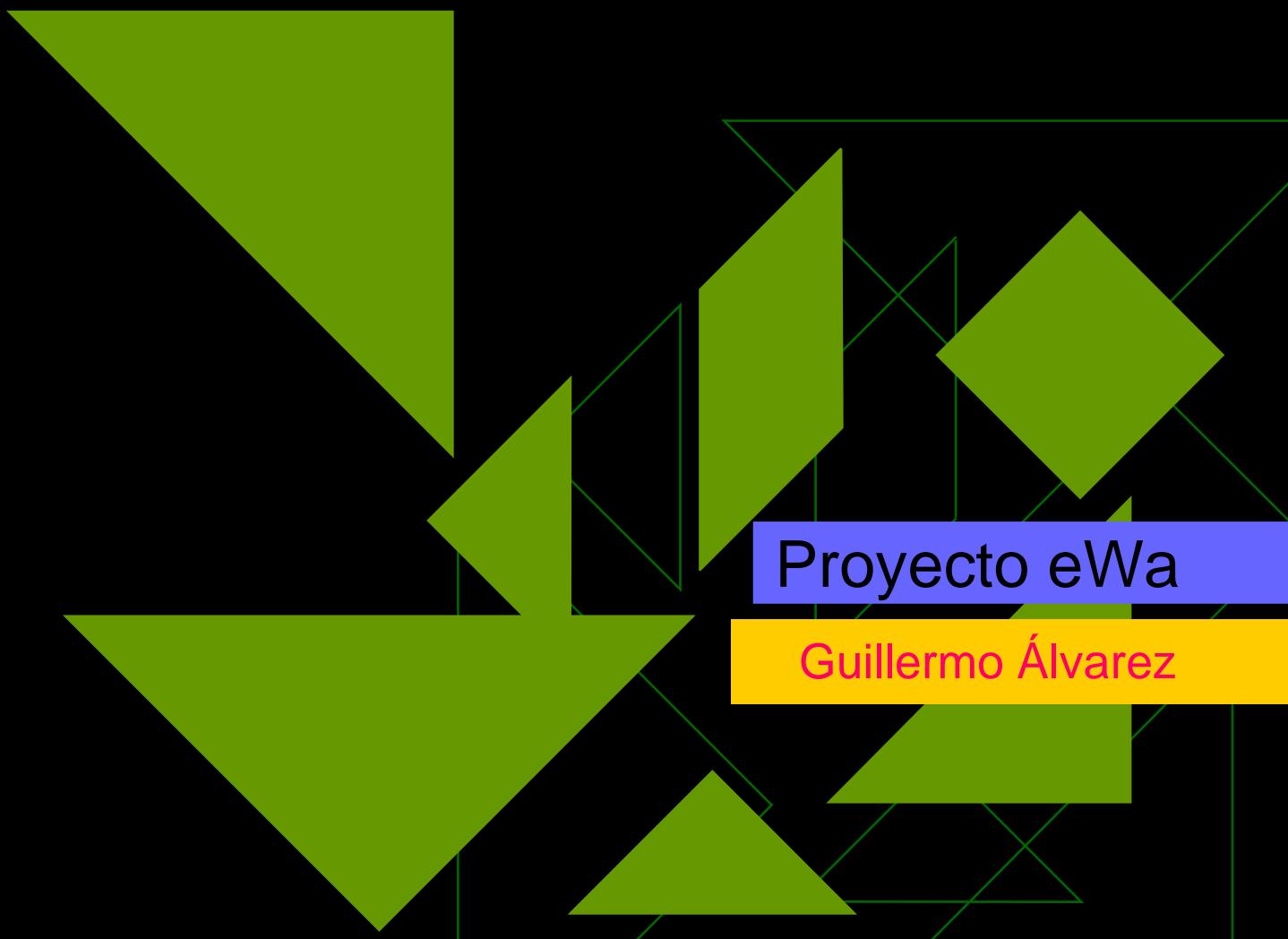


Toma de contacto con eWadrones



Proyecto eWa

Guillermo Álvarez

Temas a tratar

- ◆ Problemas de la vida real
- ◆ Solución basada en software
- ◆ Hipótesis y restricciones actuales
- ◆ Información general de la solución
- ◆ Entorno de ejecución
- ◆ Entorno de desarrollo y pruebas
- ◆ Dudas, sugerencias... y ¡ejemplos!

