

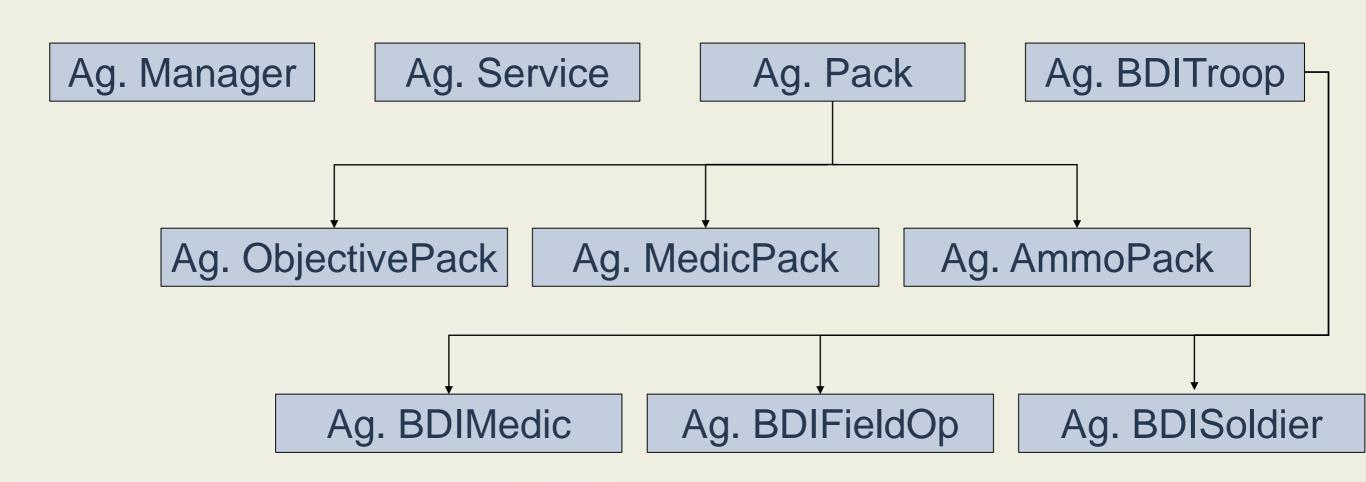
Descripción

- "Captura La Bandera" usando agentes híbridos SPADE.
- La partida se puede observar a través de visores gráficos que se pueden conectar a un servidor HTTP que se lanza en la aplicación.
- 2 equipos (**Aliados** y **Eje**) que se enfrentan en un terreno limitado durante un tiempo limitado.
- Cada equipo tiene su **base**, donde se sitúan inicialmente.
- Objetivo del juego:
 - Aliados: capturar la bandera y llevarla a su base.
 - Eje: impedir la captura (eliminando todos los aliados o si se agota el tiempo).

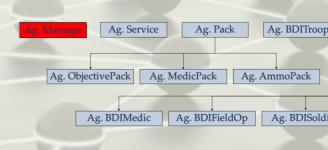
- Descripción Tropa
 Llevan un arma de fuego. Probab. de Fallar un disparo por azar:

 0.1
- Inicio de la partida con el máximo de munición (100).
- Fuego Amigo.
- Posibles roles (extensible):
 - Soldado: Sus armas hacen el doble de daño. Reciben llamadas de refuerzo.
 - Médico: crean paquetes de medicina que recuperan algo de salud a quien lo coge. Reciben llamadas de asistencia médica.
 - Operador de Campo: crean paquetes de municiones que recuperan algo de munición a quien lo coge. Reciben llamadas de recargo de municiones.

Descripción - Agentes







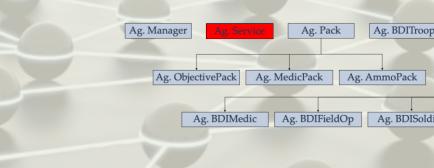
Descripción - Agentes

- Monitoriza y Gestiona la partida.
- Funciones Principales: coordinación y sincronización de los otros agentes y la aportación de información de lo que está en el campo de visión de estos.

Tareas:

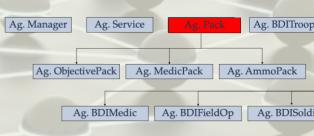
- Iniciar la partida. Mensaje de inicio a los ags. tropa con el mapa del terreno + ubicación inicial de la bandera.
- Crear el **Agente de Servicios**.
- Recibir información de cada agente tropa: posición, velocidad, orientación, salud y munición.
- Servidor de los clientes de visualización. Enviarles información de la bandera, de los agentes tropa y paquetes.





