конфликты репликации, которые могут возникать на полях с identity.
- В репликации из 4-х таблиц участвуют только две – таблицы актуальных записей. Остальные формируются автоматически триггерами.
- Поскольку в две таблицы, участвующие в репликации, записи только добавляются (insert) конфликты репликации на уровне взаимодействия SQL-серверов отсутствуют.
- Для всех серверов используется единая схема приема актуальности записей на основе глобального календарного времени (например GMT). Актуальной считается последняя по глобальному времени запись.
- Управление репликацией можно производить на уровне логики высокого уровня с помощью программно управляемых фильтров, накладываемых на две реплицируемые таблицы.
- В случае выполнения на разных серверах конфликтных действий по отношению к одним и тем же данным сервера автоматически приводятся в единое состояние, и на обоих серверах в истории модификации записи сохраняется информация о конфликте. Конфликт может быть разрешен специалистом прикладной области восстановлением нужной версии записи.

Схема репликации справочников

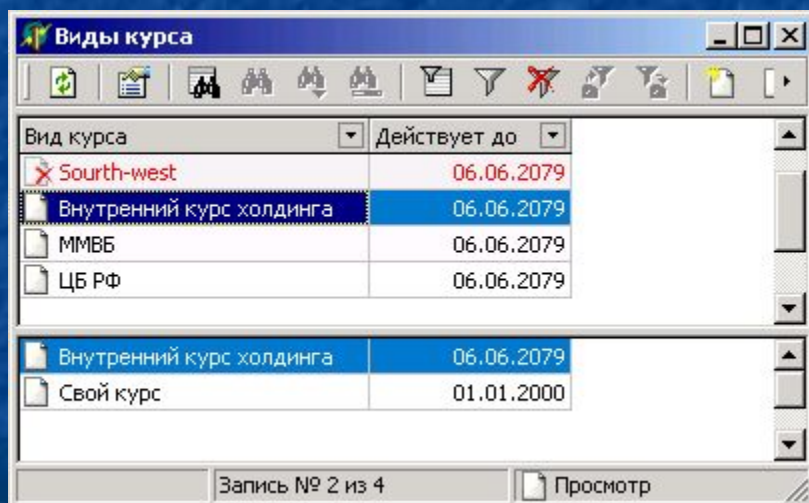


Визуализация информации справочника по осям времен

Обычный вид

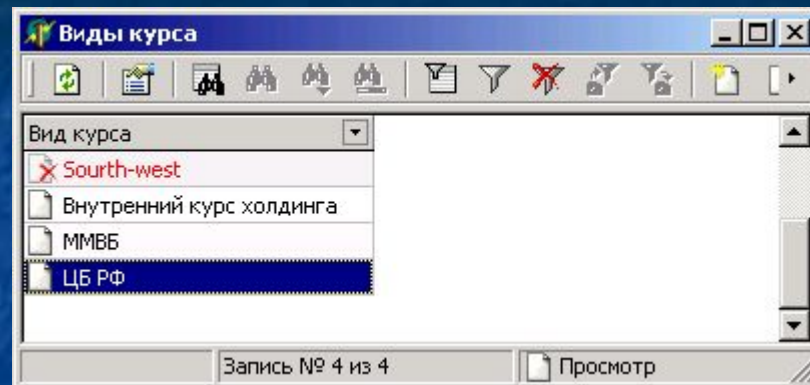


С отображением версий записей на оси учетного времени

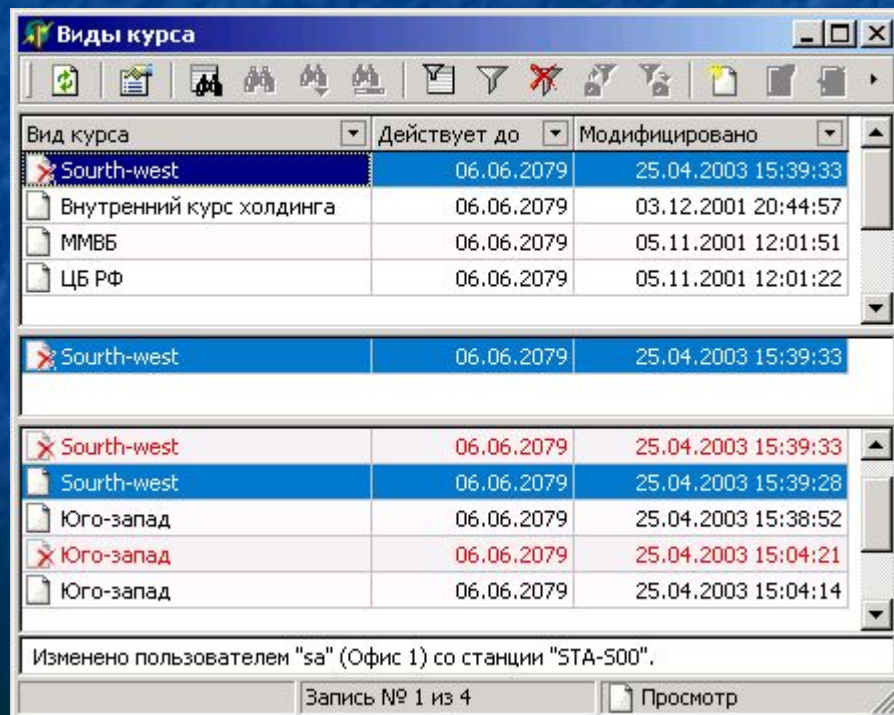


Можно отобразить только удаленные записи (поиск пропавшей записи)

Включая удаленные записи



Полная информация по всем осям времен

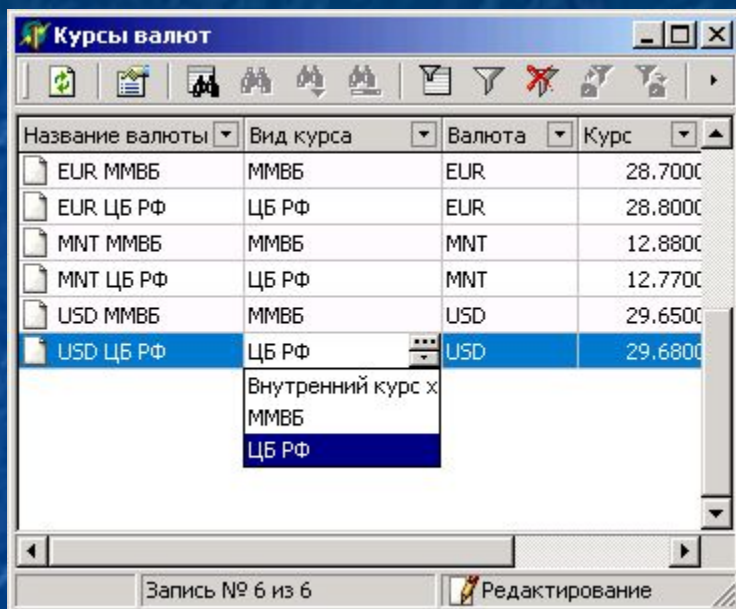


Типы полей справочников

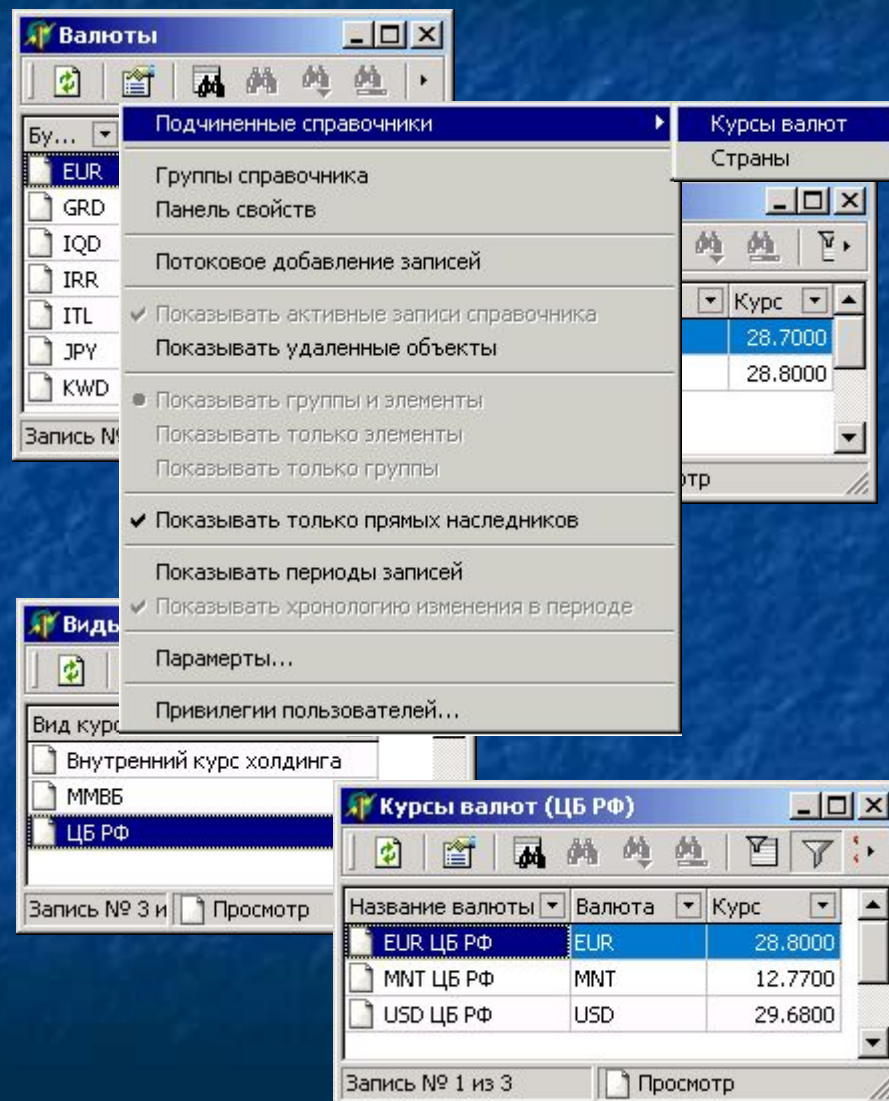
- Целое число [int]
- Строка [varchar(256)]
- Дата и время [datetime]
- Деньги [money]
- Логическое [bit]
- Число с плавающей точкой [real]
- Число с плавающей точкой двойной точности [float(53)]
- Число с фиксированной точкой [numeric(19,4)]
- Тип SQL
- **Измеряемая величина [float]**
- **Одно значение из набора [uniqueidentifier]**
- **Множество значений из набора [uniqueidentifier]**
- **Ссылка на справочник [uniqueidentifier]**
- **Множество значений из справочника [uniqueidentifier]**

