Глава 1

## Концепция автономных агентов

**§ 1. Автономные агенты**

Данная глава будет являться вводной и в ней мы более подробно познакомимся с автономными агентами, как об одном из типов программ, а также рассмотрим принцип создания программы на основе написания агентов для отдельного объекта и последующего соединения их воедино.  
 Итак, что же такое автономные агенты? Обычно, при программировании мы создаем программы, которые работают строго по тем направлениям, что мы им задали, и по сути, их поведение можно легко предсказать. Если же мы хотим, чтобы программа вела себя, как будто живая и условно говоря, помогала нам с исследованием, мы должны использовать такой тип программ, как автономные агенты. Отсюда, автономные агенты - программы*,* что самостоятельно проводят анализы, делают выводы и, исходя из него, следуют какому-то алгоритму действий, который они сами составили. Отличие автономных агентов от других программ является то, что:

* Автономные агенты сами получают для себя конечную информацию, обрабатывая полученные данные и на основе этой информации делает выводы.
* Автономные агент не имеет над собой так называемого лидера, которые как-то руководил выводами программы.
* Агенты могут обрабатывать только лишь ограниченный участок информации, поэтому мы и разбиваем большую программу не на 2 и не на 3 агента, а на намного большее количество.

Как было сказано выше, с помощью разбиения на автономные агенты проще программировать большие программы. Например, нам необходимо запрограммировать с помощью той или иной функции поведение стаи птиц, которые летят, предположим, на юг. У нас есть два варианта, как это сделать - программировать поведение всей стаи, что достаточно сложно или написать автономного агента для каждой отдельной птички и потом соединим эти агенты в одну программу, что очевидно будет намного проще первого варианта. Но независимо, каким путем мы пойдем, результат будет один и тот же.

Давайте же подробнее обсудим моделирование полета стаи птиц, которое я описывала выше. Принцип такого моделирования придумал Крейг Рейдоналдс. С помощью такого способа моделирования Крейг создал модели поведения стаек рыб и птичьей стаи и на их основе воссоздал явления, что происходят в природе, которыми можно управлять на уровне flash-игр.

**§ 2. Принцип моделирования К.Рейдоналдса**

Итак, сейчас мы рассмотрим как же Крейг Рэйдоналдс строил модель поведения стаи птиц. Как и говорилось выше, он разбил всю стаю на отдельных особей и для каждой написал своего агента, который включал в себя три параметра поведения (рис.1) :

* Разделение -

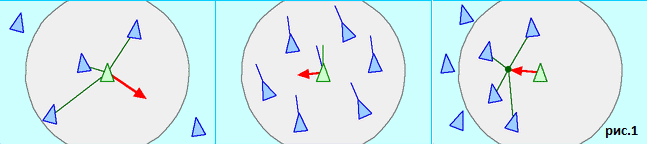
все особи стараются находиться на каком-то среднем расстоянии друг от друга, а т.е. не образуется отдельных групп.

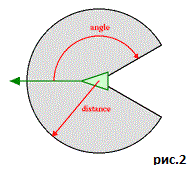
* Выравнивание -

направление вектора каждой особи стремиться к какому-то среднему направлению всех векторов.

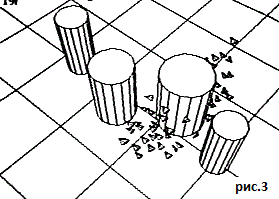
* Сплоченность -

особь старается находиться на среднем расстоянии от других особей, а т.е не происходит некой раздробленности. Является обратной функцией для разделения.(Рис. 1 )



Каждый отдельный объект, в программном коде которого описываются эти три параметра, Крейг назвал бойдом. Бойд, который является автономным агентом, соответственно, обладает свойствами агентов, а в данном конкретном случае - ограниченным участком восприятия информации. Этот участок определяется углом,измеряемом в направлении движения бойда, и расстоянием, отмеренным от центра бойда ( angle и distance соответственно) (рис.2).

Далее, после обозначения в программном коде трех параметров (рис.1) и участка восприятия и написания агента для каждой птицы, Крейг начинает “учить” бойды избегать препятствия и искать решения. Это уже являлось некой немного усложненной поведенческой моделью. (рис.3)



Собственно, именно по такому принципу, который подсказал нам Крейг Рэйнольдс, далее и будет создана программа с автономными агентами,на основе которой мы будем проводить исследование, с которым мы познакомимся далее.