Descripción - Agentes

- Función: conocer e informar los servicios que dan las tropas.
- Al crear un Agente Tropa, se registra el servicio que ofrece aquí.
- Al morir un Agente Tropa, se informa a este agente para que sepa que ya no puede ofrecer el servicio que antes brindaba.
- Sabiendo quiénes son los Agentes Tropa activos, puede responder a peticiones de solicitud de servicios.
- Se pueden registrar y posteriormente solicitar servicios nuevos.

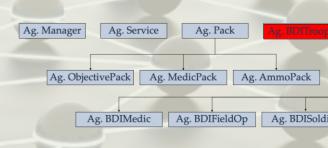


Ag. Pack

Descripción - Agentes

- Los **Agentes Pack** son los paquetes que se crean durante la partida, y pueden ser de tres tipos:
 - ObjectivePack (id = 1003): se crea al inicio del juego por el Agente Manager y representa la bandera.
 - MedicPack (id = 1001): creado por los médicos.
 - Función: incrementar la salud de quien lo reciba en 20 (0..100).
 - Auto-destruye pasado 25 segs. si no es cogido por nadie.
 - AmmoPack (id = 1002): creado por los operadores de campo.
 - Función: incrementar munición de quien lo reciba en 20 (0..100).
 - Auto-destruye pasado 25 segs. si no es cogido por nadie.





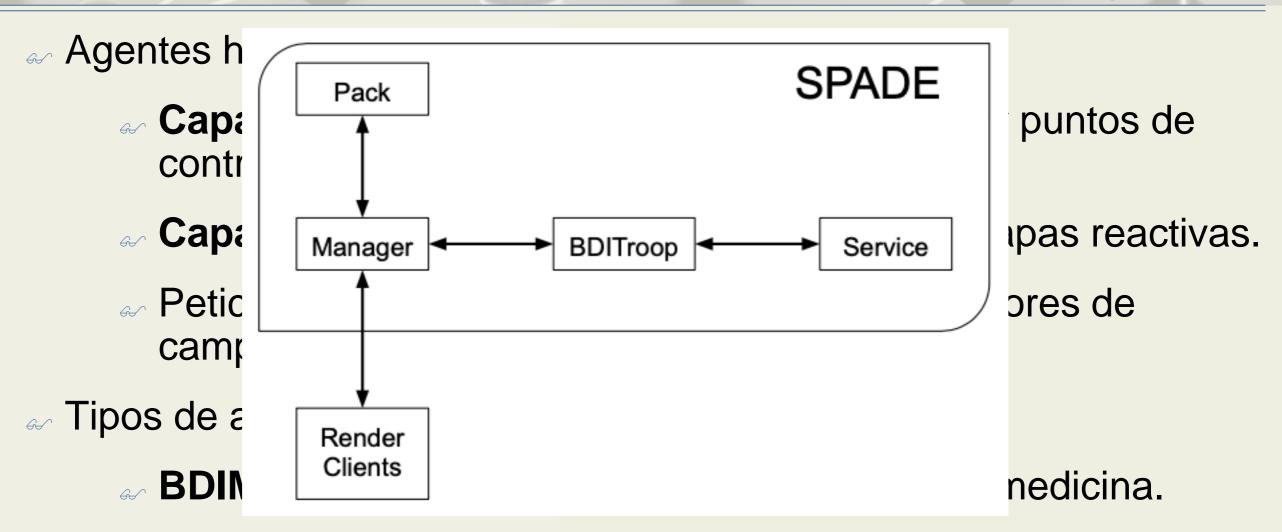
Descripción - Agentes

Agentes híbridos:

- Capa Reactiva: acciones de traslación, generar puntos de control y disparar entre otras.
- Capa Deliberativa: ASL + información de las capas reactivas.
- Peticiones al **Agente Service**: médicos, operadores de campo y soldados de su equipo disponibles.
- Tipos de agentes tropa:
 - BDIMedic: servicio médico: crear paquetes de medicina.
 - **BDIFieldOp**: servicio de recargar municiones: crear paquetes de municiones.
 - BDISoldier: servicio de refuerzo (ir a la posición de un compañero, para reforzar el ataque). Sus disparos hacen el doble de daño.