Два варианта работы связанных справочников

Вариант 1.



Вариант 2.

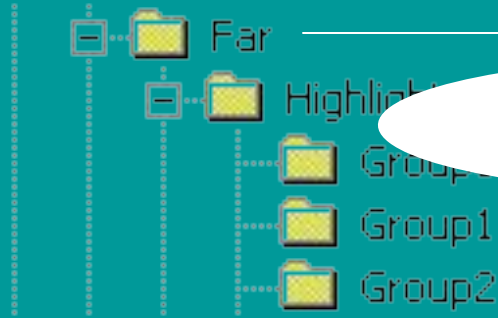


Используя механизм подчиненных справочников, можно ступенчато открывать не только два справочника, а произвольное их количество

Иерархические справочники

Жесткий состав полей на всех уровнях:

Справочник 1



Поле1	Поле2	Поле3	Поле4

Можно закрепить поля за элементами и именами группами

Иерархия с различным составом полей на разных уровнях

Справочник 1

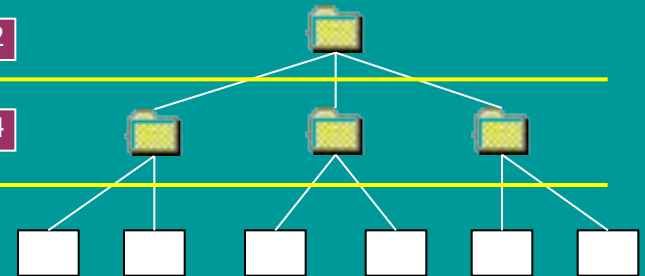
Поле 1 | Поле 2

Справочник 2

Поле 3 | Поле 4

Справочник 3

Поле 5



Подчинение справочников друг другу

Одинаковый состав полей в разных ветвях дерева

Свойства и механизм наследования в иерархических справочниках

Наследование перечня свойств.

На разных уровнях единый перечень, как в случае с полями.

Уровень 1

Поле 1

Свойство1 Свойство2

Есть перечень свойств, но нет их значений

Свойство3="18/04/2002"

Уровень 2

Поле 1

Свойство1=77 Свойство2=«RU»

Назначение значений свойствам

Свойство1=88 Свойство2=«US»

Уровень 3

Поле 1



Уровень 4

Поле 1

Свойство1=77 Свойство2=«RU»

Свойство1=88 Свойство2=«US»

Наследование перечня свойств вместе со значениями приводит к эффекту наличия различных default-значений для разных ветвей справочника

Свойство3="18/04/2002"

В этой части дерева собственный набор свойств. С помощью полей такое сделать невозможно.

Визуализация иерархического справочника со свойствами и дозирование информации по запросам на сервер

Перечень свойств расположен вертикально

Наименование

Наименование	Баз...
Двиг. 7,5 / 3000 общепромышл. Лапы 220/380 В, ...	шт.
Двиг. 7,5 / 3000 общепромышл. Комбиниров 220/...	шт.
Двиг. 7,5 / 3000 общепромышл. 220/380 В, 50Гц	шт.

Базовые характеристики дви...
Взрывозащита
Монтажное исполнени...
Мощность двигателя
Напряжение питания д...
Обороты двигателя
Дополнительные характе...
Посадочные характеристики ...
Специальные характеристик...
Примечание для двига...
Производитель двигателя
Скольжение

Общепромыш...
Лапы
7,50 Квт
220/380 В, 5...
3000

Критерий группировки: "Мощность / обороты / взрывозащита / монтаж.исполн-е" | Запись № 1 из 3 | Просмотр

Перечень полей расположен горизонтально

Перечень свойств (иерархический) вместе со значениями свойств. Критерий группировки так же как и по полям, помогает быть задействован выводом поддерживаемого автофильтра. Свойства находятся внутри выделенной в дереве группы.

