

# Bucle del Juego

Programación Orientada a Objetos  
Facultad de Ingeniería - UNLPam  
2024



Universidad Nacional de La Pampa

# Bucle del Juego

- Un juego se compone de muchos subsistemas: E/S de dispositivos, renderizado, animación, detección y resolución de colisiones, redes multijugador, audio y un gran etc. Los mismos, tienen que ser consultados mientras el juego se está ejecutando.
- Los bucles de juego se pueden implementar de diferentes maneras, pero en esencia, se reducen a uno o más bucles simples, dependiendo del enfoque.



# Bucle del Juego

- Versión didáctica y muy simple

1. Inicio

2. Bucle

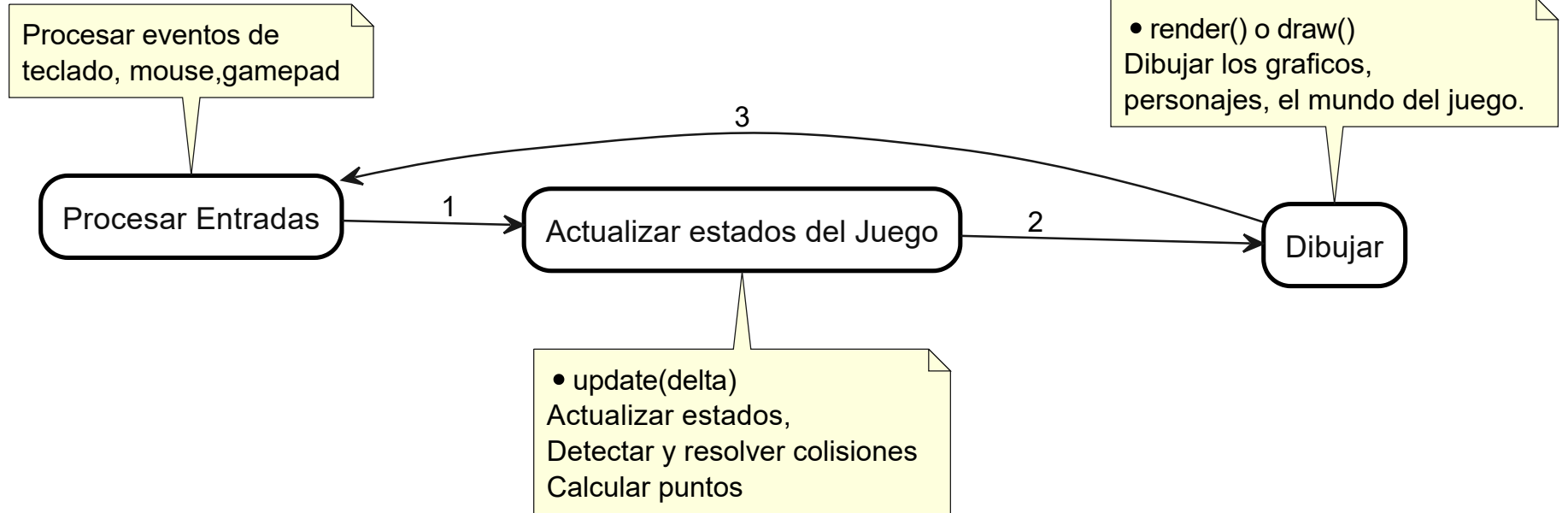
- Procesar Entradas
- Update
- Render

3. Finalizar



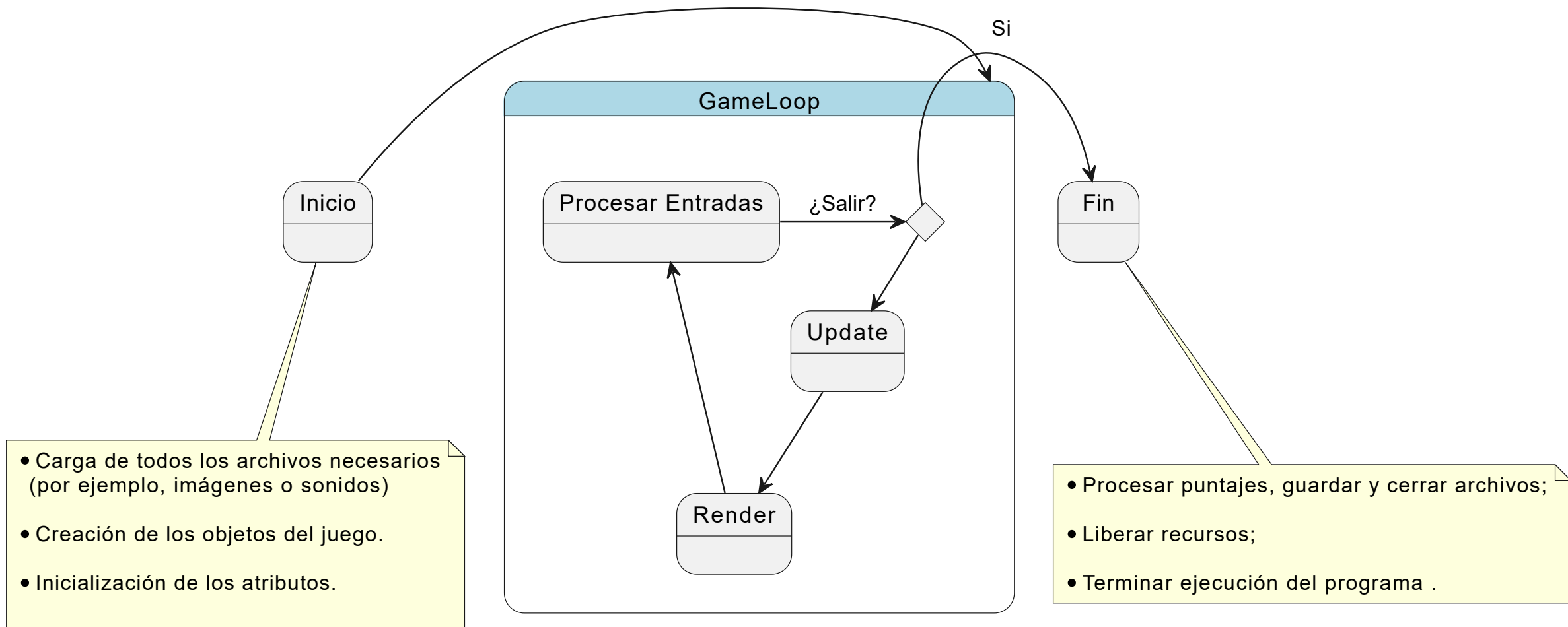
# Bucle del Juego

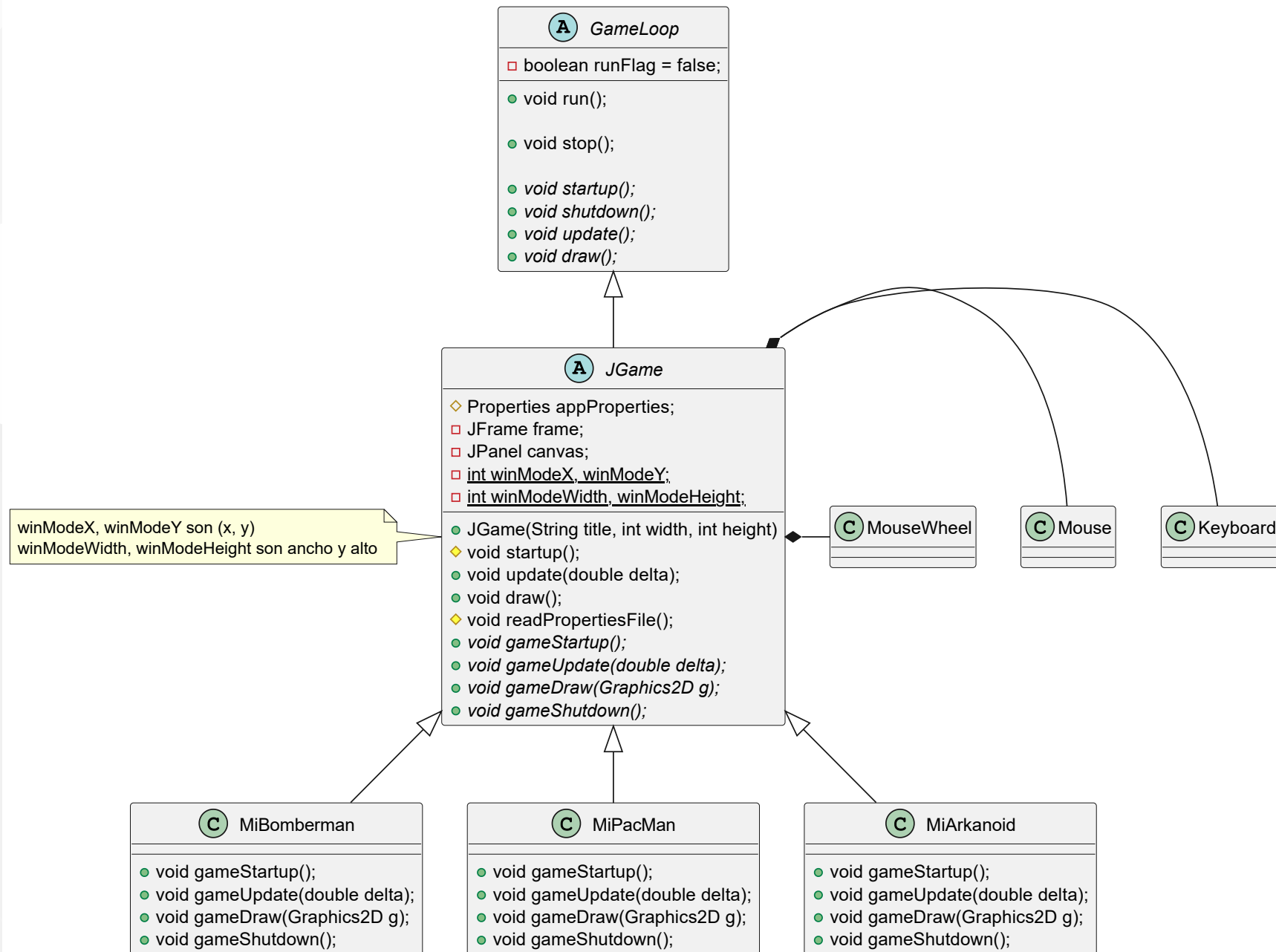
## Bucle del Juego



```
while (true)
{
    processInput();
    update();
    render();
}
```

# Bucle del Juego





# Bucle del Juego - Clase GameLoop

```
public abstract class GameLoop{
    private boolean runFlag = false;