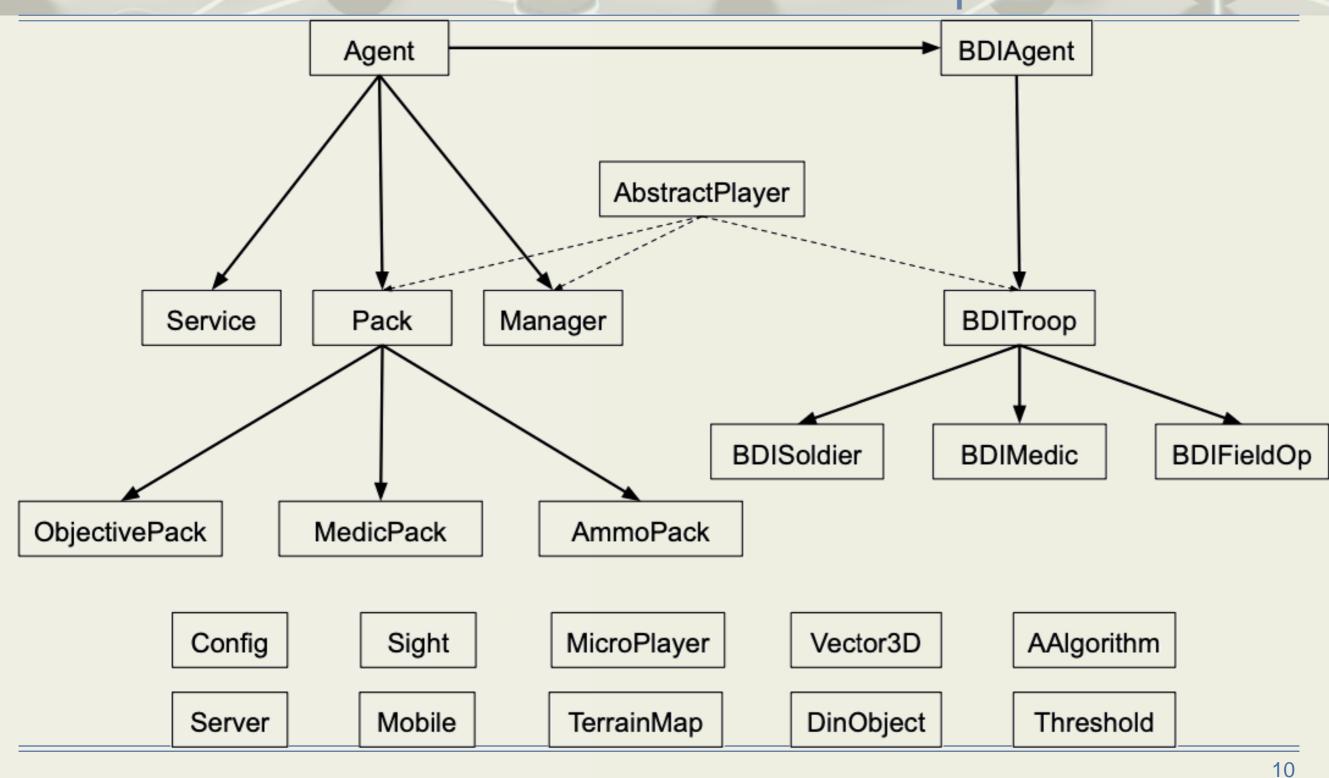


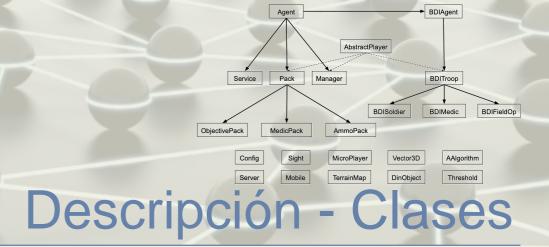
Descripción - Agentes



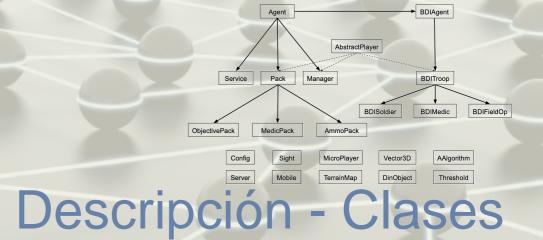
- **BDIFieldOp**: servicio de recargar municiones: crear paquetes de municiones.
- BDISoldier: servicio de refuerzo (ir a la posición de un compañero, para reforzar el ataque).

Descripción - Clases





- Agent: Es la clase base de agente proporcionada por SPADE.
- BDIAgent: agente SPADE híbrido con comportam. deliberativo.
- AbstractPlayer: Clase dependiente del juego desarrollado
 - Atributos: equipo del agente, nombre (JID) y nombre del Agente de Servicios.
 - Comportamientos: registrar y deregistrar servicios.
- MicroPlayer: Clase para el Ag. Manager para coordinar la partida.
 - Atributos de cada Agente Tropa: nombre, equipo, tipo de agente, posición, velocidad, orientación, salud, cantidad de municiones y si lleva o no la bandera.