Записи табличной части

Свойства сопоставлены только одной текущей записи табличной части

Визуализация иерархического справочника со свойствами по осям времен

Номенклатура

Корень

- Двигатели
 - Нераспределенные двигатели**
 - Простые двигатели
 - Двиг. 100,0 квт
 - Двиг. 1000,0 квт
 - Двиг. 11,0 квт
 - Двиг. 110,0 квт
 - Двиг. 1250,0 квт
 - Двиг. 132,0 квт
 - Двиг. 15,0 квт
 - Двиг. 160,0 квт

Наименование	Б...	Действ...	Модифициро...
Абракадабра	шт.	06.06.2079	27.04.2003 0:11...
Test5_1__	шт.	06.06.2079	21.02.2002 10:3...
Test4	шт.	06.06.2079	16.01.2002 18:3...
Test3	шт.	06.06.2079	16.01.2002 18:3...
Test2	шт.	06.06.2079	16.01.2002 18:3...
NewTest	шт.	06.06.2079	13.02.2002 12:0...

Модификация нескольких свойств отражается в удаленных свойствах.

Модификация свойств не приводит к созданию полной копии записи

Технические характеристики

- Характеристики двигателей
 - Базовые характеристики дви...
 - Мощность двигателя
 - Обороты двигателя
 - Специальные характеристик...
 - Примечание для двига...
 - Скольжение

Абракадабра	шт.	06.06.2079	27.04.2003 0:11...
Абракадабра	шт.	06.06.2079	27.04.2003 0:11...
9sdasd	шт.	06.06.2079	17.01.2002 12:5...

Изменено пользователем "sa" (Офис 1) со станции "NOTBOOK-HP-SRV". Находится в группе "_Нераспреде.

27.04.2003 в 00:08:31 изменено пользователем "sa" (Офис 1) со ста
Критерий группировки: "Неизвестный"

Запись № 1 из 6

Просмотр

Что использовать в качестве атрибутов - поля или свойства?

ПОЛЯ		СВОЙСТВА	
Плюсы	Минусы	Плюсы	Минусы
<ul style="list-style-type: none"> • Информация занимает меньше места на диске • Кроме фильтров по ним можно задавать визуальную сортировку • К ним проще адресоваться из отчетов 	<ul style="list-style-type: none"> • Невозможно в разных частях дерева иметь разный набор полей • Большое количество полей увеличивает объем информации, возвращаемый сервером и замедляет его вывод • В табличной части при большом количестве полей мало видно информации об одной записи • Может быть только одно default-значение • Необходимость задания их перечня в «конструкторе» • Большой объем сохраняемой в истории информации 	<ul style="list-style-type: none"> • Разный состав атрибутов у записей в разных частях дерева • Уменьшение порций информации, возвращаемых по запросам сервером • Вертикальная визуализация увеличивает обозримый объем информации об одной записи • При грамотной организации иерархии за счет множественности default-значений можно ускорить ввод информации • Возможность вносить пользователями изменения в перечень атрибутов «на ходу» без запуска «конструктора» • При модификации одного атрибута в историю не записывается копия всей записи 	<ul style="list-style-type: none"> • Для каждой записи сохраняется не только значение, но и ссылка на сам атрибут – больше места требует на диске • Неудобно задавать сортировку по этому атрибуту • Сложности со ссылками из отчетам при разложении на аналитические уровни

Поля следует использовать для небольшого количества главных атрибутов, имеющих смысл для всех записей справочника (таких как, «наименование»). А также атрибутов, по которым требуется часто выполнять упорядочивание записей. Все остальные атрибуты лучше представлять в виде свойств.

Управление структурой иерархического справочника

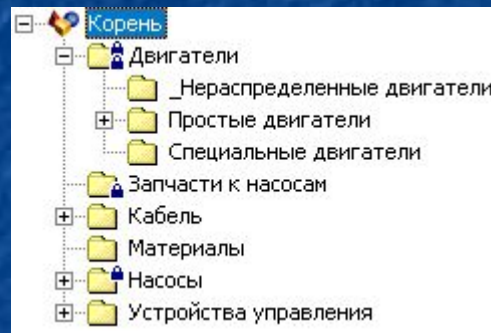
- Критерии группировки

Критерий группировки: "Группы ТМЦ по принципам организации учета и сбыта."

