

Руководство по StealthTrace

Версия 5.1

(С) Евгений Каратаев
mailto:support@minimdb.com
<http://www.minimdb.com>

27 октября 2011 г.

Содержание

1	Архитектура и компоненты StealthTrace	2
2	Инсталляция StealthTrace	3
3	Создание базы данных и StealthTrace Storage	4
4	Индексы StealthTrace	5
5	Описание куба	6
6	Загрузка данных	8
7	Запрос StealthTrace	10
8	Ассоциативные правила	11
9	Приложение 1. Форматы типов StealthTrace	13
10	Приложение 2. Демонстрационный пример Home Money	13
11	Приложение 3. Демонстрационный пример Market Basket Analysis	15

1 Архитектура и компоненты StealthTrace

StealthTrace это программный пакет хранения данных и построения запросов пользователем с выводом результатов запроса в виде крестаблицы.

Пакет состоит из серверных модулей и программ клиентской части. Серверные модули реализуют механизмы хранения данных, специальной индексации, выполнения запросов и выдачи их результатов клиентским программам.

Клиентские программы состоят из двух программ - StealthTrace как средство задания запросов и показа их результатов в виде отчета и StealthTrace Meta как средство для определения кубов StealthTrace и генерации типового кода добавления данных для каждого из сконфигурованных кубов.

На серверной стороне поддерживается так называемый StealthTrace Storage, включающий в себя исполняющие рутины (поставляются в компилированном формате) и специальные структуры данных, создаваемые автоматически.

В одной базе данных MiniM может присутствовать один StealthTrace Storage. В одном StealthTrace Storage может присутствовать несколько гиперкубов и запросов.

Программа StealthTrace предназначена для конечных пользователей для определения выборок данных и способов их показа в отчете. Программа StealthTrace Meta предназначена для разработчиков и администраторов эксплуатируемых баз данных и хранилищ данных.

Программа StealthTrace может полученный отчет как экспортировать в переносимый форматы файлов данных, так и вставить данные в клипборд для обмена с другими программами.

Кроме того, отчет может быть распечатан с сохранением раскладки по страницам как по вертикали так и по горизонтали. Получаемые крестаблицы во многих случаях могут занимать как несколько страниц по вертикали так и несколько страниц по горизонтали и распечатанные страницы могут быть склеены для получения большого многостраничного отчета.

Как серверная так и клиентская части пакета StealthTrace содержат специальные алгоритмы и структуры данных для работы с большими объемами как исходных данных хранилища так и больших отчетов.

Клиентская и серверная части StealthTrace могут работать как на одном компьютере так и на разных. Клиентские программы соединяются с сервером по специальному протоколу поверх tcp/ip. Протокол обмена входит в комплект как StealthTrace так и сервера MiniM. Администраторам необходимо лишь обеспечить работу tcp/ip.

StealthTrace Storage - совокупность программ на стороне сервера и данных в специальных форматах. В рамках StealthTrace Storage выполняется хранение всех остальных данных StealthTrace.

StealthTrace Dimension - условная ось условного пространства, в котором определяется положение факта куба.

StealthTrace Measure - собственная характеристика факта, над которой может производиться вычисление для построения отчета.

StealthTrace Fact - одна запись, объединяющая значения размерностей (Dimension) и показателей (Measure).

StealthTrace Cube - объединение однородных записей. Название Cube условно, означает что обычно положение одного факта в пространстве размерностей задается несколькими размерностями. Также часто используется термин HiperCube.

StealthTrace Query - формулирование запроса к кубу данных, включая условия выборки данных, вычисляемой функции над показателями и условия показа результата.

2 Инсталляция StealthTrace

Для работы StealthTrace необходимо иметь установленный и настроенный сервер MiniM Database Server версии 1.10 или старше и установить пакет StealthTrace на клиентских местах для которых он необходим.

Кроме клиентских программ инсталлятор StealthTrace устанавливает как эту документацию так и скомпилированные серверные рутины.

После установки клиентских программ разработчику или администратору нужно запустить программу StealthTrace Meta и настроить соединение с базой данных MiniM, в которой должен быть создан и проинициализирован StealthTrace Storage. Программа StealthTrace Meta автоматически проверяет готовность базы данных и при необходимости создания Storage спрашивает нужно ли в этой базе создать StealthTrace Storage.

При подтверждении программа StealthTrace Meta автоматически производит импорт в базу данных компилированных рутин серверной части и подготовку начальных структур данных, с которыми может впоследствии работать как сама для определения кубов так и программа StealthTrace для построения и выполнения запросов.

