

# Programando Robots con APIs

## COMO USAR APIS PARA CONSTRUIR ROBOTS

Obradoiro RTB - Sergio Padín Varela

1

O que vamos a aprender hoxe e cómo facer rápida e fácilmente un robot.

Non fai falla ter moitísimos coñecementos de programación para facelo, mesmo un alumno de primeiro ten coñecementos de sobra para crear o seu robot, o único que hai que ter é imaxinación.

# Programando Robots con APIs

- Esencial para programar un robot...
- Qué é unha API?
- Por qué usar unha API?
- Cómo programar usando unha API.
- Exemplo práctico.



Veremos ás nocións básicas que hai que ter (que non son moitas) para programar un robot

Veremos a grandes rasgos qué é unha API, cómo funciona, e tamén veremos cómo usalas poñendo como exemplo a API de C de Ruben ou a API en Java de Sergio Padín (eu) que básicamente ten o mesmo funcionamento. Polo que o código que se escriba para un robot poderíase por noutro facilmente.

## Esencial para programar un robot...

- Coñecer unha linguaxe de programación?
  - *Si.*
- Nivel de coñecemento desa linguaxe?
  - *De 0 a 10, con 1 abonda e sobra. Só se necesitan coñecer as estruturas básicas: (if-then-else, function)*
- Non é esencial...
  - *Coñecer en detalle como funciona o RealTimeBattle nin como envía e recibe mensaxes dos robots*

Obviamente é esencial coñecer unha linguaxe de programación porque de momento aínda non se fixo ningunha API de programación gráfica para aqueles que non saiba nada de programación.

Non se necesita un gran nivel de programación, só hai que saber facer ifs e algunha que outra función tampouco quedaría mal no código do robot.

Non é necesario coñecer practicamente nada do RealTimeBattle, non hai que saber como se intercambian mensaxes entre o RealTimeBattle e os Robots, o único que hai que saber e executar o realtimebattle e facer partidas para probar que os Robots funcionan

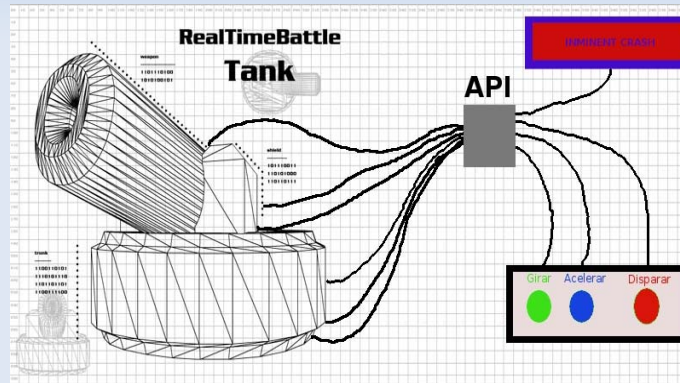
# Qué é unha API? (1)

- Qué é?:
  - *Unha API é un conxunto de módulos que ofrecen ó programador unhas funcionalidades, abstraendo todo o proceso de cómo se conseguen levar a cabo*
- Resumindo:
  - *O programador non necesita saber cómo funcionan as cousas, só necesita saber usalas.*

Unha API básicamente o que fai é o traballo sucio, e ó programador abstráeo de toda a programación que hai por embaixo, polo que o programador do robot o único que ten que coñecer son aquelas funcións que a API lle ofrece para controlar o robot.

## Qué é unha API? (2)

- Por exemplo:
  - *O condutor dun automóvil non necesita coñecer cómo fai o motor para acelerar cando se pisa o acelerador, simplemente ten que saber que se pisa o pedal o automóvil acelera.*



Obradoiro RTB - Sergio Padín Varela

5

Como o condutor dun automóvil non necesita coñecer cómo fai o motor para acelerar cando se pisa o acelerador, simplemente ten que saber que se pisa o pedal o automóvil acelera, o programador do robot non necesita saber como lle hai que enviar as ordes ó robot, simplemente se ten que preocupar de programar o comportamento deste e responder axeitadamente ante eventos que se reciba o robot.

# Por qué usar unha API?

- **Vantaxes:**
  - *Non se hai que preocupar de cómo conseguir as mensaxes do RTB nin de cómo enviarllas.*
  - *Queda moi clara a diferenza entre o que é o comportamento do robot (que é o que realmente importa) e "núcleo" do robot (API)*
- **Desvantaxes:**
  - *Non se aprende tanto como rompendo a cabeza e preguntándose por qué non funcionan as cousas*

Básicamente, por unha razón moi sinxela, porque é moito máis sinxelo e claro. Non se hai que andar preocupando mais que de programarlle ó robot una bo comportamento para que gañe as batallas.