- Блокировка записей и групп

- Права на модификацию перечня свойств

- Правила подчинения записей в дереве

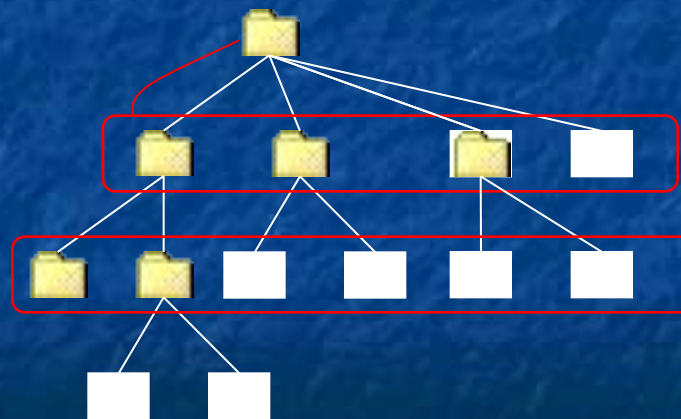


На одном уровне
элементы
разного
типа.

МОЖНО!

В подчинении у
одной группы
элементы
разного
типа!

НЕЛЬЗЯ!



Проекция на справочник

Номенклатура

Корень

- Двигатели
 - Нераспределенные двигатели
 - Простые двигатели
 - Специальные двигатели
- Запчасти к насосам
- Кабель
- Материалы
- Насосы
- Устройства управления

Высота
Высота оси
Длина
Длина статора
Климатическое исп...
Масса
Монтажное исполне
Мощность двигател
Напряжение питани
Обороты двигателя
Примечание для дв.
Производитель дви
Серия двигателей
Скольжение

Критерий группировки:

Наименование	Б...
Двиг. 90,0 квт	шт.
Двиг. 800,0 квт	шт.
Двиг. 75,0 квт	шт.
Двиг. 7,5 квт	шт.
Двиг. 630,0 квт	шт.
Двиг. 55,0 квт	шт.
Двиг. 500,0 квт	шт.
Двиг. 5,5 квт	шт.
Двиг. 45,0 квт	шт.
Двиг. 400,0 квт	шт.
Двиг. 40,0 квт	шт.
Двиг. 4,0 квт	шт.
Двиг. 37,0 квт	шт.
Двиг. 315,0 квт	шт.
Двиг. 30,0 квт	шт.
Двиг. 3,0 квт	шт.
Двиг. 250,0 квт	шт.
Двиг. 22,0 квт	шт.

□ В метаданных проекции указывается справочник, который является образующим для проекции. Эта часть дерева необходимо

представить в табличном виде. Проекция реализуется на VIEW и «системной» UDF выборки поддерева. Причем свойства должны

выглядеть как поля таблицы. Скрипты VIEW и Instead-триггеров делаются вручную, а не формируются автоматически. В Instead-триггерах реализуется логика, по которой добавляемые в проекцию записи сами распределяются в нужные ветви дерева.

□ Проекция, основанная на выборке поддерева, работает медленно. Модификация или добавление одной записи может привести к невидимому добавлению в дерево «виноградных гроздей» групп.

Двигатели (проекция на номенклатуру)

Наименование двигателя	Мощность ...	Обороты	Серия...	Высо...	Питающее ...	Монт...	I...	Взр...
NewTest	7.50							
Test2	45.00	789						
Test3	45.00	789						
Test4	45.00	789						
Test5_1__	45.00	789						
Абракадабра								
A4 L IM2031IP02 (7,5 / 3000)	7.50	3000	A4	26 L	220/380 В, 50Гц	IM 2031	IP02	Общепро...

