

Санкт-Петербургский Государственный Университет

Математико-механический факультет

Кафедра системного программирования

Выпускная дипломная работа

Карымова Антона Вячеславовича

на тему:

«Разработка шаблонов ORACLE APEX, ориентированных на создание
финансовых приложений»

Научный руководитель к. ф.-м. н. доц. Н.Г. Графеева

Рецензент к.ф.-м.н. И.Шаблинская

“Допустить к защите” д.ф.-м.н. проф. А.Н. Терехов

заведующий кафедрой

Санкт-Петербург

2012 г.

Saint-Petersburg State University
Mathematics and Mechanics Faculty
Software Engineering Department

Development of ORACLE APEX templates oriented to financial
applications

Graduate paper by

Anton Karymov
461 group

Scientific advisor Natalia Grafeeva

Reviewer Irina Shablinsky

Reviewer "Approved by" Head of Department Professor A. N. Terekhov

Saint-Petersburg

2012

Оглавление

Введение.....	2
Глава 1. Обзор графических основ трейдинга и используемых инструментариев.....	4
Основы финансового трейдинга.....	4
Обзор традиционных для трейдинга графиков.....	4
<i>График с двойной шкалой.....</i>	<i>5</i>
<i>Двухсекционный график с объемами и ценой в виде баров.....</i>	<i>5</i>
<i>Двухсекционный график с объемами и ценой в виде свечей.....</i>	<i>7</i>
<i>График с нарастающими итогами.....</i>	<i>10</i>
Описание используемых инструментариев.....	11
<i>Oracle APEX описание среды разработки.....</i>	<i>11</i>
<i>Графики Oracle APEX.....</i>	<i>15</i>
<i>Anychart.....</i>	<i>16</i>
Глава 2. Особенности реализации.....	17
<i>График с двойной шкалой.....</i>	<i>18</i>
<i>Двухсекционный график с объемами и ценой в виде баров.....</i>	<i>18</i>
<i>Двухсекционный график с объемами и ценой в виде свечей.....</i>	<i>19</i>
<i>График с нарастающими итогами.....</i>	<i>19</i>
<i>Импорт созданных графиков.....</i>	<i>20</i>
<i>Стратегия трейдинга «Путь черепах».....</i>	<i>21</i>
<i>Описание алгоритма «Путь черепах».....</i>	<i>21</i>
<i>Особенности реализации алгоритма «Путь черепах».....</i>	<i>25</i>
Заключение.....	28
Список литературы.....	30

Введение

В начале 2000-х гг. среди производителей СУБД появилась тенденция встраивать внутрь СУБД инструменты для разработки WEB-приложений. Одним из первых таких инструментов стал Oracle APEX (Oracle Application Express), разработанный компанией Oracle. За 10 лет своего существования он превратился в самостоятельный продукт, бесплатно поставляемый вместе с СУБД. Особенностью этого инструмента является то, что все элементы, возникающие в цикле разработки приложения, хранятся непосредственно в инфраструктуре Oracle Database и могут быть развёрнуты на экземпляре Oracle без дополнительного программного обеспечения (используя встроенный в СУБД Oracle WEB-сервер).

Oracle APEX имеет широкий спектр функциональных возможностей для разработки WEB-приложений. В том числе и для наглядной визуализации данных, а именно для создания различного рода графиков и диаграмм, построение которых осуществляется при помощи сторонней компоненты AnyChart.

Однако, не смотря на кажущееся разнообразие, при создании любого приложения встает проблема нехватки инструментов позволяющих визуализировать информацию в соответствии со спецификой выбранной предметной области.

Целью данной работы является расширение шаблонов визуализации данных ORACLE APEX, ориентированных на финансовые приложения, а именно, на финансовый трейдинг.

В связи с этим становится необходимым реализация поддержки в Oracle APEX базовых для трейдинга графиков, а именно: графика с двойной шкалой, двухсекционного графика с объемами и ценой в виде

баров, двухсекционного графика с объемами и ценой в виде свечей, графика с нарастающими итогами.

Для демонстрации созданных расширений в работе будет представлено APEX-приложение, реализующее известную финансовую стратегию трейдинга «Путь черепах» (1).

Поставленные цели определяют следующий список задач:

1. Изучить теоретические основы финансового трейдинга и основные виды графиков, используемые при анализе текущей ситуации на рынке и заключении торговых сделок;
2. Разработать механизм расширения возможностей Oracle APEX позволяющий использовать в приложениях графики следующих видов:
 - График с двойной шкалой;
 - Двухсекционный график с объемами и ценой в виде свечей;
 - Двухсекционный график с объемами и ценой в виде баров;
 - График с нарастающими итогами;
3. Изучить и описать в строгой математической форме алгоритм финансового трейдинга «Путь черепах»;
4. Реализовать, апробировать на живых данных и выявить актуальность практического применения алгоритма "Путь черепах".

Глава 1. Обзор графических основ трейдинга и используемых инструментариев

Основы финансового трейдинга

Трейдинг (от англ. «trade» — «торговля, торговать») - это работа биржевого торговца (трейдера), заключающаяся в торговле различными финансовыми инструментами на рынке.

Для успешного трейдинга необходимо прогнозирование изменений цен в будущем на основе анализа изменений цен в прошлом (технический анализ). Помимо ценовых рядов в техническом анализе используется информация об объемах торгов и другие статистические данные:

Цена контракта (индекса) - стоимость контракта на бирже в определенный период времени.

Объем торговли - общее количество заключенных контрактов за определенный промежуток времени, например, за торговый день.

Открытый интерес - количество позиций, не закрытых на конец торгового дня.

Главным инструментом технического анализа служат графики. По графику можно определить, где находился рынок, а после идентификации ценовых фигур и соответствующих из них выводов, он может подсказать ценные идеи относительно возможного направления рынка (2).

Обзор традиционных для трейдинга графиков

Рассмотрим основные типы графиков, на основе которых производится технический анализ.

График с двойной шкалой

При графическом анализе рынка может возникнуть необходимость отображения данных разной размерности в рамках одной системы координат. Например, для визуализации цены, размерность которой может достигать тысяч, и количества открытых позиций, измеряющегося в единицах. Поэтому появляется потребность в графике с двумя вертикальными координатными осями с разным шкалированием. Пример графика:



Двухсекционный график с объемами и ценой в виде баров

Одним из наиболее распространенных видов отображения динамики цены в графическом анализе рынка является двухсекционный график с объемами в виде баров (bar chart).

Данный график состоит из двух секций:

Первая секция : «График баров»

Вторая секция : «График объемов»

Под понятием «Бар» понимается вертикальная линия, на которой отображается информация о том, как менялась цена в течение

выбранного периода времени. Таким образом, каждому временному периоду сопоставляется соответствующий «Бар». Бары следуют один за другим в хронологическом порядке выбранного временного промежутка и образуются независимо от характера изменения цены.

