Multiplayer Game Synchronization Protocol (PSJM)

Status of This Memo

This document is not an Internet Standards Track specification; it is published for informational purposes.

This document is a product of EPITECH Network Programming course. It represents information that the author believes is valuable to share with the community.

Abstract

This document specifies the Multiplayer Game Synchronization Protocol (PSJM), a simple UDP-based protocol for real-time multiplayer games. It facilitates player connections/disconnections, position synchronization, and game state updates.

Table of Contents

1. Introduction

The Multiplayer Game Synchronization Protocol (PSJM) is a simple

UDP-based protocol for real-time multiplayer games. It supports:

- Player connections/disconnections
- Position synchronization
- Game state updates

2. Packet Format

All packets have a fixed 6-byte header:

```
0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|   Magic   | ID |  Sequence  |  Length   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|          Payload              | End Flag |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Fields:
- Magic Number
- Client ID
- Sequence Number
- Length of Paylaod
- Paylod
- End flag (/r/n)

Header:
- Magic Number
- Client ID
- Sequence Number
- Length

Body:
- Payload
- End flag

3. Packet Types

Client :

Value	Name	Description
0x01	CONNEXION	Client connection request

| 0x03 | DECONNEXION | End of connection or window close |
| 0x04 | EVENT | Sent when event happens |
+--------+------------------+---------------------------------------+

Server :

+--------+------------------+---------------------------------------+
| Value | Name | Description |
+--------+------------------+---------------------------------------+
0x02	CONNEXION_ACC	Server response to connection
0x05	GAME_STATE	Game state update
0x06	SEND_MAP	Sends map state and elements
0x07	END_MAP	Signals map has ended
0x08	END_GAME	Signals a player victory
0x09	CAN_START	Client ready to start movement
+--------+------------------+---------------------------------------+

4. Packet Details

4.1 Client Details

4.1.1 CONNEXION (0x01) – Sent from client to server

Data:

- Player name (UTF-8, max 7 chars + null terminator)

4.1.2. DECONNEXION (0x03) – Client requests to disconnect to server

Data:

- Player ID (4 bytes)

4.1.3. EVENT (0x04) – Client notifies input

Data:

- Player ID (4 bytes)
- Event type (1 byte, e.g., Up, Down, Left, Right, Space)

4.2 Server Details

4.2.1 CONNEXION_ACC (0x01) – Sent from Server to Client

Data:

- Player ID (4 bytes)

4.2.2. GAME_STATE (0x05) – Server sends games state to clients

Data:

- Player ID (4 bytes)
- State (position, velocity, state...)

4.2.3. SEND_MAP (0x06) – Server sends the map to the clients

Data:
- Player ID (4 bytes)
-Map Data (json)

4.2.4. END_MAP (0x07) – Server notify the end of the map

Data:
- Player ID (4 bytes)

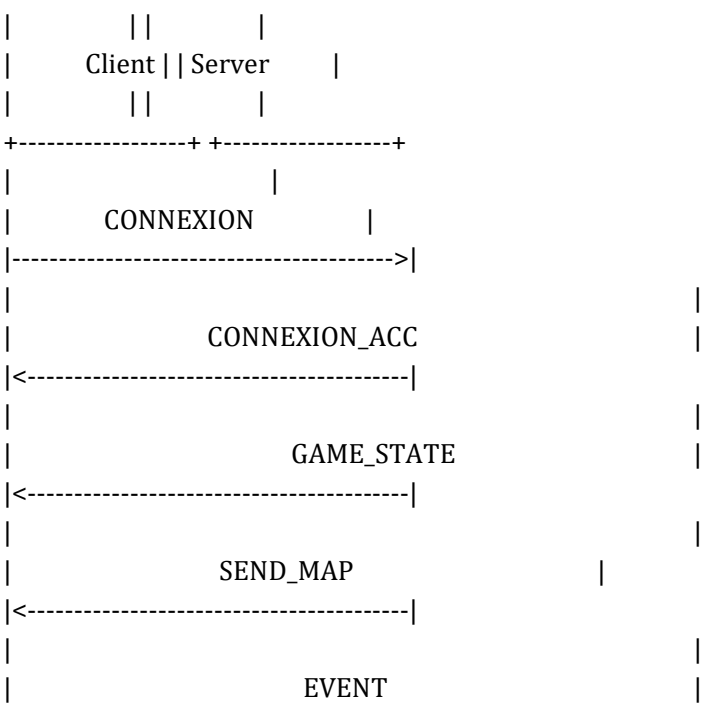
4.2.5. END_GAME (0x08) – Server notify end of game and who won

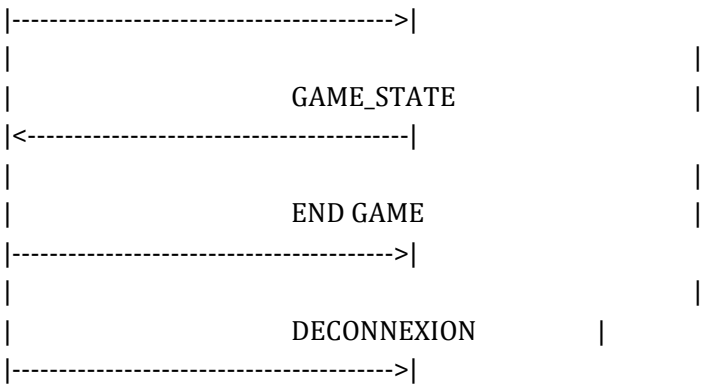
Data:
- Player ID (4 bytes)
- Player ID who won (4 bytes)

4.2.6. CAN_START (0x09) – Server tells client game can start

Data:
- Player ID (4 bytes)

5. Communication Example





6. Packet lost consideration

6.1 Tracking :

To avoid and track easier what was lost, each packet is numbered and assigned to a user so that the server can now when a package was lost.

6.2 Rollback :

The client will have an interpalation logic, so that if needed he can predict and advance until the server (the absolute truth), sends a new packet or starts reponding again.

7. Technical Considerations

- Encoding: UTF-8 text
- Number format: Network order (big-endian)
- Frequency: ?

8. Map Format Protocol

8.1. File Formatting

- File type :
  File containing the map must be a .json

8.2. Map Format

- Element type ?

8.3. Map Rendering

- One elem = ?x? pixel square

9. References

[RFC7322] Flanagan, H. and S. Ginoza, "RFC Style Guide", RFC 7322, DOI 10.17487/RFC7322, September 2014, <https://www.rfc-editor.org/info/rfc7322>.

10. Author's Address

Matisse Marsac
EPITECH
Email: matisse.marsac@epitech.eu

Marin Lamy
EPITECH
Email: Marin.lamy@epitech.eu

Eliott Tesnier
EPITECH
Email: eliott.tesnier@epitech.eu

Alban Roussee
EPITECH
Email: alban.roussee@epitech.eu