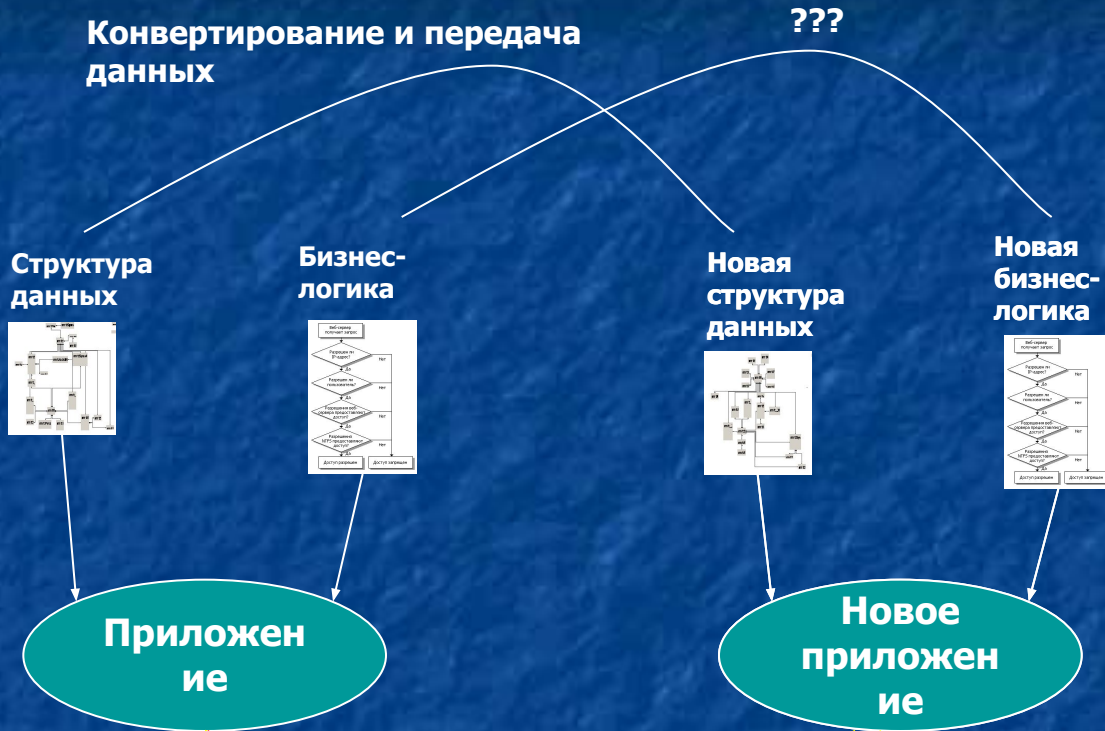
Запись № 1 из 7

Просмотр

Метаданные

- Структура метаданных строится на рассмотренном выше механизме справочников.
- Систематизируется и упрощается поддержка журнализации (отслеживание модификации информации по оси календарного времени).
- Автоматически решаются проблемы с конфликтами репликации и ее настройкой. В том числе структуры данных приложения.
- По оси учетного времени каждый объект структуры данных (например, документ) может иметь множество редакций, действующих на различных отрезках учетного времени.
- Упрощается модификация структуры данных и бизнес-логики при изменении работающих в прикладной области бизнес-процессов

Типичная схема модификация приложения



- Проблемы обработки данных в соответствии со старой бизнес-логикой
- Возможная утеря части старых данных из-за их несовместимости с новой бизнес-логикой

Проблемы с преемственностью

Проблемы с преемственностью

Большие проблемы с преемственностью

Предлагаемая схема модификация приложения



- Можно вносить изменения в бизнес-логику и в структуру данных во время работы пользователей
- Новая версия приложения 100% совместима со старой

Документы

Накладная №		123	от	01/06/200	
Отправитель	ЗАО «Азот»				
Получатель	ЗАО НПП «Карат»				
№ п/п	Наименование	Кол-во	Цена, руб	Сумма, руб	В т.ч. НДС, руб
1	Кран-букса	2	600.00	1200.00	200.00
2	Флагшток	3	1200.00	3600.00	600.00
ИТОГО				4800.00	800.00

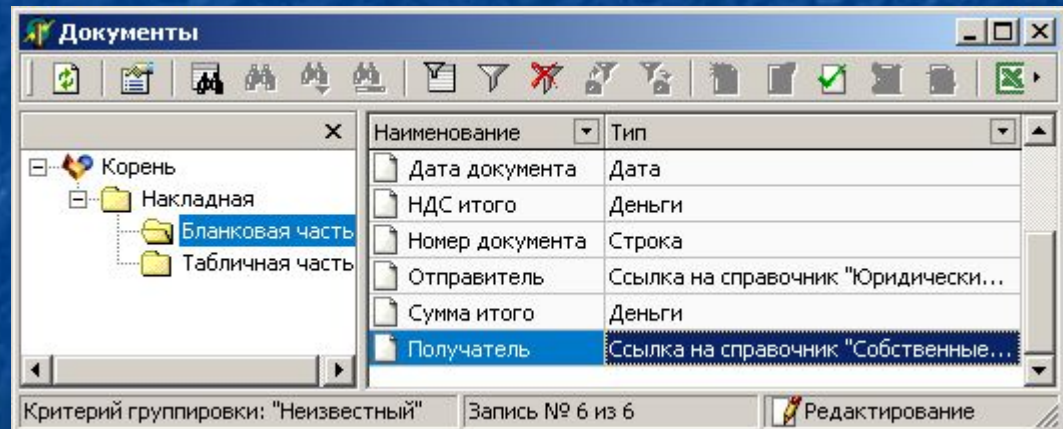
Поля
бланковой
части

- Поля документа могут ссылаться на справочники

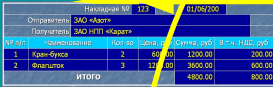


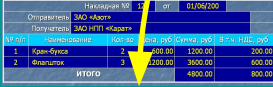
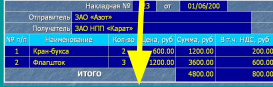

- Структура документа может быть описана содержимым иерархического справочника

- На структуру документа распространяется версия, приемы репликации и т.д.