Каждый бар содержит четыре важнейших показателя движения цены:

Open (цена открытия) - значение, которое было в самом начале выбранного периода. Отображается небольшой горизонтальной чертой слева от бара.

Close (цена закрытия) - значение, которое было в самом конце выбранного периода. Отображается небольшой горизонтальной чертой справа от бара.

High (максимальная цена) - максимальное значение, достигнутое в течение выбранного временного периода. High - это верхняя граница бара.

Low (минимальная цена) - минимальное значение, достигнутое в течение выбранного временного периода. Low - это нижняя граница бара.



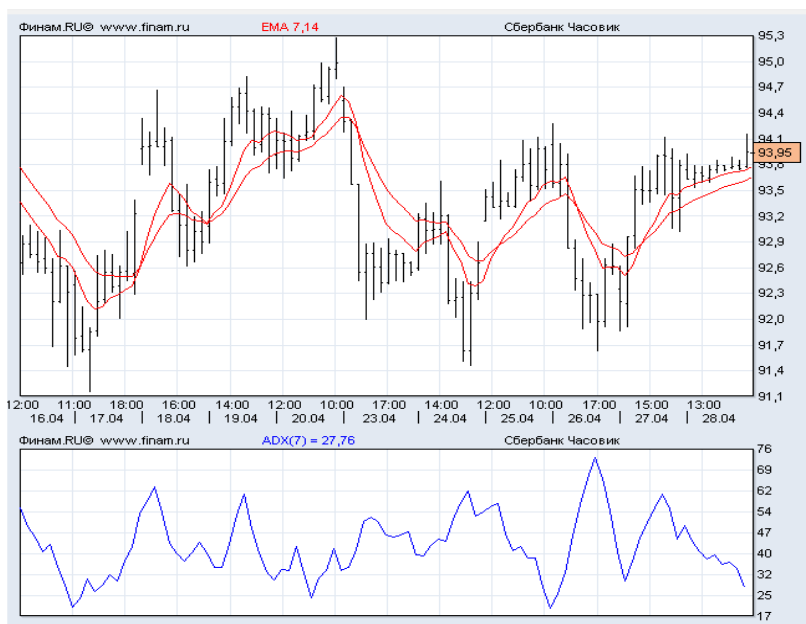
Трейдеры предпочитают использовать в анализе рынка баровый график из-за его наглядности и информативности. Одним из

преимуществом данного вида графиков является возможность просмотра, так называемых «гэпов» (gap).

«Гэп» (от англ. «gap» - разрыв) – в анализе потока котировок, это ситуация существенной разницы между ценой закрытия предыдущего временного периода и ценой открытия следующего (3).

Обычно трейдеры используют дневные и более короткие интервалы отображения цены на графике, но для анализа длительных тенденций могут применяться недельные и более продолжительные графики баров.

Анализ объема торгов так же является неотъемлемым элементом технического анализа, поэтому кроме цены на графике часто отображают объем. По динамике объема можно судить о значимости и силе ценового движения (4). Пример графика, взят с finam.ru, сайта одного из наиболее крупных инвестиционных холдингов России(5):



Двухсекционный график с объемами и ценой в виде свечей

В графическом анализе японские свечи (candle chart) получили распространение с 90-ых годов. Название «японские свечи» берет свои

истоки от японских торговцев рисом, которые еще несколько веков назад использовали такие графики в своем деле.

Данный график состоит из двух секций:

Первая секция : «График свечей»

Вторая секция : «График объемов»

График свечей отображает движение цены за определенный период времени с дополнительным цветовым выделением, показывая связь и взаимное расположение цен открытия (open) и закрытия (close).

Каждому временному периоду – сопоставляется соответствующая свеча, которая состоит из «тела» и «тени».

«Тело» свечи – это расстояние между ценами открытия и закрытия данного временного периода.

«Тень» свечи – это минимальное и максимальное значения цен, достигнутые за этот же отрезок времени.



В случае если high или low расположены в пределах тела свечи, то тень не отображается. Также встречаются свечи без тела, когда цена открытия совпадает с ценой закрытия. Разнообразие свечей огромное количество, даже существует специальный метод свечного

анализа для предсказания дальнейшего движения цены, выделенный в отдельную категорию исследования рынка (6).

По внешнему виду свечи можно сразу определить направленность изменения цены за рассматриваемый промежуток времени:

- если цена закрытия окажется ниже цены открытия, то тело свечи закрашивается (относительно фона графика);
- если цена закрытия располагается выше цены открытия, то тело свечи остается полым.

Вторая секция (график объемов) совпадает со второй секцией двухсекционного графика с объемами в виде баров. Пример графика с finam.ru:

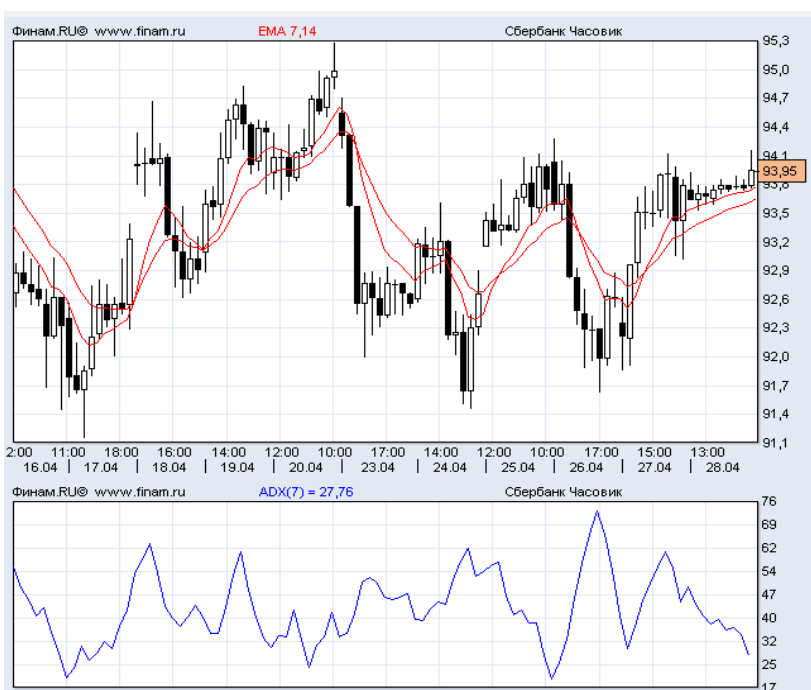
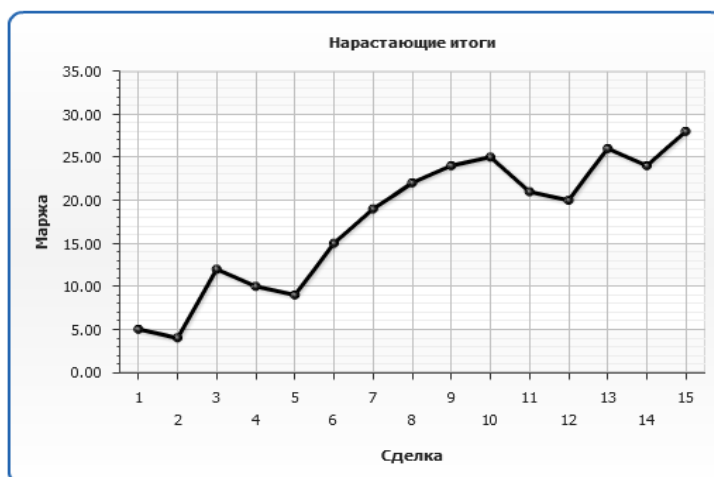