- DinObject: Usada por el Manager para representar los paquetes.
 - Atributos: posición, nombre, equipo, tipo, si ha sido tomado o no y quién lo ha tomado en caso afirmativo.
- Sight: Usada por el Manager para saber qué paquete (*DinObject*) o tropa (*MicroPlayer*) hay en el campo de visión de qué agente tropa.
 - Información que provee: tipo de agente, equipo, salud, posición y distancia a la que se encuentra.
- Server: Pone en funcionamiento el servidor que es lanzado por el Manager para que se conecten los visores gráficos.
- Threshold: Atributos umbrales de los Agentes Tropa: máximo no de disparos por ráfaga, de municiones y de salud.
- TerrainMap: Carga el mapa del campo de batalla.



- Vector3D: Clase soporte vectorial.
 - Creación, suma, resta, normalización, módulo y productos escalares y vectoriales.
- Mobile: Uso de Vector3D para atrib. de mov. de los Ag. Tropa:
 - Atributos: posición, velocidad, orientación y destino.
 - Métodos: cálculo de posiciones, velocidades y orientaciones.
- AAlgorithm: Algoritmo A para mover Ags. Tropa por el mapa.
 - Heurística modificable (recomendación: distancia Euclidea que permite desplazamientos diagonales).
- Config: Inform. de config. del juego: ruta de los mapas por defecto, ontologías, performativas y valores de precisión.

Visualización

- Ag. Manager crea Serv. para clientes de visores gráficos a los que enviará inform. sobre el estado actual de la partida.
- Para cada Agente Tropa, se enviará la siguiente información:
 - Nombre del Agente Tropa (JID del agente).
 - Tipo (soldado, médico u operador de campo).
 - Equipo al que pertenece.
 - Salud.
 - Número de municiones que lleva.
 - Si lleva la bandera o no.
 - Vectores de posición, velocidad y orientación.
- Para cada paquete, (excepto **bandera** capturada), se enviará:
 - Nombre del paquete (JID del agente).
 - Tipo (bandera, paquete de medicina o munición).
 - Vector de posición.

Visualización

3 visores gráficos online:

- Consola texto.
- Consola pygame.
- Unity3D.

pyGOMAS

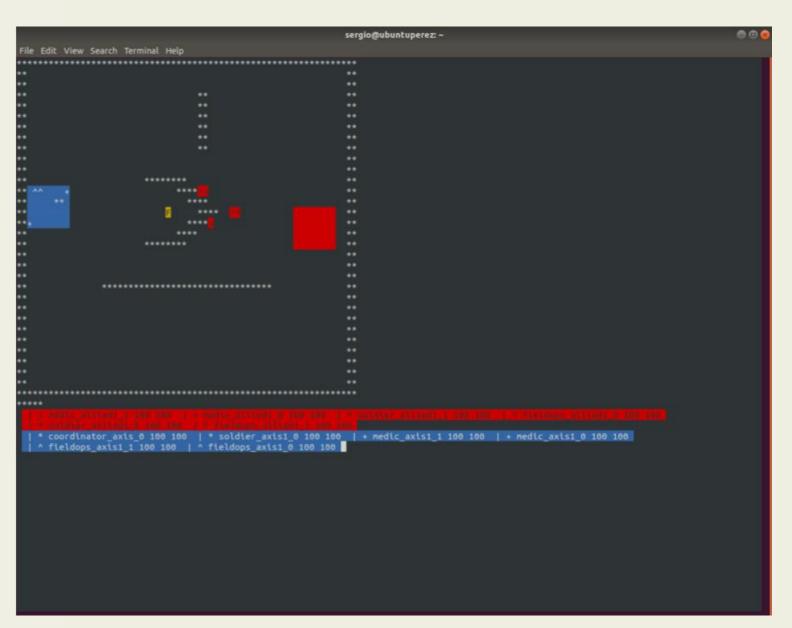
Visor offline:

Consola pygame cargando traza desde fichero.