StealthTrace Storage может быть создан в любом числе баз данных MiniM.

Полный комплект клиентских программ требуется лишь для администраторов и разработчиков. На компьютерах конечных пользователей для избежания случайного редактирования определения кубов рекомендуется после инсталляции удалить программу stmeta.exe.

3 Создание базы данных и StealthTrace Storage

При создании базы данных и StealthTrace Storage администраторам нужно понимать технические особенности работы хранилищ данных и учитывать их при конфигурировании если для решаемых задач эти особенности могут быть важны.

К ним относятся:

1. Работа с большими объемами данных
2. Данные в хранилище попадают как вторичные из первичных источников данных
3. Данные в хранилище добавляются эпизодически
4. Хранилища работают в основном в режиме чтения

Исходя из особенностей применения хранилищ данных может быть рекомендовано следующее:

1. Использование отдельной базы данных MiniM для хранилища
2. Использование увеличенного кеша глобалов на сервере MiniM
3. Отключение журналирования базы данных для хранилища
4. Использование отдельного компьютера или отдельной инсталляции MiniM

В любом случае администратор может использовать любую имеющуюся базу данных MiniM, пакет программ StealthTrace в любом случае будет выполнять требуемые действия.

В одном StealthTrace Storage может размещаться несколько кубов. При этом кубы могут использовать совпадающие размерности. В текущей версии StealthTrace нет возможности строить запросы, использующие одновременно два или более кубов, но в последующих версиях такая возможность запланирована, поэтому к рекомендациям по планированию баз данных можно также добавить рекомендацию размещать те кубы, которые имеют совпадающие размерности, в одном хранилище.

4 Индексы StealthTrace

Пакет StealthTrace использует классические bitmap индексы. Для определения того, какие записи имеют заданное значение, для этого значения поддерживается специальная последовательность нулей и единиц. Наличие единицы в этой последовательности означает что запись с этим номером присутствует.

Для перечисления всех записей имеющих заданное значение берется последовательность записанных при индексации нулей и единиц и для каждой найденной в ней единицы выдается положение этой единицы в общей последовательности. Это и будет номер записи имеющей заданное значение.

Для выполнения операций И и ИЛИ пакет StealthTrace выполняет побитовые операции над соответствующими битовыми последовательностями. Полученная в результате последовательность битов означает номера записей, попадающих под указанное условие. Сами операции побитовых И и ИЛИ выполняются весьма быстро, на низком уровне, поэтому получение результата выборки из большого массива данных происходит намного быстрее чем при использовании традиционных индексов.

Bitmap индексы по своим характеристикам намного практичнее для операций над большим набором объектов при пакетных операциях, а традиционные намного практичнее для операций с единичными объектами.

Пакет StealthTrace логически выполняет вырезание областей данных из общего куба по правилу: условия по одной оси (Dimension) объединяются операцией ИЛИ, условия по разным осям объединяются операцией

И. Таким образом, чтобы можно было задать попадание в отчет одновременно двух или более областей по заданной оси.

Формирование внутренних индексных структур и их актуальную поддержку пакет StealthTrace выполняет автоматически.

5 Описание куба

StealthTrace Cube состоит из набора записей, или фактов. Куб характеризуется совокупностью осей (Dimensions) и показателей (Measures) фактов.

Ось куба представляет собой условное пространство в котором может условно находиться факт и отличаться от других фактов или совпадать по своему положению с другими фактами.

Для понимания условной оси опишем краткую справку из математики. Положим, у нас есть несколько объектов. До тех пор пока мы не определили над ними никаких операций, это просто некая условная куча.

Если каждый из объектов мы можем пронумеровать некими условными номерами или именами или еще каким-то способом, то после добавления такой операции этот набор объектов уже становится множеством объектов.

Если к нашим условным номерам или именам, которые мы приписали этим объектам, мы можем добавить правило их упорядочивания и указать какой номер или имя должны находиться условно левее а какой правее, то эти номера уже образуют пространство, а множество становится упорядоченным.

Если мы к нашим условным номерам или именам, приписанным объектам, добавим правило определения меры для произвольно взятой пары номеров или имен, или расстояния между двумя объектами, или другими словами операцию сложения, то мы уже вводим геометрию на этом пространстве. Традиционным для операции сложения инвариантом является понятие нулевой меры или нулевого объекта.

Если мы к упорядоченному пространству с мерой и операцией сложения добавим операцию умножения двух произвольно взятых номеров или имен или мер, то получим алгебру на этом множестве номеров или имен. Традиционным для операции умножения инвариантом является понятие единичного объекта.