График с нарастающими итогами

Потребность в данном виде графиков может возникнуть, например, при подсчете суммарной прибыли по сделкам. Например, имеется информация следующего содержания (дата, маржа):

Дата сделки	Маржа
01.02.08	5
02.02.08	-1
03.02.08	8
04.02.08	-2
05.02.08	-1
06.02.08	6
07.02.08	4
08.02.08	3
09.02.08	2
10.02.08	1
11.02.08	-4
12.02.08	-1
13.02.08	6
14.02.08	-2
15.02.08	4

На графике хотелось бы видеть не прибыль по отдельным сделкам, а суммарную прибыль на момент времени t_1 , t_2 и т.п. При приведенных выше данных график с нарастающими итогами должен выглядеть так:



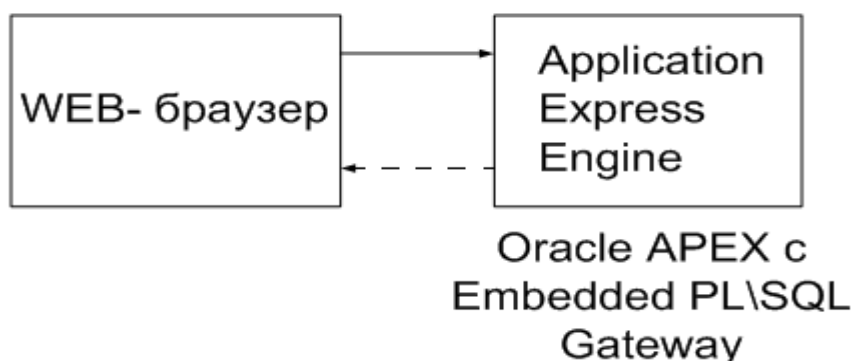
Описание используемых инструментариев

Oracle APEX описание среды разработки

Oracle APEX (Oracle Application Express) - среда разработки прикладного программного обеспечения на основе СУБД Oracle Database, целиком реализованная как WEB-приложение. Позволяет быстро проходить весь процесс разработки WEB-приложения и генерировать из данных, хранящихся в БД Oracle, динамические страницы программ, а так же обрабатывать их в режиме реального времени. При создании или расширении программ, написанных в APEX, метаданные, хранящиеся в таблицах БД Oracle, модифицируются, а во время работы программы APEX считывает эти метаданные и отображает программу в браузере. APEX полностью располагается в БД Oracle и представляет собой набор данных, структурированных в таблицах с большим количеством PL/SQL кода (порядка 200 PL/SQL объектов и 215 таблиц, содержащих более 300 тысяч строк кода). Oracle APEX совместима с версией Oracle 9.2 и выше, а, начиная с версии Oracle 11g, среда устанавливается по умолчанию вместе с СУБД. В качестве Web-сервера, для отображения страниц, используется Apache или Embedded PL/SQL Gateway (EPG) (встроенный в СУБД Oracle Web-сервер). EPG используется в Oracle Database Express Edition 10.2.0.1 и выше. Вся разработка – декларативная, проводится внутри БД и для ее осуществления достаточно знать только PL/SQL. Так же характерно полное отсутствие ООП и всяческих «прослоек» типа ORM (Object-relational mapping).

Во время работы потребление ресурсов CPU минимизируется за счет использования асинхронной архитектуры управления сессией. Всегда выполняются одни и те же процессы, вне зависимости от того запускаете среда разработки или уже готовое приложение, созданное в

Oracle APEX. Браузер посылает URL-запрос, который преобразуется в соответствующий вызов Application Express PL/SQL. База данных обрабатывает PL/SQL запрос и отправляет результат обратно в браузер в виде HTML страницы. Эти операции выполняются всегда при запросе или получении страницы. Благодаря тому, что управление сессией осуществляется в базе данных, отсутствует необходимость в дополнительном программном обеспечении для организации связи. Показ новой страницы сопровождается появлением новой сессии базы данных, за счет чего достигается экономия ресурсов сервера так, как они используются лишь во время запроса к APEX, генерации и обработке страниц. В версиях Oracle новее 10g или 11.1 процесс преобразования URL выглядит следующим образом:



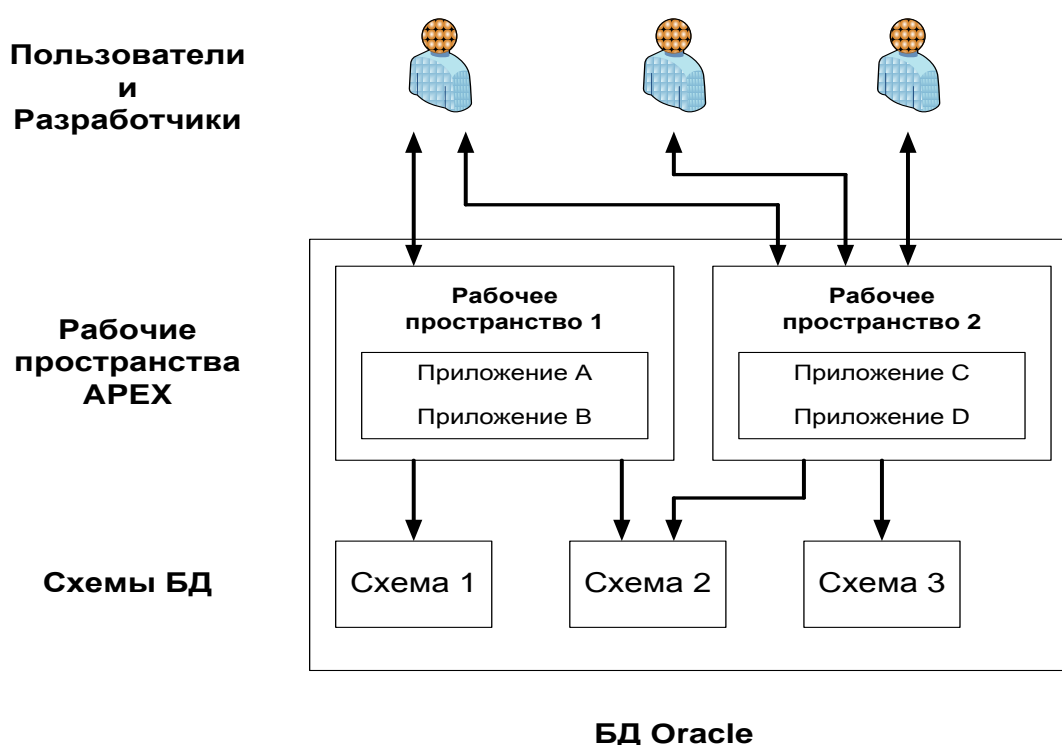
EPG выполняет функции Web-сервера и обеспечивает инфраструктуру, необходимую для создания динамических приложений, причем не требует Oracle HTTP Server (7).

