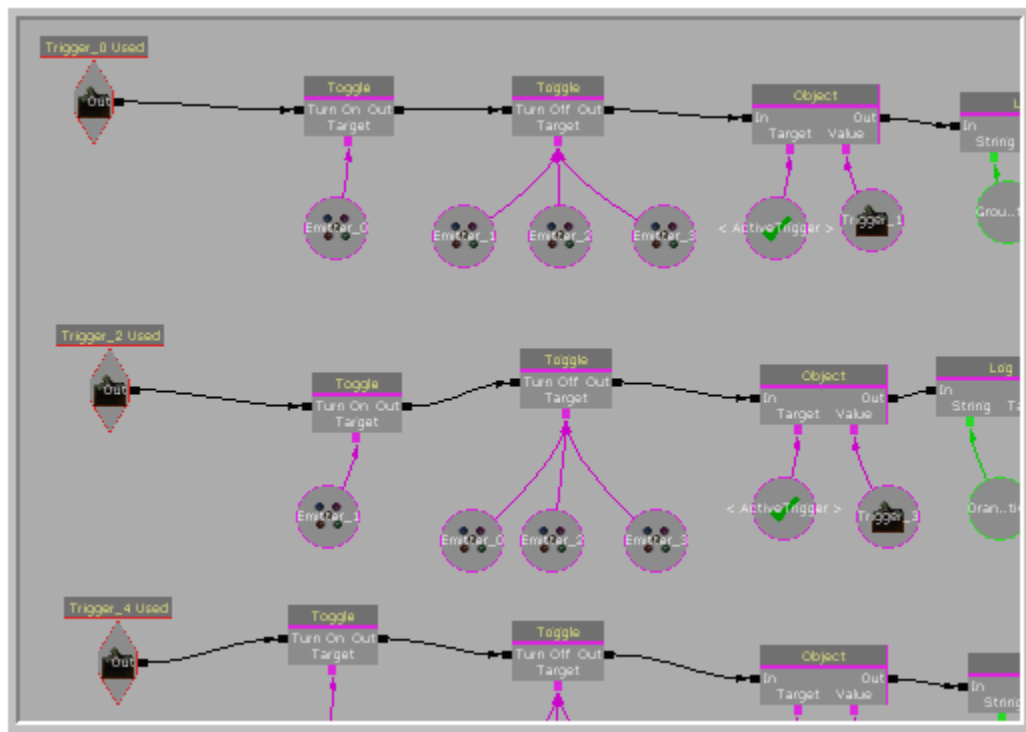


## Kismet Tutorial-1 Introduction to Kismet

### 1. What is Kismet?

Kismet is a visual interface used to help non-programmers control the interactions within a simulation which would otherwise be done by programmers using a programming language such as C++ or Java.



Example of a Kismet Program.

### 2. What does Kismet do?

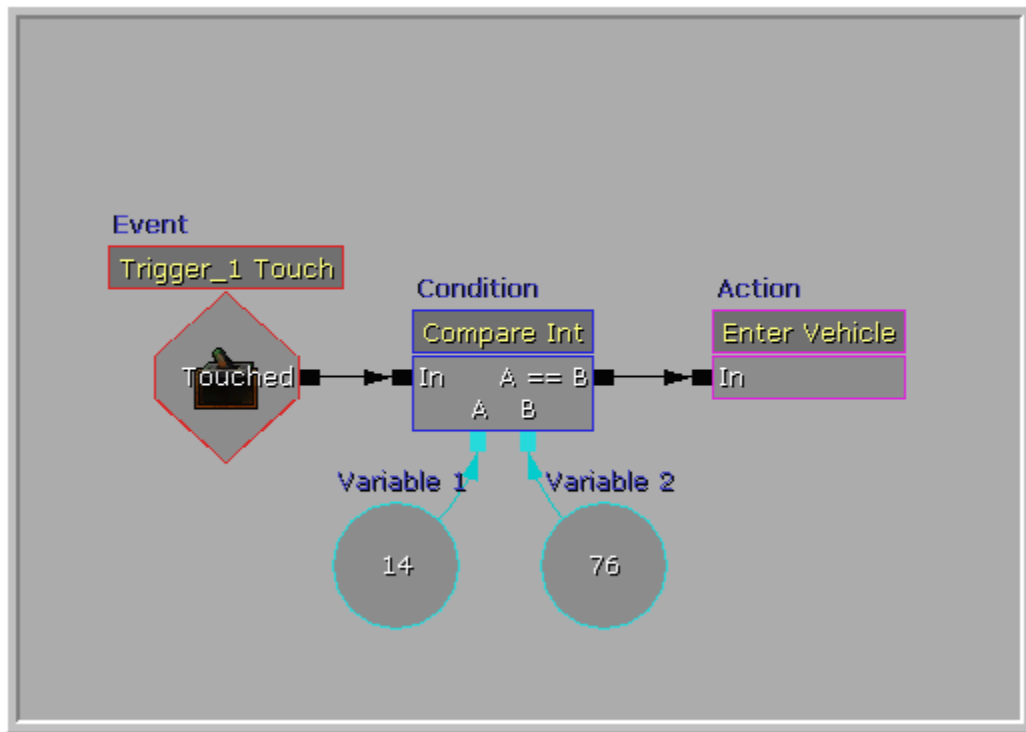
Kismet presents pictures of blocks (called *Sequence Objects*) which represent program code that preforms programming functions such as events, actions, conditions, or variable retrieving and storage.

### 3. How do I use Kismet?

Kismet is used by graphically connecting together functional blocks (*Sequence Objects*) into a sequence of functions and variables to preform complex simulation interactions.

