Feature overview

⏱ Highly **performant** (~3x faster[1])

📉 Significantly **reduces bandwidth** usage (~5x smaller[1])

💾 Multiple popular **compression algorithms** available (e.g LZ4, ZSTD ...)

🔽 Advanced **delta compression algorithms** to fit specific use cases (e.g for data with low variability)

🗜 Support for **range quantization** of integer types (e.g integers with limited range require less bandwidth)

📏 Support for **precision quantization** of floating points (e.g floats with limited precision require less bandwidth)

🕹 Support for **most popular platforms** (64-bit Windows/Linux/macOS, Android[2], iOS[2], WebGL[2])

[1] Compared to C# built-in BinaryWriter/BinaryReader with 100 randomly moving transforms (position, rotation, scale).

[2] Compressor API is currently not available on this platform.

Why

Modern **multiplayer games are becoming increasingly complex** which naturally increases the demand for performance in lower levels of the network stack. The serialization of objects is the cornerstone of networked games and efficiency is crucial in situations where there may be hundreds of entities in motion. **Higher frame-rates** are possible with carefully crafted and efficient implementations, ultimately leading to a more enjoyable experience for end-users. But perhaps more importantly, smart serialization and compression schemes lead to a significant **reduction in bandwidth and latency**.