В процессе работы ядро Oracle APEX занимается следующими задачами:

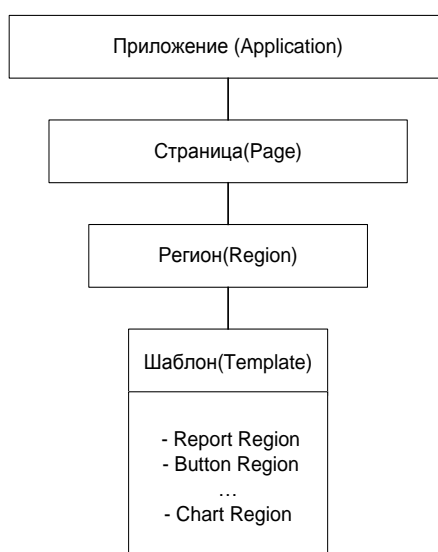
- управляет состояниями сессии;
- выполняет аутентификацию и авторизацию сервисов;
- осуществляет контроль выполнения приложений;

- выполняет проверку состояния обработок запросов;
- управляет потоком страниц.

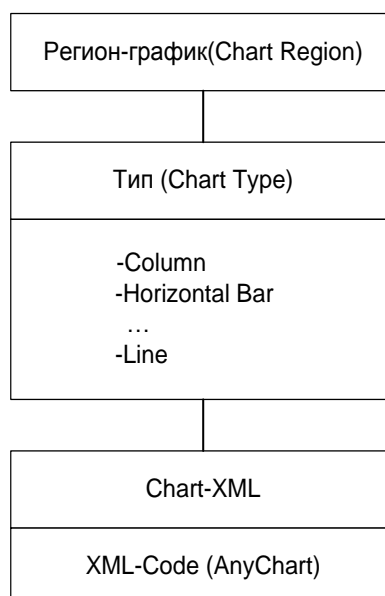
Все пользователи Oracle APEX работают с одной общей БД, для доступа к которой достаточно Web-браузера, дополнительное ПО не требуется. Область, где разрабатываются приложения, называется рабочим пространством (workspace). Рабочее пространство - это виртуальная частная БД, позволяющая множеству пользователей работать с одним экземпляром Oracle APEX, скрывая при этом свои объекты, данные и приложения. При создании рабочего пространства необходимо указать в какой схеме оно будет располагаться, существующей или новой. Схема - это логический контейнер для объектов базы данных, таких как таблицы, представления и хранимые процедуры. Одна схема может быть связана с одним или несколькими рабочими пространствами. На следующем рисунке демонстрируется взаимосвязь между пользователями/разработчиками, рабочими пространствами и схемами базы данных:



Любое приложение (Application) Oracle APEX состоит из страниц (Page). Страницы в свою очередь содержат регионы, внешний вид которых задается посредством шаблонов, это может быть, например, регион-отчет (Reports region), отображающий данные в виде таблиц, кнопка (Button region), регион-график(Chart Region) и т.д., всего имеется порядка 30 различных шаблонов (8). Структуру приложения Oracle APEX можно описать следующей схемой:



Как видно именно на уровне шаблонов задается конечный вид региона. Чтобы отображать данные в виде графика нужно, чтобы шаблон региона был: Chart Region. Рассмотрим структуру Chart Region:

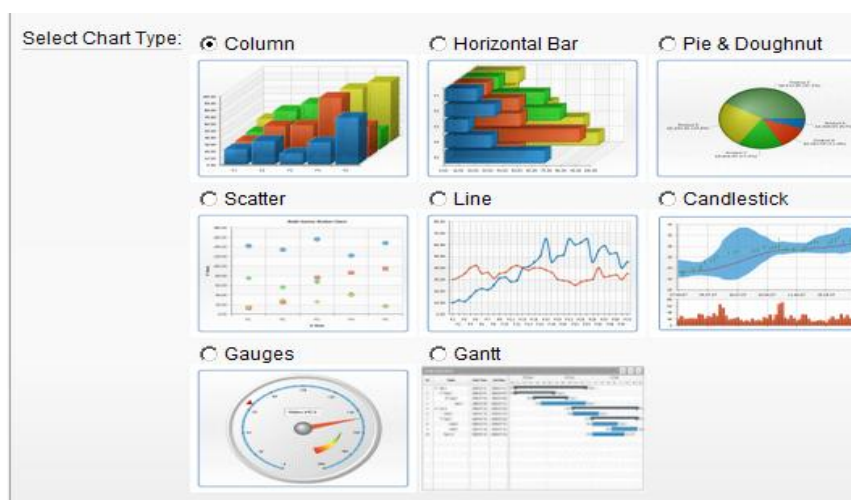


Далее рассмотрим, какие типы графиков поддерживаются в APEX на данный момент.

Графики Oracle APEX

В Oracle APEX поддерживаются следующие типы графиков (ChartType):

1. **Column** - отображение данных в виде «столбиков», направленных вверх;
2. **Horizontal Bar** - отображение данных в виде «столбиков», направленных слева на право;
3. **Pie & Doughnut** - отображение данных в виде круга с долями;
4. **Scatter** - отображение точечно разбросанных данных;
5. **Line** - обычный линейный график;
6. **Candlestick** - отображение японских свечей. Картинка, описывающая график полностью не соответствует реальному содержанию. График не удовлетворяет поставленным задачам в силу отсутствия поддержки визуализации объемов и несоответствия необходимому стилю пользовательского интерфейса;
7. **Gauges** - отображение данных в виде циферблата;
8. **Gantt** - изображения диаграмм Ганта.