Поля
табличной
части



Формирование одних документов на базе других

Накладные	Счета	Счета-фактуры	Спецификации
			
			

- Автоматическое формирование документов одних на базе других настраивается заранее заданием соответствия полей источника и приемника.
- Документ можно сформировать на базе другого даже если их содержимое различается – чтобы только подправить его, уменьшив объем ручного ввода
- Информация о первоисточниках и документах-приемниках сохраняется. При изменении содержимого первоисточника будет предложено автоматически переформировать основанные на нем документы-приемники по всем цепочкам.
- Если документ был вручную подправлен, система предупредит.

Сессия, черновик и беловик

- Два вида документов – моментальный и протяженный. Моментальный не имеет версий на оси учетного времени. Это основной вид документа для учетных задач (накладные, счета и т.п.). Он двигает регистры.
- Протяженный документ напрямую не двигает регистры. Он является группирующим для множества моментальных. Пример протяженного документа – договор.
- По оси учетного времени поддерживается версионность структуры и бизнес-логики регистрации экземпляра.
- Сессия ввода/редактирования документа может длиться несколько дней с промежуточными сохранениями.
- Черновик – редактируемая заготовка экземпляра документа. Процесс его модификации не протоколируется. Изменение его содержимого не влияет на журналы документов и регистры.
- Беловик получается из черновика установкой статуса. В этот момент протоколируется все содержимое документа разом. Запускается бизнес-логика отражения его содержимого в регистрах. Документ отображается в журналах документов.
- Для изменения документа сначала ему присваивается статус черновика – при этом из системы удаляется вся информация, которая в ней ранее появилась в результате регистрации документа.
- По оси календарного времени регистрируются только моменты изменения статуса черновик/беловик с протоколированием содержимого документа на этот момент. Реплицируются моменты изменения статуса.

Журналы документов

- Три вида журналов – глобальный, локальный и настраиваемый.
- Глобальный журнал – один. В нем регистрируются все документы. Отражаются обязательные для всех документов поля – учетная дата, номер (а также автор, дата модификации, и тип документа).
- Локальный журнал создается автоматически по одному на один вид документа. Он содержит значения всех полей бланковой части документов данного вида.
- Настраиваемый журнал создается пользователем для удобства поиска документов разного вида. Задается перечень документов, которые должны в нем регистрироваться и задается перечень полей журнала. А также сопоставление полей документов различных видов полям журнала.
- Информация журналов не реплицируется. Журналы формируются триггерами по появлению в системе информации о новых документах.

Регистры

Регистр движения товаров

Склад	Номенклатура	Количество	Сумма	Приход / расход	Основание

Остатки

Аналитические
разрезы

Ресурсы

Системные
поля

Справочник
складов

Справочник
номенклатуры

Экземпляры
документов

t
учетное

- Обороты (приход за период, расход за период)
- Остатки на определенный момент

Взаимодействие документов и регистров

Платежное поручение от нас по счету ЗАО «Азот» на сумму 4000 рублей

Накладная на получение нами товара от ЗАО «Азот» на сумму 5000 рублей

Платежное поручение на доплату на сумму 1000 рублей

Платежное поручение № 11/01010			
Страница: 30 из 30			
Счет ЗАО «Азот»			
1	Счет	2	4000,00
2	Счет	3	4000,00
Итого			4000,00

Накладная № 11/01010			
Страница: 30 из 30			
Счет ЗАО «Азот»			
1	Счет	2	5000,00
2	Счет	3	5000,00
Итого			5000,00

Платежное поручение № 11/01010			
Страница: 30 из 30			
Счет ЗАО «Азот»			
1	Счет	2	1000,00
2	Счет	3	1000,00
Итого			1000,00

Регистры

10'000

6'000

6'000

5'000

-
Деньги в банке

Задолженность ЗАО «Азот» перед нами
4'000

Задолженность наша перед ЗАО «Азот»
1'000

0

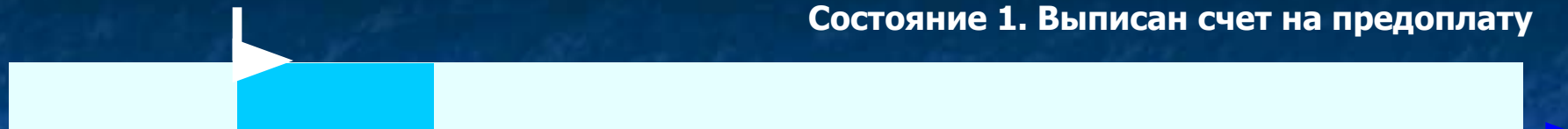
+
Взаиморасчеты

Товар1 1шт x 2'000 = 2'000
Товар2 2шт x 1'500 = 3'000

+
Товар на складе

Концепция детерминированности учетного времени

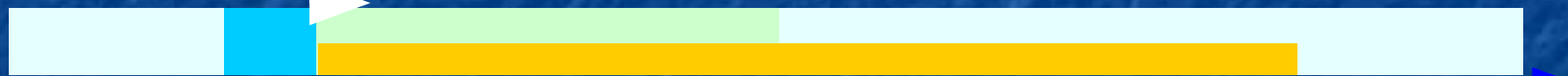
Состояние 1. Выписан счет на предоплату



Постановка товара в мягкий резерв до момента прихода денег.

