

# Visual Age for Java 3.5

Владислав Каменский

Данный краткий обзор будет посвящен, не так давно выпущенному в свет, средству разработки Java-приложений от IBM.

Что можно сказать по поводу Visual Age for Java (VAJ). Начал я свое знакомство, с этим продуктом этим летом. Тогда еще в ходу была версия 3.0, первое впечатление было просто замечательным. Очень оригинальный дизайн, непохожий на семейство JBuilder от Borland, главной его особенностью, является простота интерфейса. Нет тех многочисленных окон и кнопок, которые пугают новичка, и мешают настроиться на серьезную работу опытного разработчика (последнее шутка). А самой прикольной фишкой является то, что IDE показывает в окошке только одну функцию, а не весь класс сразу. Причем каждый раз после проделанных изменений, при переходе к другому методу, VAJ выдает запрос на сохранение и тут же компилирует класс, указывая на ваши ошибки. Это просто шокирует людей которые привыкли сохранять и компилировать по своей команде. Зато потом к этому привыкаешь, и мне это уже, очень нравится. Например, это предохраняет от неожиданного обесточивания вашего компа :). Но была одна проблема, VAJ 3.0 не держал Java 2, это портило весь внешний вид.

И вот чудо свершилось, и в конце августа, IBM возрождается новой версией 3.5, которая, как вы уже понимаете, держит Java 2.(Реализованная спецификация — JDK 1.2.2 )

Перед тем как перейти к возможностям этого продукта, должен предупредить, что если на вашей машине не стоит метров эдак 128 мозгов, то можно даже дальше и не читать, это пока не для вас, к сожалению. (Процессор, как вы понимаете, тоже нужен не ниже P-II).

Итак, чем же хорош, наш конек(VAJ). Меня как разработчика прежде всего интересовал надежный и удобный отладчик. И здесь он присутствует, в виде очень удобной тулзы, которая очень быстро находит те тонкие места, которые сделал в своей

программе незадачливый программист. А вот то, что этот отладчик может легко и непринужденно наблюдать за работой сервлетов, и о чудо, даже JSP, делает его вообще не заменимым. Об отладке серверных приложений я расскажу поподробней. Для отладки сервлетов, предусмотрен встроенный веб сервер (IBM Websphere). Эта штукавина запускается перед тем как вам нужно проверить свои сервлеты (делает она это, надо заметить, мучительно долго. Лениво помигивая, своими немногочисленными лампочками, Websphere как бы оценивает, возможности вашего компьютера, и ежели находит его вполне пригодным, запускается). Замечу, в документации минимальный объем оперативки оценивается, как 128, рекомендуемый, как 256.

Итак веб сервер запущен и что же?. А вот теперь вы можете запускать свои сервлеты, которые находятся на своих местах — в рабочем пространстве вашего VAJ, командой из браузера (типа localhost:8080/servlet/name\_of\_servlet). Причем, ежели вы в каком-нибудь из них поставили свою, контрольную точку, то как только дело дойдет до него, отладчик сделает свое дело. В отображаемом окне, вы увидите всю необходимую для вас информацию, и, я не сомневаюсь, быстро найдете свою ошибку. Работает надежно, но тормозит, при малом объеме оперативки. (Релизованная спецификация — JSDK 2.1)

Насчет JSP, я еще не слишком большой спец в этой области, но я сам видел, как отладчик позволяет по шагам смотреть, что делает JSP, соотнося код JSP и сгенерированного сервлета (для тех кто не в курсе — jsp перед свои исполнением превращается в сервлет). Разве что эта фитча, у меня работала не так надежно, но это я грешу, на свои 128 метров мозгов, видимо надо больше. (Реализованная спецификация — JSP 1.0 )

Касательно визуально разработки — ничего не могу поведать вам, так как никогда не пользуюсь такими услугами. Единственно, что отмечу — приложения, написанные на Swing при своем запуске очень тормозят (о причине вы, наверное, уже догадались).

Ну вот вроде бы и все, что я хотел сказать. Какой же сделать вывод ? VAJ очень требователен к ресурсам, но если его отблагодарить, и поставить на приличную машину, то он вас отблагодарит, и вы насладитесь полноценной разработкой Java — приложений.