Итак, среди стандартных графиков Oracle APEX, только один (Candlestick) можно было бы использовать для финансовых приложений, но он не включает объемы, характерные для технического анализа.

Anychart

Для построения графиков и диаграмм в Oracle APEX встроена компонента AnyChart (9), для корректной работы которой, необходим установленный Flash Player версии 9 и выше.

Поддержка компоненты Anychart позволяет моделировать графики в соответствии со спецификой предметной области и создавать компактную, интерактивную и красивую визуализацию данных.

Anychart представляет собой набор Flash (SWF) файлов, благодаря которым ставится возможным создание анимированных графиков. Компонента полностью управляется XML интерфейсом, который позволяет задавать огромное количество настроек таких как: прозрачность, цвет, WEB-ссылки, подписи и другое. Все настройки устанавливаются через XML файл, поэтому отсутствует необходимость в изменении Flash-файла (10).

Глава 2. Особенности реализации

Исходя из архитектуры Oracle APEX (раздел работы «Oracle APEX описание среды разработки») можно заметить, что для того чтобы добиться поддержки нового вида графиков необходимо опуститься до уровня Chart-XML, за который отвечает компонента AnyChart. К сожалению, при помощи создания шаблона вида: регион-график (Template Class: Chart Region), через стандартные интерфейсы APEX доступ к этому уровню осуществить невозможно. Поэтому было решено использовать Application Process типа OnDemand, решающий эту проблему. Для каждого нового вида графика был создан свой Application process, который возвращает данные в виде XML, сформатированном для компоненты AnyChart (формируется XML-строка простым циклом по SQL-запросу), а также задает внешний вид графиков на все том же XML. Таким образом, для поддержки нового вида графика необходимо создать регион, в котором он будет отображаться и в его коде (Region Source) осуществить вызов соответствующего Процесса приложения (Application Process). Процессы приложения (Application Process), так же как и шаблоны, можно выгружать в формате .sql и импортировать или экспортировать между разными экземплярами Oracle Database.

При создании внешнего вида графиков ориентация, там где возможно, шла на дизайн пользовательского интерфейса информационного портала о финансовых рынках finam.ru (3).

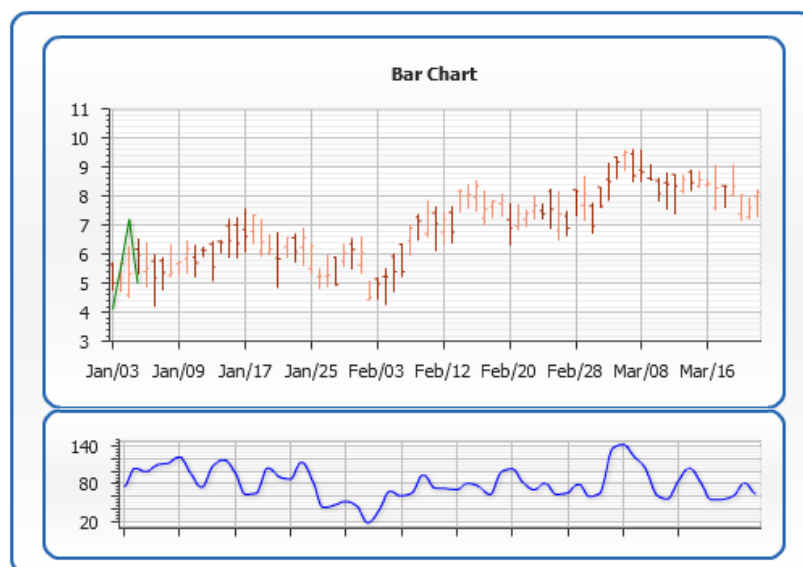
График с двойной шкалой

После реализации Процесса приложения (Application process) под названием ExtraAxes и его вызова, в Oracle APEX описанный выше график имеет следующий вид:



Двухсекционный график с объемами в виде баров

После реализации Процесса приложения (Application process) под названием BarChart и его вызова, в Oracle APEX описанный выше график имеет следующий вид:



Двухсекционный график с объемами в виде свечей

После реализации Процесса приложения (Application process) под названием CandleChart и его вызова, в Oracle APEX описанный выше график имеет следующий вид:

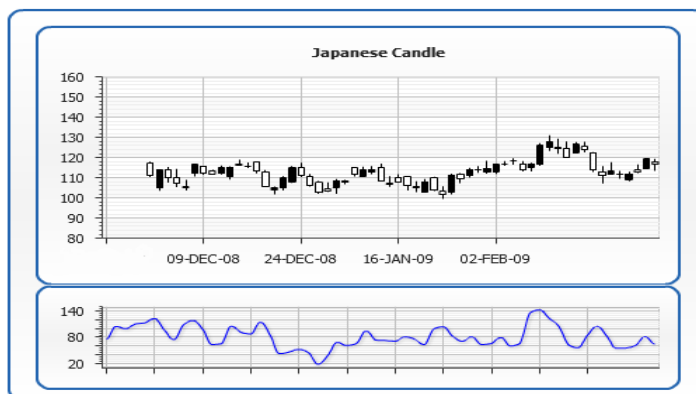
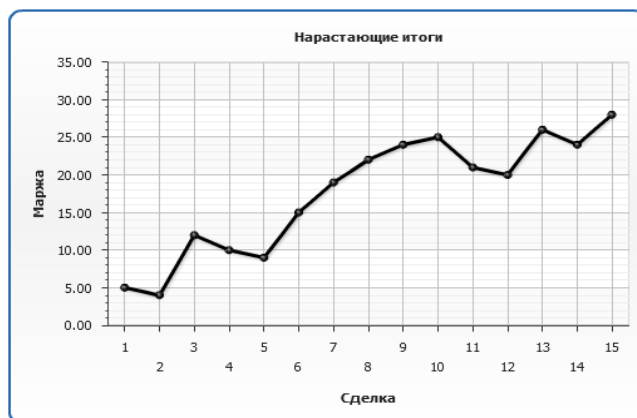


График с нарастающими итогами

Нарастающие итоги можно было бы построить SQL-запросом, на основе таблицы из раздела «Глава 1. График с нарастающими итогами». Однако такой запрос имел бы очень сложный вид и крайне долго исполнялся.

Поэтому было решено справиться с данной задачей графически, без построения дополнительных таблиц, просто прибавляя к абциссе предыдущей точки на графике соответствующее значение из таблицы.