Visualización

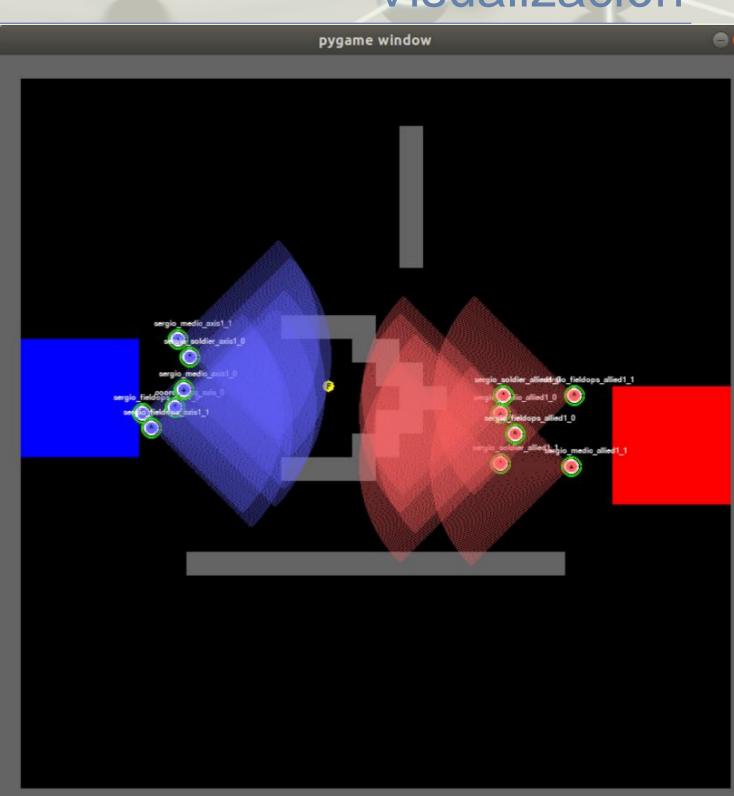
Consola texto:

- +: médicos, A: paquetes de municiones, ...
- Espacios en blanco: pos. sin obst., * donde sí.
- Base del Eje: azul, Base Aliada: roja.
- Cada Ag. Tropa:
 nombre, salud y nº de munics.



Visualización

- Consola pygame:
 - Campo de visión de cada tropa.
 - Cursores: mov. de la cámara.
 - Z, X: Zoom (in/out).
 - F: Mostrar/ocultar conos visión.



Visualización

pyGOMAS

- Permite mover la cámara.
- H: Ayuda.



Creación de Mapas

- Mapa: dos ficheros ".txt":
 - Dimensiones, ubicación de la bandera y bases.
 - Mapa de obstáculos (asteriscos).
- Movimiento:
 - Comp. Reactivo Periódico.
 - Cola de destinos: puntos del mapa a visitar para llegar a un destino.
 - Alg. JPS (Jump Point Search) para generar camino hacia su destino.
 - Velocidad y Orientación del ag. (cambia su campo de visión).

*					*
*					*
*				**	**
*				*	*
*		*		*	*
*		*		*	*
*		*		*	*
*		****	****	**	*
*		*	****		*
*		*		*	*
*	***	*****		*	*
*		*			*
*					*
*				*	*

Creencias de los Agentes Tropa (I)

- class(X): X es la clase a la que pertenece el agente:
 - NONE = 0, SOLDIER = 1, MEDIC = 2, ENGINEER = 3, FIELOPS = 4
- enemies_in_fov(ID, TYPE, ANGLE, DIST, HEALTH, [X,Y,Z]): El Ag. Tropa ha visto un enemigo con identificador ID, del tipo TYPE, a un ángulo ANGLE, a una distancia DIST, con una salud HEALTH, y en la posición [X, Y, Z].
- friends_in_fov(ID, TYPE, ANGLE, DIST, HEALTH, [X,Y,Z]): EI Ag. Tropa ha visto un compañero de equipo...
- packs_in_fov(ID, TYPE, ANGLE, DIST, HEALTH, [X,Y,Z]): El Ag. Tropa ha visto un pack ...
 - Tipos de Pack: 1000 (None), 1001 (MEDICPACK), 1002 (AMMOPACK), 1003 (FLAG).

Creencias de los Agentes Tropa (II)

- heading([X, Y, Z]): el Ag. Tropa está orientado hacia [X, Y, Z].
- health(X): X es la salud actual del agente.
- ammo(X): X es la munición actual del agente.
- base([X,Y,Z]): La base del equipo del agente está en [X, Y, Z].
- name(X): X es el nombre del agente.
- myMedics([id ...]): Lista de médicos del equipo activos.
- myFieldops([id ...]): Lista de FieldOps del equipo activos.
- myBackups([id ...]): Lista de Soldados del equipo activos.
- position([X,Y,Z]): [X, Y, Z] es la posición actual del agente.