Таким образом, отличие множества от пространства и геометрии от алгебры условно и определяется лишь тем, какие операции мы определили, и, формально, алгебра это лишь расширенная геометрия.

Для использования пакета StealthTrace не требуется введения алгебры и даже геометрии, достаточно просто упорядоченного множества. Для значений осей используется лишь операция больше-меньше и при формулировании запроса к кубу StealthTrace достаточно понимать, что означает больше-меньше в отношении значений размерностей.

В пакете StealthTrace реально используется лишь две операции больше-меньше - это определенность сортировки строк и сортировки чисел. Числа сортируются по возрастанию арифметической величины, а строки по алфавиту.

Факт или одна запись куба может находиться одновременно в нескольких пространствах, например событие подключения компьютера к провайдеру интернета может быть задано в пространствах Город, Улица, Номер дома, Дата, Время, Тип подключения, Номер бригады.

Показатель (Measure) факта это собственная характеристика события, с которой имеют смысл арифметические операции, например оценка качества подключения к интернету по какой-либо шкале для последующего вычисления среднего качества.

В случае если требуется выборка данных для отчета также и по значениям показателей, то необходимо определить также соответствующую размерность и при заполнении куба данными вносить копию значения данных и в показатель и в размерность.

Традиционно для хранилищ данных к размерностям относят величины, которые можно условно перенумеровать и число их различных значений не так велико. А значения показателей могут принимать произвольные, в том числе дробные значения, и число различных значений показателей обычно очень велико, включая случай что каждый из фактов имеет собственное значение показателя.

Описание куба выполняется программой StealthTrace Meta. К традиционной последовательности проектирования можно отнести:

1. Разработчик создает куб и определяет входящие в него размерности и показатели.
2. Разработчик использует сгенерированный StealthTrace Meta типовый код добавления одной записи в своем коде добавления данных.

3. Разработчик экспортирует определение куба из StealthTrace Meta и набор рутин добавления данных и передает полный пакет администратору сервера хранилища.
4. Администратор выполняет запуск StealthTrace Meta и создает с его помощью при необходимости начальное состояние StealthTrace Storage.
5. Администратор выполняет импорт определения куба с помощью StealthTrace Meta и рутин добавления данных.
6. Администратор выполняет определенные разработчиком действия по планированию периодичности добавления данных.

В многих случаях разработчик и администратор могут быть и одним и тем же лицом, а сервер разработки и эксплуатации может быть одним и тем же сервером.

6 Загрузка данных

По своему назначению хранилища данных это периодически пополняемые наборы данных. Со временем в хранилище накапливается все большее количество данных, представляющее все большую ценность для анализа.

Заполнение данными кубов StealthTrace выполняется по одной записи, с передачей системе денормализованных значений. StealthTrace Storage автоматически выполняет необходимые преобразования данных и использует внутренние номера по необходимости.

Разработчик для получения рутины заполнения данными должен запустить программу StealthTrace Meta, открыть интересующий куб и вызвав пункт меню Meta | Code to Add Data взять образец кода на языке MUMPS для добавления записи.

Код добавления представляет собой передачу в StealthTrace Storage информации о кубе, о значениях размерностей и показателях в порядке определенном внутренней структурой куба.

StealthTrace Storage при импорте определения куба на другом сервере поддерживает тот же порядок значений размерностей и показателей, чтобы разработанный код добавления мог работать и на другом сервере.

В случае если куб был создан не импортом его определения, а самостоятельно, то такой порядок уже не гарантируется, хотя и может совпадать.

Разработчик должен в своем коде добавления сохранить указанный порядок. В случае редактирования определения куба и изменения его размерностей или показателей нужно повторно сгенерировать такой образец и проверить правильность следования значений.

Кроме того, разработчик должен определить источник данных для добавления и способ определения, какие данные следует добавить при следующем периодическом обновлении. Это может быть произвольный источник данных - внешний файл в специальном формате, специальная программа которая обращается к серверу MiniM, источник данных ODBC, или данные в другой базе данных MiniM.

Например, это может быть импорт файла целиком, или обращение к данным прикладной системы с запросом данных позже даты и времени последнего обращения.

В каких-то случаях данные для добавления одного факта могут собираться и из нескольких источников данных, например информация, актуальная для анализа, может просто отсутствовать в обычной учетной системе и находиться в сторонних справочниках. Например, если для анализа актуален условный класс дома, то в первичных учетных данных его может просто не содержаться и по адресу дома разработчик должен определить класс дома, обратившись к дополнительным справочникам.