После реализации Процесса приложения (Application Process) под названием RunningTotal и его вызова, в Oracle APEX описанный выше график имеет следующий вид:



Пожалуй, главным достоинством этого графика является то, что данные из таблицы в 1000 строк он прорисовывает почти в 500 раз быстрее, чем это делалось бы при помощи обычного графика после исполнения соответствующего sql-запроса.

Импорт созданных графиков

Чтобы импортировать процесс приложения на свой экземпляр Oracle APEX, необходимо внутри Oracle APEX зайти в «Export\ Import» -> «Import», выбрать файл, например, ChartTemplate.sql (он должен быть заранее скачен), в File Type выбрать Database Application, Page or Component. После нажатия кнопки «Next» компонента добавится в приложение и будет готова к использованию. Все загруженные файлы хранятся в отдельном репозитории, посмотреть который можно в «Application Builder» -> «Export Repository».

Для корректной установки необходимо, чтобы ID рабочей области (Workspace ID), Name , theme_id в файле экспорте и приложении, куда устанавливается компонента, совпадали. В случае несовпадения нужно вручную прописать в файле экспорта соответствующие значения того приложения, куда устанавливается компонента.

Стратегия трейдинга «Путь черепах»

«Путь черепах» - стратегия финансового трейдинга, описывающая стратегию биржевой торговли двух известных американских трейдеров: Ричарда Денниса и Вильяма Экхардта. Эта стратегия была опубликованная их учеником Куртисом Фейсом в одноименной книге в 2008 году.

Чтобы продемонстрировать работу созданных мною шаблонов, было решено реализовать приложение демонстрирующее работу этой стратегии.

Книга написана не математиком, поэтому основные шаги алгоритма торговли не собраны в едином месте, а разбросаны по всей книге, в связи с чем, первостепенной задачей стояло написание алгоритма торговли в строгой математической форме, а уже затем создание приложения реализующего эту стратегию.

Созданное приложение позволит узнать является ли «Путь черепах» прибыльной стратегией в условиях современной экономики. Или же, способы торговли, придуманные известными трейдерами конца прошлого столетия, в настоящее время уже являются непригодными. Изначально алгоритм предназначался для дневных баров, так как на момент создания алгоритма не было возможности получать данные по более мелким временным периодам. Отсюда возникает вопрос возможности адаптации алгоритма к данным имеющим более низкую периодичность баров например часовую, получасовую и т.д

Описание алгоритма

На вход алгоритму подается максимально возможное количество открытых позиций. Далее основываясь на некоторых правилах

торговли и исходных данных, осуществляются сделки по купле\продаже контрактов выбранного финансового инструмента.

Введем несколько основных определений и приведем ключевые формулы.

Истинный диапазон (ATR) - технический показатель описывающий уровень изменчивости цены анализируемого финансового инструмента. ATR вычисляется по следующей формуле:

$$\text{Истинный диапазон} = \text{Максимум} (H - L, H - PC, PC - L)$$

Где:

H - Максимум за временной период (High)

L - Минимум за временной период (Low)

PC - цена закрытия предыдущего временного периода (Previous Close)

Число N - показатель для отображения волатильности конкретного рынка. В течение первых 20 временных промежутков число N не вычисляется, а на 21 инициализируется, как среднее значение истинных диапазонов последних 20 временных промежутков. Далее число N вычисляется по следующей формуле:

$$N = (19 * PN + TR) \setminus 20$$

Где:

PN - значение N предыдущего временного промежутка (Previous N)

TR - текущий истинный диапазон

Юнит - единица измерения размера позиции, измеряется в контрактах.

Размер юнита вычисляется по следующей формуле:

$$U = \text{Round} [(0,01 * Inv) \setminus V_{\text{rub}}]$$

Где:

U - размер юнита

Inv – размер торгового счета

V_{rub} - волатильность в рублях

Волатильность в рублях в свою очередь вычисляется как:

$$V_{rub} = N * CP$$

Где:

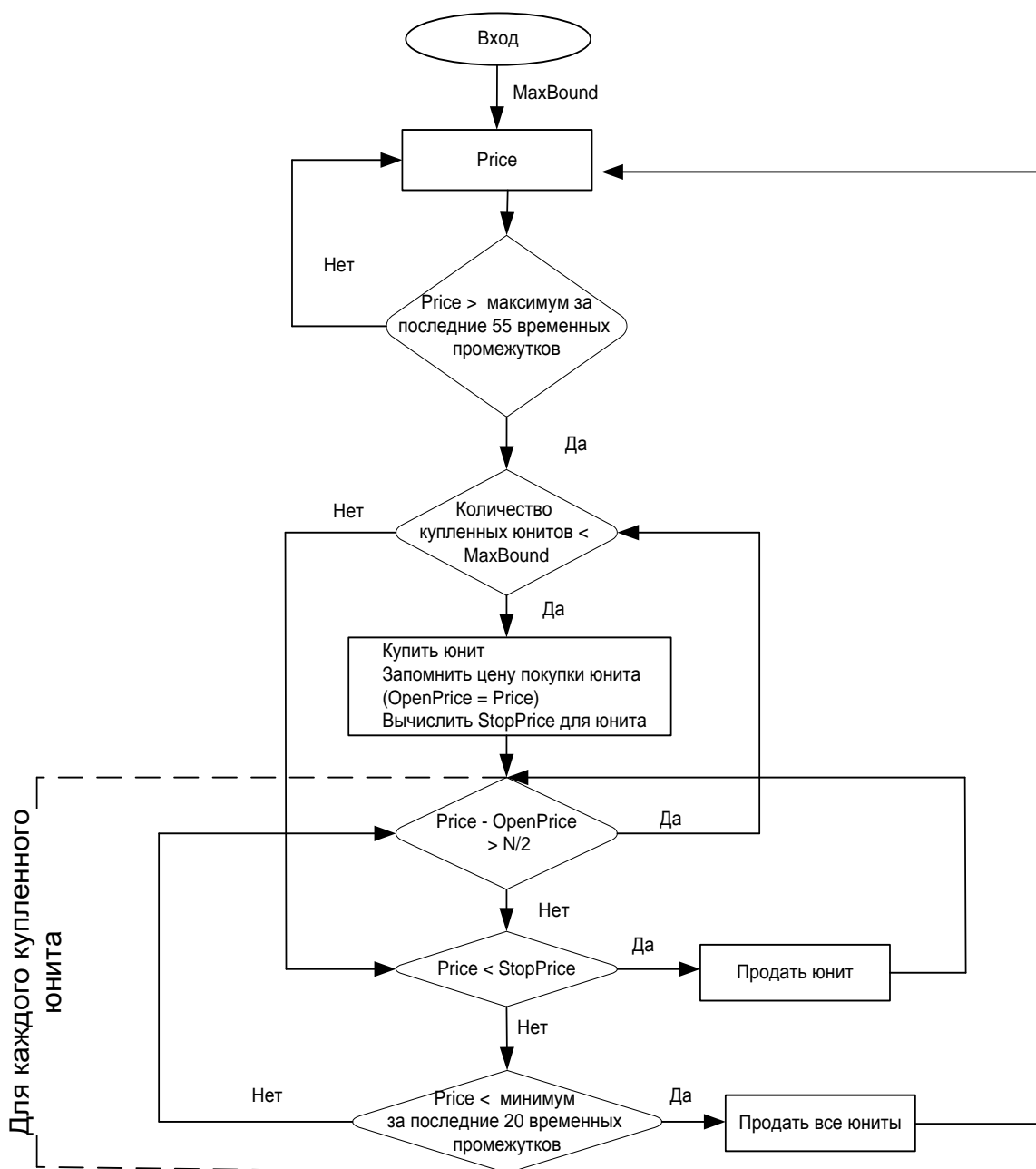
CP - цена одного контракта в рублях

N – число N временного промежутка

В качестве исходных данных для алгоритма выступает таблица со следующей информацией:

- 1)Дата
- 2)Максимум за временной промежуток
- 3)Минимум за временной промежуток
- 4)Цена открытия за временной промежуток
- 5)Цена закрытия за временной промежуток
- 6)Истинный диапазон
- 7)Число N

Алгоритм торговли может быть изображен следующей блок-схемой:



Где:

Price – текущая цена за наблюдаемый контракт;

MaxBound – максимально возможное количество юнитов (входной параметр);

OpenPrice – цена покупки юнита;

N – число N текущего временного промежутка (берется из таблицы);

StopPrice – критическая отметка цены, при которой мы еще держим юнит. Для первого купленного юнита вычисляется, как: $OpenPrice - 2 * N$, при покупке последующих stopPrice увеличивается на $N/2$ для всех юнитов.

Например, рассмотрим следующие цены на золото за конец 2008 года.

DATE...	▲	OPEN	HIGH	LOW	CLOSE	VOL	N	ATR	OPEN_PRICE	CLOSE_PRICE	STOP_PRICE
12.12.2008		825	832,70001	814	821,70001	10262		18,70001			
15.12.2008		825,5	847	825	840,5	15060		25,29999			
16.12.2008		840,40002	863,59998	834,20001	859,5	13185		29,39997			
17.12.2008		860	887,20001	851	874	17099		36,20001			
18.12.2008		872	882	852,79999	860	16463		29,20001			
19.12.2008		858,29999	858,29999	834,40002	844,79999	19128		25,59998			
22.12.2008		846	856,70001	844	853,5	10916	23,2399953333333	12,70001			
23.12.2008		850,5	852	834,40002	843,59998	11900	23,0329945666667	19,09998			
24.12.2008		844,90002	855	839	853,90002	6228	22,6813448383333	16			
25.12.2008		851	877	825	849	959	24,1472775964167	52			
26.12.2008		850,79999	871,79999	848	871	3909	24,1299132165958	23,79999			
29.12.2008		871,79999	893	871,79999	884	13174	24,023417555766	22			
30.12.2008		881,29999	882,29999	869	879	7401	23,5722466779777	15	877		829,855506644045
31.12.2008		877	881,40002	869,20001	870,59998	1586	23,0036348440788	12,20001			
11.01.2009		859,90002	867,59998	854	864,79999	910	22,6834521018749	16,59998			
12.01.2009		861	861	818,09998	822,20001	18154	23,8842799967812	46,70001			
13.01.2009		819,5	832,59998	816,20001	823,90002	12016	23,5100644969421	16,39997	824,90002		
14.01.2009		824,90002	831	809,20001	814,90002	8440	23,424560772095	21,79999			
15.01.2009		811	823,29999	802,29999	821,70001	16366	23,3033327334902	21			

30.12.2008 цена достигла максимума за последние 55 дней и мы решаем открыть позиции. StopPrice в таком случае будет равняться: $877 - 23,57 * 2 = 829,86$. Соответственно если цена достигнет уровня $877 + 23,57 / 2 = 888,785$ будем покупать еще один юнит, а если спустится ниже отметки 829,86 будем продавать только что купленный.

Особенности реализации алгоритма «Путь черепах»

Алгоритм оформлен в качестве самостоятельного модуля, который автоматически создает все необходимые для реализации алгоритма таблицы, вызывает соответствующие процедуры и функции и запускает алгоритм на основе заранее подготовленных данных по

цене выбранного финансового инструмента (например, экспортированных с finam.ru).

В ходе реализации алгоритма для каждого выбранного контракта была создана основная таблица с полями (id,date_deal, open, high, low, close, open_price, N, ATR, close_price, open_position), и вспомогательная units (id, stop_price, contracts). Где: date_deal – дата; open, high, low, close – величина открытия, закрытия, максимальной и минимальной цены за промежуток от предыдущего date_deal до текущего; N - число N текущего дня; ATR-текущий истинный диапазон цены; open_price, close_price – цена по которой совершалась сделка купли или продажи, если таковая была; open_position – количество купленных или проданных контрактов; stop_price – критическая отметка цены для продажи юнита; open_position – размер юнита в контрактах.

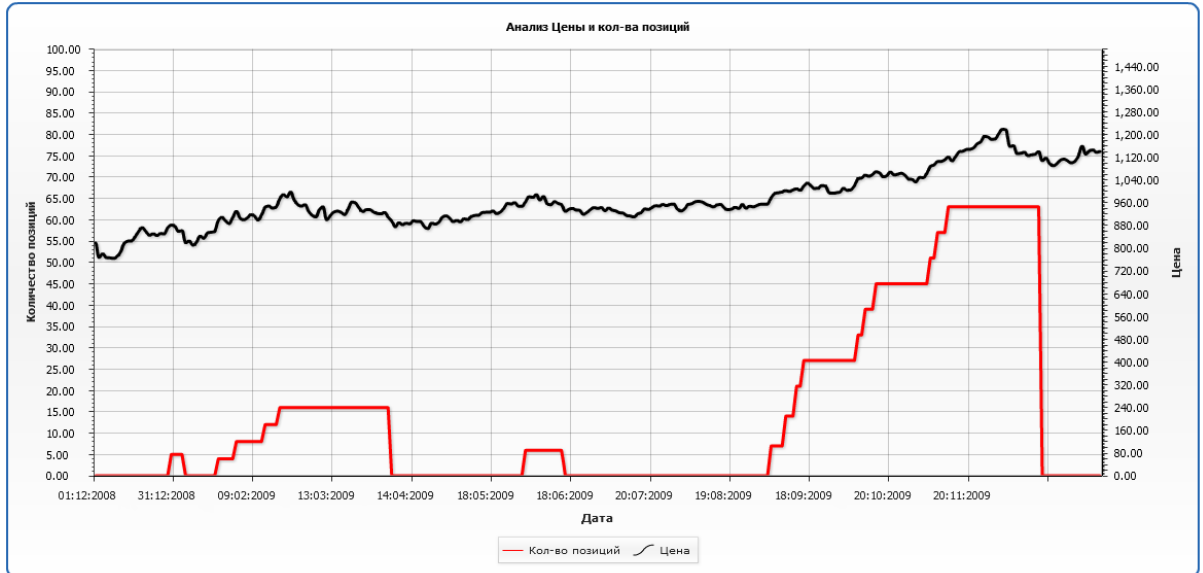
Ориентируясь на основную таблицу, проводился анализ работы алгоритма и строились графики, визуализирующие его работу. Таблица units была необходима для того, чтобы отслеживать количество юнитов имеющихся на руках, а так же для хранения стоп-цены по каждому купленному юниту.