    public void run(){
        runFlag = true;
        startup();
        while(runFlag){
            update();
            draw();
        }
        shutdown();
    }

    public void stop(){
        runFlag = false;
    }

    public abstract void startup();
    public abstract void shutdown();
    public abstract void update();
    public abstract void draw();
}
```

# Bucle del Juego - Clase JGame

```
public abstract class JGame extends GameLoop{
    atributos frame canvas keyboard mouse mouseWheel;
    public JGame(String title, int width, int height){
        .... readPropertiesFile() .... etc .....
    }
    protected void startup(){
        gameStartup();
    }
    public void update(double delta){
        ..... teclado ... mouse .... etc ....
        gameUpdate(delta);
    }
    public void draw(){
        ..... gameDraw(g); .....
    }
    protected void shutdown(){ gameShutdown(); }
    public abstract void gameStartup();
    public abstract void gameUpdate(double delta);
    public abstract void gameDraw(Graphics2D g);
    public abstract void gameShutdown();
    protected void readPropertiesFile(){
        //busca leer el archivo jgame.properties
    }
}
```



# Bucle del Juego - MiSuperJuego

```
//JGame soporta pantalla completa y ventana
public class MiSuperJuego extends JGame {

    public static void main(String[] args) {

        MiSuperJuego game = new MiSuperJuego();
        game.run(1.0 / 60.0);

        System.exit(0);
    }

    public void gameStartup() {.....}
    public void gameUpdate(double delta) {.....}
    public void gameDraw(Graphics2D g){.....}
    public void gameShutdown() {.....}
}
```

# Bucle del Juego - Conclusión

- Al optar por el bucle del juego provisto por la cátedra, los pasos a seguir son:

## 1. Crear una subclase concreta de JGame

```
public class MiSuperJuego extends JGame {.....}
```

## 2. Rellenar con código propio y la lógica del juego indicado, los 4 métodos:

```
public void gameStartup() {.....}  
public void gameUpdate(double delta) {.....}  
public void gameDraw(Graphics2D g){.....}  
public void gameShutdown() {.....}
```