Creencias de los Agentes Tropa (III)

- threshold_health(X): X es la salud mínima antes de lanzar una acción especial como respuesta.
- threshold_ ammo(X): X es la munición mínima antes de lanzar una acción especial como respuesta.
- threshold_shots(X): Límite máximo de disparos simultáneos.
- velocity([X,Y,Z]): [X, Y, Z] es la velocidad actual del Ag. Tropa.
- destination([X,Y,Z]): Objetivo del Ag. Tropa: [X,Y,Z].
- pack_taken(TYPE, N): Si el agente ha cogido un pack de tipo TYPE (medic o fieldops) y la cantidad a aumentar de vida/munición.
- flag_taken: Si el agente ha cogido la bandera.
- target_reached([X, Y, Z]): Se añade cuando el agente llega a su destino ([X, Y, Z]).

Acciones de los Agentes Tropa

Movimiento:

- "goto([X,Y,Z]): Establecer [X,Y,Z] como destino del ag. Pone al ag. tropa en marcha hacia dicho lugar, usando un algoritmo JPS para desplazarse por el terreno.
- .stop: Detener el mov. del ag. tropa.
- .turn(R): Modificar la orientación del ag. tropa una cantidad (pos. o neg.) R de radianes. Útil para alterar el campo de visión.
- .look_at([X,Y,Z]): Orientar el ag. tropa hacia [X,Y,Z].
- .create_control_points([X,Y,Z],D,N,C): Crear un grupo de N puntos aleatorios de control a una distancia D dada de una ubicación [X,Y,Z] en el mapa. La lista de puntos se almacena en C. Ej.: patrullar alrededor de la bandera.

Acciones de los Agentes Tropa

- Envío de mensajes al Service Agent:
 - .register_service("servicio_a"): Enviar mens. al Service Ag. para registrar un servicio especificado.
 - "get_medics: Enviar mens. al Service Ag. solicitando los médicos de su equipo.
 - .get_fieldops: Enviar un mensaje al Service Ag. solicitando los operadores de campo de su equipo.
 - .get_backups: Enviar un mensaje al Service Ag. solicitando los soldados de su equipo.
 - "get_service("servicio_a"): Enviar un mensaje al Service Ag. solicitando otro servicio (distinto de los tres anteriores) a los agentes tropa de su equipo que lo ofrezcan.

Acciones de los Agentes Tropa

- .shoot(N,[X,Y,Z]): Disparar N disparos a [X,Y,Z].
- .cure: Crear paquetes de medicina. Solo los médicos pueden realizar esta acción.
- .reload: Crear paquetes de munición. Solo los operadores de campo pueden realizar esta acción.



Implantación Instalación:

GitHub:

https://github.com/javipalanca/pygomas python setup.py install

Anaconda: Crear entorno con Python

pip install windows curses

pip install pygomas

Lanzar una partida:

Lanzar el Ag. Manager:

pygomas manager -j <login manager>@gtirouter.dsic.upv.es -m map 01 -sj <login service>@gtirouter.dsic.upv.es -np 6

Lanzar Ags. Tropa:

pygomas run -g troops.json

Lanzar Visor(es):

pygomas render --text pygomas render



Usar login como parte del nombre de TODOS los agentes

pyGOMAS

🗻 Instalación:

GitHub:

https://github.com/javipalanca/pygomas python setup.py install

- ↔ PyPi:
 - Anaconda: Crear entorno con Python 3.7
 - pip install windows curses
 - pip install pygomas

Lanzar una partida:

Lanzar el Ag. Manager:

```
pygomas manager -j
<login manager>@gtirouter.dsic.upv.es
                      -m map 01 -sj
<login service>@gtirouter.dsic.upv.es
                      -np 6
```

Lanzar Ags. Tropa:

pygomas run -g troops.json

Lanzar Visor(es):

pygomas render --text pygomas render



Implantación pyGOMAS

Fichero JSON con:

- Ags. Tropa: equipo, nombre, passwd. y rango (soldado, médico, fieldops, ...).

 Nº ags. de ese tipo a crear + fichero ASL.
- Ags. Manager y Service.
- La ayuda del juego brinda un fichero JSON para que sirva de ejemplo.
 - ≈ pygomas help run



"manager": "cmanager", byGOMAS "service": "cservice", "rank": "BDISoldier",

on:

"name": "soldier_axis1", "password": "secret",

"rank": "BDIMedic", "name": "medic_axis1",

"password": "secret",

"rank": "BDIFieldOp",

"password": "secret",

"asl": "myASL/mymedic.asl"

"name": "fieldops_axis1", "password": "secret",

"asl": "myASL/myfieldops.asl"

"rank": "mytroops.MySoldier", "name": "soldier_allied1",

"asl": "myASL/mybditroop.asl"

"rank": "mytroops.MyMedic", "name": "medic_allied1", "password": "secret",

"asl": "myASL/mymedic.asl"