## Cómo programar usando unha API (1)

- **Paso 1:**
  - *Elixir a linguaxe de programación que se prefira.*
- **Paso 2:**
  - *Buscar a API (p.e. en [rtb.belay.es](http://rtb.belay.es)).*
  - *Se non se encontra API volver ó paso 1.*
- **Paso 3:**
  - *Buscar documentación ou similar para saber qué funcións da API podemos usar e o que fan.*

Primeiro elíxese aquela linguaxe de programación da que se coñeza polo menos a sintaxe. Logo búscase unha API coa que programar nesa linguaxe, pódense encontrar googleando un pouco ou mirando na páxina do RTB FIC.

Cun pouco de sorte tamén se encontrará algo de documentación coa que comezar. Se non se encontrara nin API nin documentación hai dúas vías a seguir, a primeira e máis sinxela é elixir outra linguaxe, e a segunda require xa máis coñecementos de programación e consiste en programar o robot sin API ou para facilitarlle a vida a futuros programadores, xa postos facer tamén a API.

## Cómo programar usando unha API (2)

- Paso 4:
  - *Pensar o comportamento que se quere para o robot.*
- Paso 5:
  - *Implementar ese comportamento.*
- Paso 6:
  - *Probalo e volver ó paso 4.*

Os seguintes pasos non teñen ciencia ningunha, trátase de pensar o que se quere que faga o robot ante determinados eventos e plasmalo mediante sentencias de control de robot ou outra loxiga de IA que se lle ocorra o programador.

Por último só queda probar o robot no RTB e como sempre volver a modificar o comportamento xa que nunca se consegue se comporte como un quere.



## Exemplo Prático (1)

- C ou Java?
  - **C: Abrir Editor de texto**
  - **Java: Abrir Eclipse ou Netbeans (ou editor de texto)**
- Comezamos.
  - ***Abrimos unha plantilla para comezar a poñerlle comportamento ó robot. Nas APIs de C e Java isto faise dentro da función/método "radar".***

Da igual a linguaxe de programación, tanto se podería facer en C como en Java ou noutra linguaxe, aínda que por simplicidade vámolos a facer en C que é case de dominio común (supoño) mentres que Java e Orientación a Obxectos non o ten porque saber todo o mundo.

Para comezar vamos a ver un par de plantillas en C e en Java, que se poden copiar porque son curtas e pódense modificar de para proporcionar un comportamento máis ou menos decente ó robot.

## Exemplo Prático (2)

### **C:**

```
#include "../rtbapi.h"
#include <stdio.h>
void initialize(int i){
    RobotOption( SEND_ROTATION_REACHED,1);
    if (i==1) {
        Name("Example Robot");
        Colour(0xffaaaa,0xaaaaaff);
    }
}

void radar(double distance, object_type observed,double angle)
{
    //O comportamento do robot vai aqui
}

int main() {
    Initialize=initialize;
    Radar=radar;
    Scanner();
    return 1;
}
```

### **Java:**

```
import rtbapi.*;
public class TestRobot extends Robot {
    @Override public void Initialize(int option){
        Name("Example Robot");
    }

    @Override public void Radar(double distance, int
    object_type, double radar_angle){
        //O comportamento do robot vai aqui
    }

    public static void main(String[] args) {
        TestRobot t = new TestRobot();
        t.go();
    }
}
```

Aquí estan as plantillas para comezar a facer o robot, hai que ter en conta que para utilizar a plantilla de C fai falla a API de C e para usar a de Java fai falla a de Java que tendes dispoñibles na páxina [rtb.belay.es](http://rtb.belay.es)

Preferiblemente vamos a utilizar a de C para non ter que explicar como funciona a de Java, o que queira utilizar Java podeme preguntar despois do obradoiro e respondereille encantado.

## Exemplo Prático (3)

**AGORA  
A PROGRAMAR!!!**

# FIN

**Grazas por asistir!**  
**e**  
**Boa sorte no Torneo!**

**Máis información:**

- Páxina RTB FIC (<http://rtb.belay.es>)
- Páxina proxecto RealTimeBattle (<http://realtimebattle.sourceforge.net>)
- Contacto Sergio Padin Varela: [ingspv01\(at\)gmail\(dot\)com](mailto:ingspv01(at)gmail(dot)com)