Разработчик должен предпринять меры к тому, чтобы разработанные им рутинные операции в идеальном случае могли быть экспортированы, импортированы и запущены на сервере эксплуатации. Кроме файла определения StealthTrace Cube и рутин добавления данных разработчик должен при необходимости предоставить инструкции по настройке дополнительных шлюзов данных либо следовать инструкциям определенным администратором сервера эксплуатации.

Кроме того, разработчик должен выработать правила нормализации значений размерностей и показателей. StealthTrace Storage будет различать различные значения как разные. Как пример, разработчик должен придерживаться правил именования, например названия городов должны быть строчными буквами, первая буква заглавная, лидирующие и завершающие пробелы убраны, если есть пробелы внутри названия, то они должны быть только из одного пробела, если есть дефис в написании то перед и после него не должно быть пробелов. Общее правило таково, что если для анализа некие два значения означают одно и то же, то они

должны быть одинаковыми. Или, если какое-то значение размерности имеет дробную величину, то должно использоваться одинаковое число десятичных цифр после десятичной точки.

Если данные для загрузки получаются гарантированно из одного и того же источника и выдаваемые им значения уже гарантированно нормализованы, то часть работы по нормализации значений разработчик может опустить и довериться источнику данных.

Общий результат работы разработчика состоит в пакете файлов, который нужно передать администратору сервера эксплуатации, которые он может импортировать и запустить. Рекомендуется избегать неких магических глобалов, магических идентификаторов, или системных настроек, значения которых на другом сервере могут иметь иное значение или отсутствовать, из-за чего данные могут прийти в некорректное с точки зрения анализа состояние.

7 Запрос StealthTrace

Запрос StealthTrace формулируется и выполняется программой StealthTrace (st.exe).

Для запроса выбирается один из имеющихся на сервере в StealthTrace Storage кубов. Для куба выбираются оси по которым необходимо показать отчет. Оси выбираются для левого набора и для верхнего. Например, если для левых выбрана ось Город, а для верхних оси Год и Месяц, то отчет будет показывать Города по годам и месяцам.

Если не указаны условия выборки данных из куба, то отчет строится по всем данным которые есть в кубе. Можно сузить область куба указав для каких именно данных нужен отчет. Условия запроса определяются для осей куба. При выборе условий запроса используется правило: условия по одной оси объединяются правилом ИЛИ, условия по разным осям правилом И.

Оси задают как показать данные, условия задают какие данные показать, а выбранные показатели запроса задают что показать. Для показа нужно выбрать один или несколько показателей куба и указать какую функцию к ним применить.

Если для осей указана опция строить подитоги, то подитоги будут строиться применяя ту же функцию, что задана для показателей. Например, если выбрана функция Сумма, то в подитогах будет рассчитываться

также Сумма, если выбрано Среднее, то в подитоге также будет среднее по соответствующей группе данных, например если выбраны оси Год и Месяц, то подитоги строятся по всем месяцам входящим в год.

В отчете клетки подсвечиваются разными цветами для показа что эти клетки означают. Цвета могут быть настроены на усмотрение пользователя.

Формулированный запрос StealthTrace автоматически сохраняется в базе данных и может быть впоследствии открыт снова, изменен или по нему выполнен отчет. Отчет на сервере не хранится и строится каждый раз снова по текущим данным.

Данные отчета могут быть экспортированы в распространенные форматы файлов для передачи в другие программы или напечатаны. Кроме того, можно экспортировать отчет в формате HTML для последующего его просмотра браузером или в формат PDF.

В окне показа отчета можно выбрать нужное число десятичных знаков после десятичной точки (запятой) и привести вид данных к нужному.

8 Ассоциативные правила

Программа StealthTrace Association Rules выполняет анализ данных и выявляет ассоциативные правила.

Для анализа выбирается один из имеющихся на сервере кубов StealthTrace и задаются параметры поиска правил - предпосылки, следствия, условия выборки данных для анализа, параметры оценок и соединений фактов.

Если не указаны условия выборки, то программа анализирует все факты куба, иначе проводит анализ только тех фактов которые удовлетворяют заданным условиям. Условия поиска определяются для осей куба. При выборе условий поиска используется правило: условия по одной оси объединяются правилом ИЛИ, условия по разным осям правилом И.