"password": "secret",

"amount": 2,

"rank": "mytroops.MyFieldOp", "name": "fieldops_allied1",

"asl": "myASL/myfieldops.asl"

"asl": "myASL/mybditroop.asl"

ipo, nombre, passwd. y médico, fieldops, ...). ipo a crear + fichero

Service.

ego brinda un fichero sirva de ejemplo.

run

Implantación

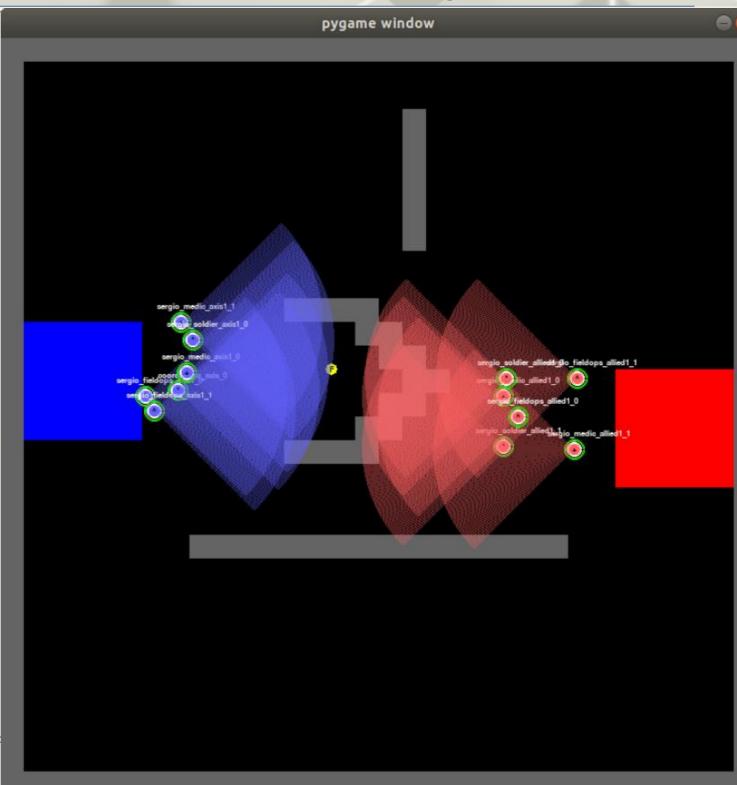
Visualización Offline:

Volcar en un fichero la inform. de la partida:

pygomas dump --log
partida.log

Visualizar la partida:

pygomas replay -game
partida.log



Implantación

pygomas_stats.txt:

Generado al acabar la partida en la carpeta del fichero JSON.

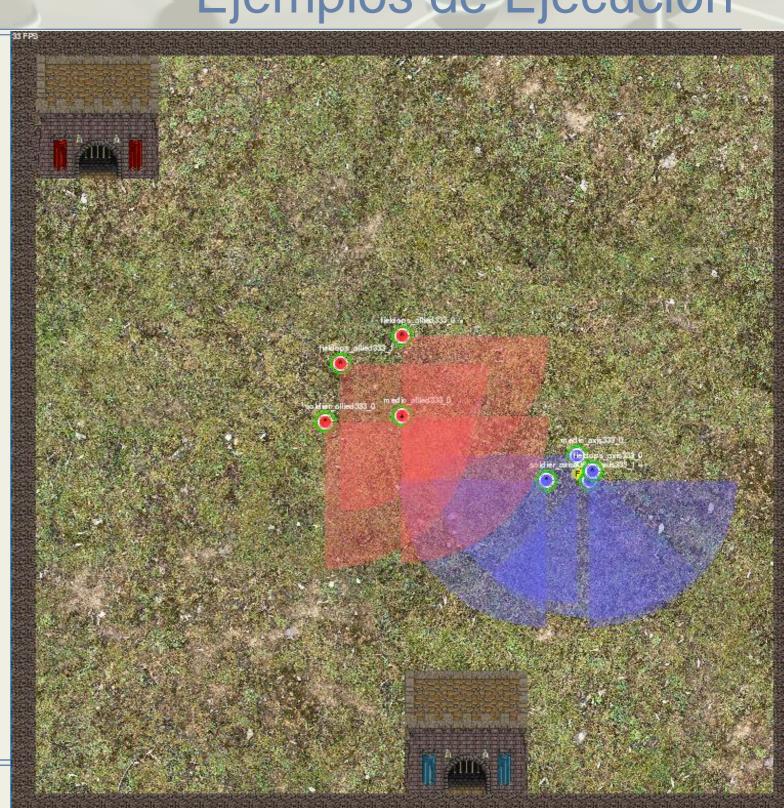
```
Winner Team: ALLIED
Duration: [439012h:14m:32s]
Statistics for ALLIED TEAM
        -GENERAL:
                * Alive:
                * Avrg. Health: 74.66666666666667
        -OBJECTIVE:
                * Times Taken: 1
                * Times Lost:
        -SHOTS:
                * EnemyHit:
                                327
                ★ TeamHit:
                * FailedHit:
                                23
                * TOTAL:
        -MEDIC PACKS:
                * Delivered:
                ★ Team Taken:
                * Enemy Taken: 0
                * Not Taken:
        -AMMO PACKS:
                * Delivered:
                * Team Taken:
                * Enemy Taken: 0
                * Not Taken:
        -EFICIENCY:
                * Medic:
                * FieldOps:
                * Army:
                                1.0
        -ANTI-EFICIENCY:
                * Medic:
                * FieldOps:
                                0
                * Army:
```

```
Statistics for AXIS TEAM
        -GENERAL:
                * Alive:
                * Avrg. Health: 0
        -OBJECTIVE:
                * Times Taken: 318
                * Times Lost:
        -SHOTS:
                                159
                * EnemyHit:
                ★ TeamHit:
                                0
                * FailedHit:
                                17
                * TOTAL:
                                176
        -MEDIC PACKS:
                * Delivered:
                ★ Team Taken:
                ⋆ Enemy Taken:
                * Not Taken:
        -AMMO PACKS:
                * Delivered:
                * Team Taken:
                ★ Enemy Taken:
                * Not Taken:
        -EFICIENCY:
                * Medic:
                * FieldOps:
                                0.9829545454545454
                * Army:
        -ANTI-EFICIENCY:
                * Medic:
                * FieldOps:
                                0
                * Army:
```

