Xast: Comparing relational and event-driven ECS models

Oleksandr Hnutov

December 29, 2025

Problem

Dealing with single-entity systems, like

$$\mathcal{R} \to \Delta\mathcal{R}, \text{where } \Delta\mathcal{R} \subseteq \mathcal{R},$$

e.g. system, which moves player towards X axis

$$\{\text{Player}, \text{Transform}\} \to \{\text{Transform}\},$$

is straightforward, because both for relational and event-driven systems, formal definition will be the same, so it will have the same properties. We need to define formal definitions for systems, which process 2 and more entities both for the relational ECS model (System_R) and event-driven ECS model (System_E). Systems in the event-driven model are restricted to single-pass iteration over the event sequence and may not construct auxiliary indices whose size depends on $|\,\text{Seq}(\text{Event})|$.

As an example, we will consider a simple system with two entities:

$$\begin{aligned}\textbf{EnemyFollowPlayer}\colon & \\ & \{\text{Player}, \text{Transform}\}, \\ & \{\text{Enemy}, \text{Transform}\} \\ & \to \{\text{Enemy, Transform}\}.\end{aligned}$$

Formal definition

$$\text{System}_\text{R}\colon \mathcal{R}_1 \times \mathcal{R}_2 \times \cdots \times \mathcal{R}_n \to \Delta\mathcal{R},$$

where:

1. $\mathcal{R}_i$ is a set of entities with a component set C_i,
2. $\Delta\mathcal{R}$ denotes a finite set of component updates applied to entities in $\mathcal{R}$.

$$\text{System}_\text{E}\colon \mathcal{R} \times \text{Seq}(\text{Event}) \to \Delta\mathcal{R},$$

where:

1. Event is a tuple (e, C) where e is an entity identifier and C is a finite component set, associated with e;
2. Seq(Event) denotes an ordered sequence, allowing only sequential access.

Let E be the set of enemies entities and P the set of players entities. Let $n = |E|$ and $m = |P|$.
Let $f\colon \text{Transform} \times \text{Transform} \to \text{Transform}$ be a pure function, which computes the next enemy position, moving to a player, based on player's and enemy's positions.
Let $\mathbf{choose}(P, e)$ be a deterministic selection function from players for any enemy.

Relational model system will be

$$\begin{aligned} &\text{System}_\text{R}\colon \forall e \in E\colon \\ &\qquad \text{targetP} = \mathbf{choose}(P, e) \\ &\qquad e.\text{Transform} = f(e.\text{Transform}, \text{targetP}.\text{Transform}); \end{aligned}$$

If **choose** is O(1) (e.g., the **lookupMap** function on a precomputed map), total complexity will be the following:

$$T_\text{R} = \text{O}(n + m);$$

In worst case, if **choose** need to scan all the players, total complexity will be

$$T_\text{R} = \text{O}(n \cdot m).$$

Let Event — a tuple: (Player, Transform). Event-driven model system will be

$$\begin{aligned} &\text{System}_\text{E}\colon \forall e \in E\colon \\ &\qquad \forall \texttt{ev} \in \text{Seq(Event)} \\ &\qquad\qquad e.\text{Transform} = f(e.\text{Transform}, \texttt{ev}.\text{Transform}); \end{aligned}$$

So we every time we will scan the entire sequence of events, so total complexity is

$$T_{\mathrm{E}} = \mathrm{O}(n \cdot m).$$

Theorem. Lower bound for event-driven ECS

Let E be a set of n enemy entities and P a set of m player entities. Assume a system processes enemies by iterating over a linear sequence of events $\mathrm{Seq}(P)$ and has no auxiliary indexing structure. Then any system computing a function

$$f : E \times P \to \Delta E$$

has worst-case time complexity $\Omega(n \cdot m)$.

Proof

Let $\mathrm{Seq}(P)$ be a sequence containing exactly one event for each player. By assumption, the system can only access player data by iterating through $\mathrm{Seq}(P)$.

For a fixed enemy $e \in E$, any algorithm that depends on player data must, in the worst case, inspect all m events to determine the relevant player (e.g. nearest player, target selection).

Thus, processing a single enemy requires $\Omega(m)$ time.

Since the system processes all n enemies independently, the total worst-case time complexity is $\Omega(n \cdot m)$.