### 4. What is a Simple Example of Kismet?

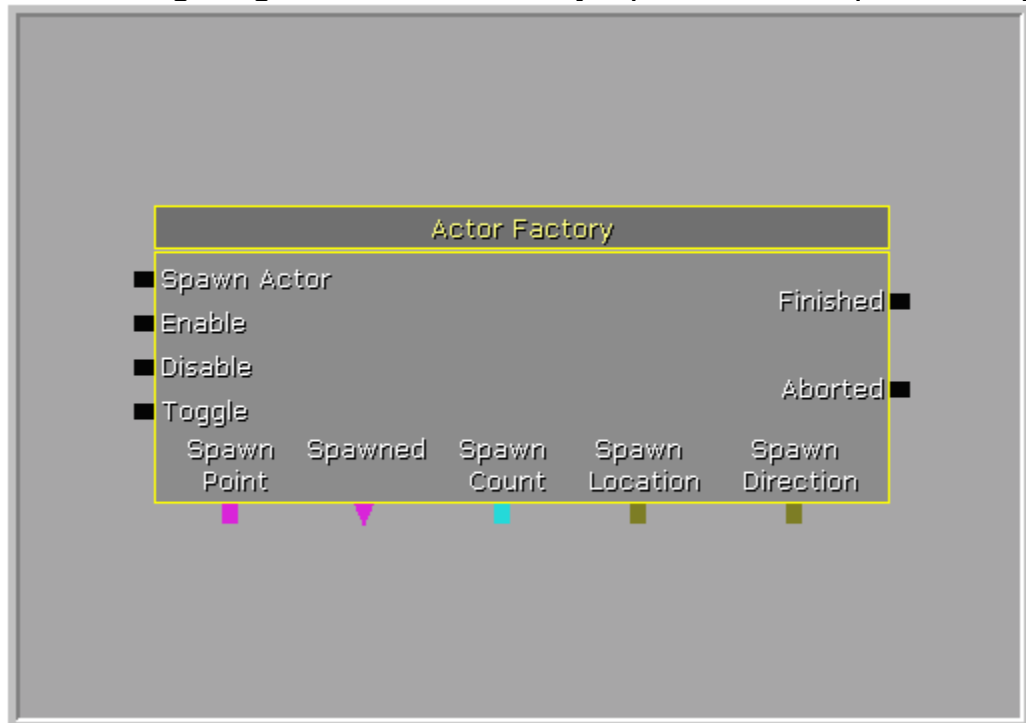
A simple example of Kismet would be two or more sequence objects connected together to form a functional sequence.



Connecting sequence objects in Kismet

### 5. How Does a Sequence Object Work?

The following diagram shows the major parts of a Sequence Object.



Left-side is *Input*, right-side *Output*, bottom are connections to *Variables*.

## 6. What are the Different Kinds of Sequence Objects?

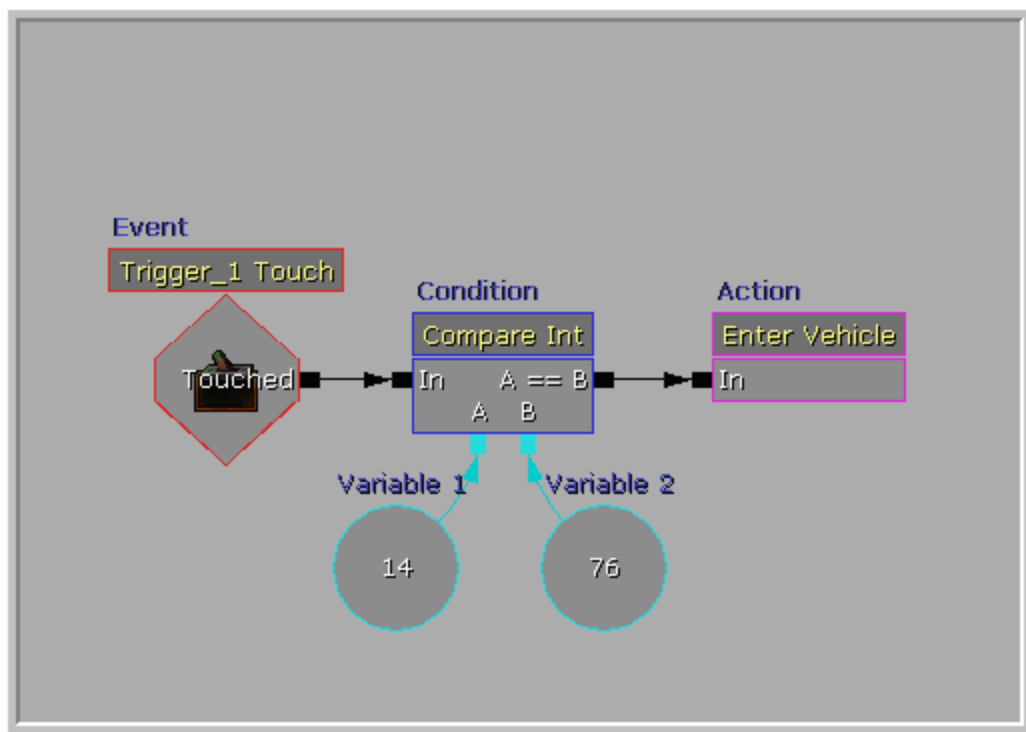
Sequence Objects fit into one of four categories:

A. **Events** – An event sends a message from the simulation to a Kismet sequence – Shown with red diamonds.

B. **Actions** – An action sends a message from a Kismet sequence to the simulation – Shown with rectangles.

C. **Conditions** – A condition controls the flow of sequences within a Kismet Sequence – Shown with rectangles.

D. **Variables** – A variable is a holding place for information within a Kismet Sequence – Shown with circles.



*Event sequenced to a Condition with two Variables sequenced to an Action.*