Оплата в срок не поступила – снятие товара с резерва. Перевод в свободный остаток.

Состояние 2. Поступление денег по счету



Поступили деньги – перевод товара в жесткий резерв. Регистрация задолженности.

Истечение срока жесткого резервирования товара. Перевод в свободный остаток.

Списание задолженности в связи с истечением срока исковой давности.

Состояние 3. Товар отгружен



Поступили деньги – перевод товара в долгосрочный резерв. Регистрация задолженности.

Списание товара. Погашение задолженности.

 Товар в свободном остатке

 Товар в долгосрочном резерве

 Товар в мягком резерве

 Задолженность перед покупателем

Главная проблема концепции

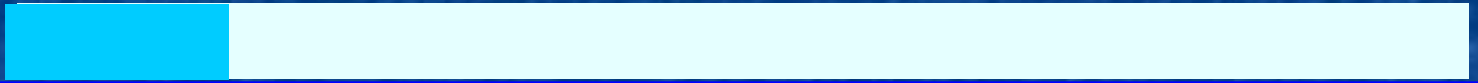
Состояние 1. Выписан счет на предоплату



Постановка товара в мягкий резерв до момента прихода денег.

Оплата в срок не поступила – снятие товара с резерва. Перевод в свободный остаток.

Состояние 2. Удален или изменен ранее введенный документ на приход товара



Возникновение товара из вакуума

НЕОБХОДИМО ПЕРЕПРОВЕДЕНИЕ ДОКУМЕНТОВ, ОТНОСЯЩИХСЯ К БОЛЕЕ ПОЗДНИМ ПЕРИОДАМ!!!

Чтобы не делать перепроведение **ВСЕХ** документов более поздних периодов, необходимо хранить граф зависимостей документы-показатели-документы

Решение проблемы графа зависимостей

документы регистры показатели документы регистры показатели



Особенности

- «Заккрытие» периодов производится не физически, а логически — только с целью защиты информации от случайной модификации.
- Допускается «заккрытие» только части информации, а не всей за нужный период.
- Регистры НЕ реплицируются.
- Содержимое регистров формируется автоматически на базе реплицируемых документов.
- Регистры содержат картину данных по всей оси учетного времени. Можно получать любые срезы и даже планы будущего.
- Бизнес-логика реализуется UML-схемами.
- Сами схемы являются частью реплицируемых данных.
- Репликация бизнес-логики легко реализуется в рамках общей концепции.
- Никакие операции не требуют прерывания работы пользователей (изменение структуры данных, изменение бизнес-логики, «заккрытие» периодов)

Резюме

Предлагаемая концепция позволяет:

- Получить приложения, гибко настраиваемые под места внедрения.
- Существенно упростить сопровождение ранее внедренного приложения.
- Решить проблемы с низкой оперативностью OLAP-сервисов.
- Переложить с DBA на специалистов прикладной области функции управления репликацией и устранением конфликтов.
- Предоставить специалистам прикладной области удобные средства контроля и мониторинга за вводом информации. Снять эти функции с DBA.
- Решить проблемы с любыми произвольными и непредсказуемыми видоизменениями учитываемых объектов.

Просмотры и отчеты

- Стандартные просмотры позволяют без усилий получать ответы на ~95% вопросов. По справочникам, журналам документов, документам, регистрам.
- Стандартные просмотры по регистрам обладают функциями для анализа.
- Документы могут иметь несколько печатных форм, которые задаются шаблонами и включаются в состав структуры документа.
- Генератор отчетов произвольной формы.
- Отчеты могут быть периодическими. Их состав и доступность задается положением заданных рисков по оси учетного времени относительно базовых рисков (начало года, квартала, месяца и т.п.).

Ссылки

http://akop.ru/personal/1234?PARENT_RUBR=akop_art_it А.Х.Акопянц
«Автоматизация хаоса»

http://akop.ru/personal/1235?PARENT_RUBR=akop_art_it А.Х.Акопянц
«Автоматизация «от данных» и информационное моделирование»

http://www.msaccess.ru/Raznoe_About.html А.А.Гордиенко «Некоторые
соображения по поводу разработки крупных проектов на Access-97»

http://www.msaccess.ru/Raznoe_BigPrgComment5.html «Переписка Андрея
Гордиенко и Ирины Николаевой»

Контактная информация

Форум www.sql.ru

Форум MAUG

...

Ник «Garya»

E-mail: agordienko@hms.ru