Программа обнаруживает правила по перечисленным предпосылкам. Необходимо задать по крайней мере одну ось. В случае если заданы две или более оси, то программа ищет правила для их комбинаций. В качестве следствий программа использует указанные для следствий оси, их также можно указать от одной и более.

Соединение фактов или перечень осей для соединения используется для обнаружения взаимосвязей между различными фактами, объединенными по указанным в соединении осям. Если соединение не указано,

то программа может анализировать только взаимосвязи между раными осями фактов. Если указаны, то может анализировать и одинаковые оси для разных фактов, для которых значения соединения совпадают.

К типичным задачам поиска ассоциативных правил относят две задачи: поиск правил ассоциации между разными осями и поиск правил ассоциации по одной оси для разных фактов с указанным соединением.

Например, к первой задаче относится поиск правила какие товары предпочтительнее продаются по дням недели или по времени суток. В этом случае в качестве предпосылок задаются дни недели или часы продажи, а в качестве следствий товар.

К второй задаче относится классическая задача анализа покупательской корзины (Market Basket Analysis) с поиском правил с каким товаром чаще приобретается другой товар. В этом случае в качестве предпосылок и следствий указывается ось товар, а в качестве соединения ось номер чека. Для того, чтобы такой анализ мог быть проведен, куб StealthTrace должен быть заранее спроектирован так, чтобы он содержал необходимые для соединения оси.

В демонстрационные примеры StealthTrace Association Rules входит пример MarketBasket с демонстрацией использования куба для поиска ассоциативных правил.

Важными для понимания характеристиками анализа являются значения поддержки правил и достоверности правил (Support и Confidence). Поддержкой правил называется процент доли предпосылок от всех встреченных предпосылок. Если этот процент велик, то комбинация предпосылок встречается часто. И наоборот, если мал, то таких предпосылок в кубе мало. Достоверностью правила называется процент доли комбинации предпосылки и следствия для всех таких предпосылок. Если этот процент велик то правило подтверждается часто, и наоборот, если мал то подтверждается редко.

В понимании анализа значение поддержки (Support) характеризует насколько часто встречается, или насколько очевидно это правило. Значение достоверности (Confidence) характеризует насколько это правило сильно и устойчиво. Поэтому программа имеет фильтр ограничения долей поддержки и достоверности сверху и снизу, чтобы отсеять малоинтересные правила с малыми значениями поддержки и достоверности и очевидные правила с большими значениями поддержки и достоверности.

Сформулированные параметры поиска правил StealthTrace автоматически сохраняются в базе данных и могут быть впоследствии открыты

снова, изменены или по ним выполнен поиск. Отчет о правилах на сервере не хранится и строится каждый раз снова по текущим данным куба.

Данные отчета могут быть экспортированы в распространенные форматы файлов для передачи в другие программы или напечатаны. Кроме того, можно экспортировать отчет в формате HTML для последующего его просмотра браузером или в формат PDF.

Для поиска ассоциативных правил используется отдельная программа `star.exe` с интерфейсом, специально ориентированным на задачу поиска правил.

9 Приложение 1. Форматы типов StealthTrace

Для записи данных в хранилище StealthTrace значения должны соответствовать следующим соглашениям:

1. Строка - строка как есть.
2. Число - число в нотации MUMPS, с возможной дробной частью.
3. Дата - значение даты в формате переменной \$HOROLOG.
4. День недели - число от 1 (Понедельник) до 7 (Воскресение).
5. Неделя - число от 1 (первая неделя года).
6. Месяц - число от 1 (Январь) до 12 (Декабрь).
7. Квартал - число от 1 (первый квартал) до 4 (четвертый квартал).
8. Год - номер года. Рекомендуется использовать четырехзначную запись года.
9. Денежный - число в нотации MUMPS, с двумя опциональными знаками после десятичной точки.

10 Приложение 2. Демонстрационный пример Home Money

Пакет StealthTrace содержит демонстрационный пример Home Money. Пример условно описывает домашние финансы некоторой семьи. Пример

находится в подкаталоге /demo и состоит из двух файлов HomeMoney.stm с определением куба и HOMEMONEY.rou с рутинной генерации случайных данных для него.

Пример описывает расходы членов семьи, для кадного факта расхода отмечаются его характеристики - Account (Счет расходов), Person (Член семьи), Year (Год), Month (Месяц), Day (День), Type (Вид оплаты) и показатели Bill amount (сумма счета оплаты), Discount (сумма скидки на оплату) и Payment (итоговая сумма платежа).