Основная логика алгоритма была реализована, как хранимая процедура `tortoise_way`, принимающая параметром количество юнитов, которое максимально можно держать на руках (`maxBounds`). Процедура `tortoise_way` использовала внутри себя вызов ряда вспомогательных процедур.

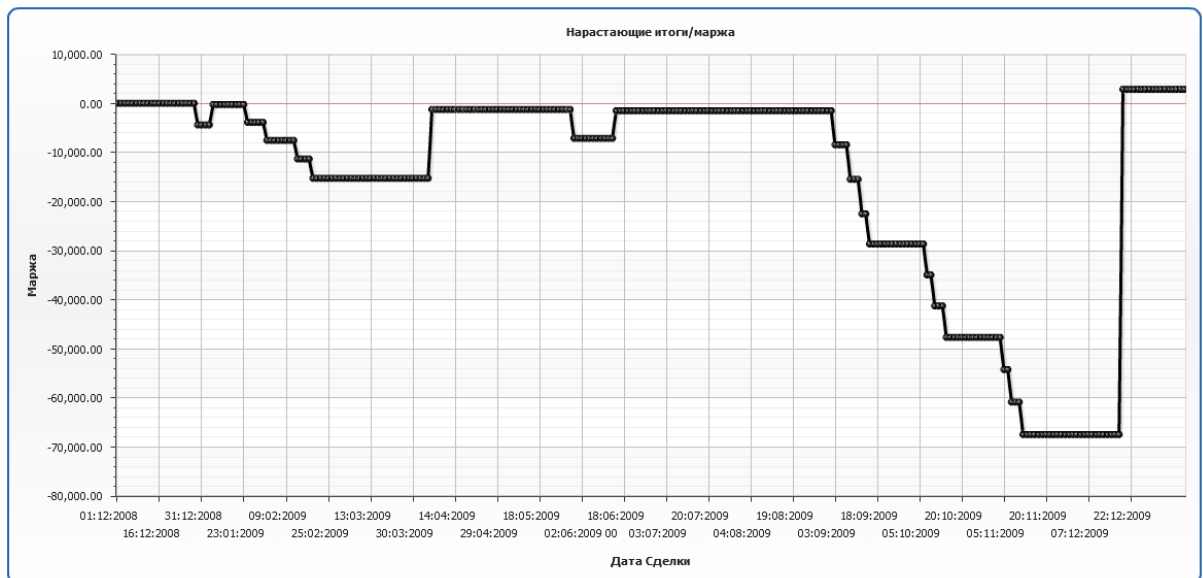
Апробировать алгоритм было решено на низковолатильном контракте SPFB.GOLD (фьючерсы по золоту), SPFB.SILV(фьючерсы по серебру) и высоковолатильном RIM2 (индекс РТС) за период с 01.12.2008 по 04.04.2012. Периодичность цены для сравнения бралась дневная и часовая.

Все необходимые данные по выбранным контрактам были экспортированы с сайта finam.ru.

Например, результат работы алгоритма на дневном контракте SPFB.GOLD за год можно наблюдать на следующем графике:



Нарастающие итоги маржи по результатам работы на основе цены по дням:



Результаты апробации алгоритма на перечисленных контрактах можно наблюдать в следующей сводной таблице:

Название контракта	Периодичность цены	Маржа	Маржа\Средняя цена 1 контракта
SPFB.GOLD	день	3846	5
SPFB.GOLD	час	-11918	-12
SPFB.SILV	день	4891	212
SPFB.SILV	час	-8331	-361
RIM2	день	-16565	0
RIM2	час	-208280	-1

После произведенной апробации можно сделать вывод, что алгоритм для баров с меньшей периодичностью работает хуже, чем для баров с большей периодичностью, что не удивительно, так как алгоритм был разработан в то время, когда информацию по часовым и минутным барам получить было невозможно. Так же просматривается тенденция хороших результатов работы алгоритма непосредственно на сырьевых контрактах. Чтобы более точно оценить прибыльность алгоритма, необходимо апробировать его на большем количестве контрактов.

Заключение

В ходе данной дипломной работы были достигнуты следующие результаты:

1. Разработан механизм расширения возможностей Oracle APEX позволяющий использовать в приложениях графики следующих видов:
 - График с двойной шкалой;
 - Двухсекционный график с объемами и ценой в виде свечей;
 - Двухсекционный график с объемами и ценой в виде баров;
 - График с нарастающими итогами;
2. Реализован и визуализирован при помощи полученных графиков алгоритм финансового трейдинга «Путь черепах».
3. Алгоритм апробирован на живых данных с finam.ru по дневным и часовым барам контрактов по золоту, серебру и индексу РТС за период с 01.12.2008 по 04.04.2012.

Список литературы

1. Фейс, К. (2008). Путь черепах. Москва: Манн, Иванов и Фербер, 304 стр.
2. Эрлих, А. (1996). Технический анализ товарных и финансовых рынков. Москва: Инфра-М, 176 стр.
3. Обучение форекс начинающим, URL: <http://www.forex-mix.ru/>
4. Акелис, С. (1999). Технический анализ. Москва: Диаграмма, 214 стр.
5. Официальный сайт холдинга «Финам», URL: <http://www.finam.ru/>
6. Нисон.С. (1998). Японские свечи. Графический анализ финансовых рынков. Москва: Диаграмма, 336 стр.
7. Oracle® Application Express Application Builder User's Guide (Release 4.1), 818 стр.
8. Oracle® Database 2 Day + Application Express Developer's Guide (Release 4.1), 82 стр.
9. Официальный сайт Anychart, URL: <http://www.anychart.com/>
10. John Edward Scott and Scott Spendolini. (2008). Pro Oracle Application Express. New York: Apress, 712 стр.