Ejemplos de Ejecución

Ejecución 1:

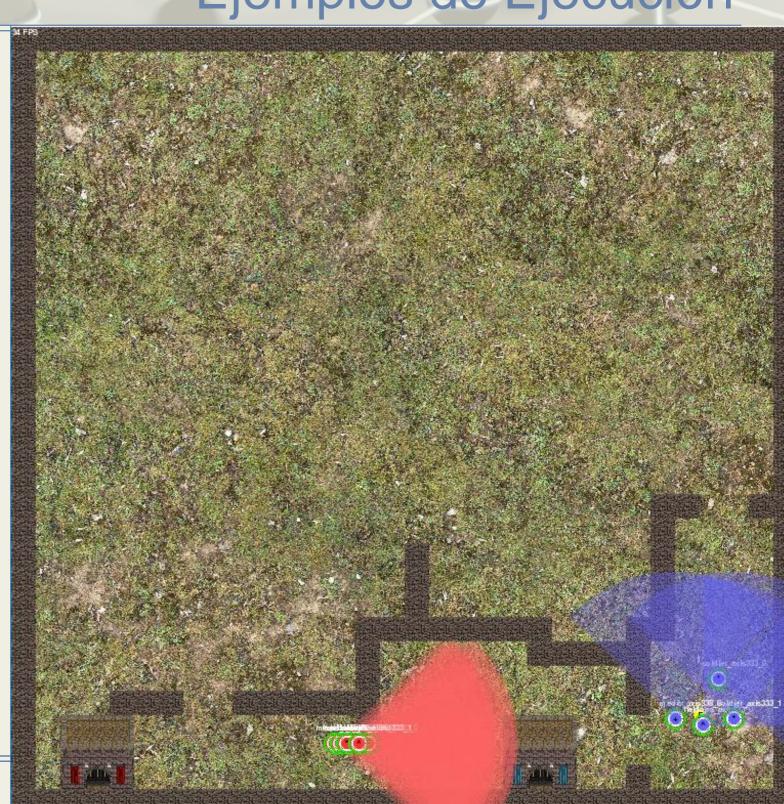
- ⊗ Soldados
 - Aliados: 1 soldado, 2 médicos y 1 operador de campo.
 - Eje: 1 soldado, 2 médicos y 1 operador de campo.



Ejemplos de Ejecución

Ejecución 2:

- - Aliados: 4 soldados, 2 médicos y 2 operadores de campo.
 - Eje: 2 soldados, 1 médico y 1 operador de campo.



Ejemplos de Ejecución

Ejecución 3:

- 20 soldados
 - Aliados: 6 soldados, 2 médicos y 2 operadores de campo.
 - Eje: 8 soldados, 1 médico y 1 operador de campo.