Для запуска примера нужно выполнить следующие шаги:

1. Выбрать, в какой базе данных нужно создать хранилище StealthTrace и куб в нем.
2. Запустить программу StealthTrace Meta (stmata.exe). При старте программа проверяет наличие в базе данных StealthTrace Storage. Если его там еще нет, то нужно подтвердить его создание.
3. Из меню Meta | Import Metadata импортировать файл HomeMoney.stm из подкаталога /demo.
4. Закрыть программу StealthTrace Meta.

После этого в базе данных создано StealthTrace Storage и определение куба Home Money. Для наполнения куба данными нужно выполнить следующие шаги:

1. Запустить MiniM Routine Editor или MiniM Control Center и в ту же базу данных импортировать файл HOMEMONEY.rou из подкаталога /demo, содержащий рутину HOMEMONEY.
2. Запустить консоль или телнет, перейти в ту же базу данных, и выполнить

```
d CreateRecords^HOMEMONEY ( )
```

После этого в куб Home Money добавлено 2000 записей со случайно выбранными значениями.

Далее можно использовать пользовательскую программу создания и выполнения запросов к кубу, например:

1. Запустить программу StealthTrace (st.exe)

2. Создать новый запрос (например, Home Query), указав используемый куб Home Money.
3. Далее можно указывать параметры запроса, например Left Dimensions: Person; Top Dimensions: Year, Month; Query Condition: Account = Transport; Measures: Payment (Summary).
4. Запустить построение отчета по запросу.

Изменяя параметры запроса, можно изучать данные куба Home Money в разных разрезах.

11 Приложение 3. Демонстрационный пример Market Basket Analysis

Пакет StealthTrace содержит демонстрационный пример Market Basket Analysis. Пример условно описывает информацию о продажах некоторого магазина. Пример находится в подкаталоге /demo и состоит из двух файлов MarketBasket.stm с определением куба и MBASKETAN.rou с рутинной записью данных для него.

Пример описывает сокращенный вариант информации о продажах, касающийся только данных для поиска правил. Куб включает оси Дата продажи, День недели продажи, Номер чека и Наименование товара. В качестве показателя используется только Количество продаж, и при генерации данных ему присваивается только значение 1.

Нужно отметить, что для поиска ассоциативных правил построение куба может отличаться от построений куба для других задач. В частности, для классического Market Basket Analysis необходимо значение номера чека. В случае если номера чеков в разных магазинах могут совпадать, нужно использовать для соединения комбинацию осей Магазин и Номер чека. Также, зачастую, анализ выполняется не по Наименованию товара, а по его некоторым образом определенному классу. Например, для продажи разных видов молока важно различать какое именно молоко, а для поиска правил все виды молока могут быть объединены в один класс Молоко. Или, для поиска ассоциаций времени продажи с объемом тары важно различать класс тары.

Для запуска примера нужно выполнить следующие шаги:

1. Выбрать, в какой базе данных нужно создать хранилище StealthTrace и куб в нем.
2. Запустить программу StealthTrace Meta (stmata.exe). При старте программа проверяет наличие в базе данных StealthTrace Storage. Если его там еще нет, то нужно подтвердить его создание.
3. Из меню Meta | Import Metadata импортировать файл MarketBasket.stm из подкаталога /demo.
4. Закрывать программу StealthTrace Meta.

После этого в базе данных создано StealthTrace Storage и определение куба Market Basket. Для наполнения куба данными нужно выполнить следующие шаги:

1. Запустить MiniM Routine Editor или MiniM Control Center и в ту же базу данных импортировать файл MBASKETAN.gou из подкаталога /demo, содержащий рутину MBASKETAN.
2. Запустить консоль или телнет, перейти в ту же базу данных, и выполнить

```
d ^MBASKETAN
```

После этого в куб Market Basket добавлены серия записей с условными примерами продаж.

Далее можно запускать программу поиска правил star.exe и выполнять поиск правил.

В качестве примера поиска правил ассоциации одного товара с другим выбрать в качестве предпосылок и следствий ось Selling Good, а в качестве соединения ось Selling Check.

При запуске анализа программа показывает список всех правил удовлетворяющих заданным условиям. Показ выполняется в двух видах - полный отчет и отчет вида Что-Если (What-If). При переключении отчета на вид Что-Если слева показывается список найденных предпосылок, справа список найденных следствий. При выборе предпосылки программа показывает справа найденные для этой предпосылки следствия. Колонки могут быть отсортированы по возрастанию или убыванию.

В частности, в демонстрационном примере показывается, что с беконом часто берут картофель, с печеньем молоко, а с пивом чипсы.