Problemas de la vida real

Propuestos en las actividades y en las bases de la competición

- ◆ Operaciones básicas con el dron:
 - Despegar y aterrizar
 - Ascenso/descenso, alabeo, cabeceo, y rotación... o combinación de ellas
- ◆ Reconocimiento de señales
 - Procesamiento de imágenes (visión artificial)
 - ❖ Imágenes QR
 - ❖ Detección de colores, objetos, gestos, personas, etc.
 - Procesamiento de audio
 - Procesamiento de telemetría (GPS, acelerómetro, etc.)
- ◆ Planificación de rutas
- ◆ Programación de respuestas a eventos

Soluciones basadas en software

Divide y vencerás, de abajo a arriba:

- ◆ Operaciones básicas con el dron:
 - Despegar y aterrizar
 - Ascenso/descenso, alabeo, cabeceo, y rotación... o combinación de ellas
 - Calibración y magnificación de los comandos
 - Obtención de telemetría
- ◆ Operaciones extendidas:
 - Procesamiento de datos de sensores
 - Ejecución en lote de operaciones básicas
- ◆ Infraestructura:
 - Red inalámbrica compartida/común/distribuida
 - Manejo múltiple de enjambres de drones

Hipótesis y restricciones actuales

- ◆ Conjunto de soluciones basadas en tres sistemas básicos:
 - **Sistema de Vuelo**
 - ❖ Ascenso/descenso, alabeo, cabeceo, y rotación las implementa el software de abordo...
 - **Sistema de Sensores**
 - ❖ Obtención de la altitud, batería, giroscopio, GPS, sensores de proximidad/ultrasonidos, micrófono, vídeo...
 - ❖ Configuración de LEDs, control de cámara a bordo (para girarla, alternarlas o enfocar), tren de aterrizaje móvil...
 - **Sistema de Comunicaciones**
 - ❖ Recepción y emisión de mensajes tierra-aire a través de sockets.
 - ❖ Potencialmente prescindible para algunas tareas livianas
- ◆ Por ahora, disponibles solo con ***AR.DRONE***
- ◆ Posibilidad de usar ***MAVLINK***, ***ODroid con ROS...***

Información general de la solución “eWadrones”

- ◆ www.ewadrones.com
- ◆ Se divide en tres paquetes interoperables:
 - Comunicaciones con Drones
 - ❖ Es la parte encargada de transmitir las **acciones** al dron o a los drones (**enjambre**) a través de la red
 - ❖ También obtiene los datos de los diferentes sensores
 - Visor para Operadores de Vuelo
 - ❖ Es la utilidad que se utiliza para operar el dron
 - Procesamiento de Datos
 - ❖ Son las librerías adicionales que permiten añadir funciones de tratamiento de los datos recibidos

Entorno de ejecución

Más fácil, ¡imposible!

- ◆ Disponible varios tipos de control:

- **Vuelo instrumental básico**

- ❖ Con teclas; joystick, ratón, etc. próximamente

- **Vuelo instrumental semi-automático**

- ❖ Planificar acciones manuales y repetirlas automáticamente

- **Vuelo automático**

- ❖ Desarrollo de acciones programadas

- ❖ Interpretación/Carga dinámica: no es necesario recompilar la aplicación

- ◆ Multiplataforma

- Cliente para *Windows*

- También para *Linux*, *Android*, etc. próximamente

Entorno de desarrollo y pruebas

- ◆ Editor (IDE):
 - **XML**: *Notepad++* (o cualquier otro editor)
 - **C++**: *Eclipse*, *Visual Studio*, *Notepad++*, etc.
- ◆ Pruebas:
 - En el propio **Visor para Operadores de Vuelo**:
 - ◆ En **C++**, cargando extensión (dll/so).
 - ◆ En **XML**, importando.
 - Datos
 - ◆ Vuelo real → grabar escenario para luego cargarlo
 - ❖ Con total control del aparato en caso de necesidad
 - ◆ Vuelo simulado → cargar un escenario grabado
 - ❖ Se puede pausar, rebobinar y repetir tantas veces como se desee

Dudas, sugerencias y ...

¿Está claro todo hasta aquí?



Ejemplos

Código C++:

```
#include "w_ar_drone_control_globales.h"  
{  
    ::GetArDrone().  
        getControlador().  
        despegar();  
}
```

Código XML:

```
<Despegar />
```



Ejemplos

Código C++:

```
#include "w_ar_drone_control_globales.h"  
{  
    ::GetArDrone().  
        getControlador().  
            aterrizar();  
}
```

Código XML:

```
<Aterrizar />
```

