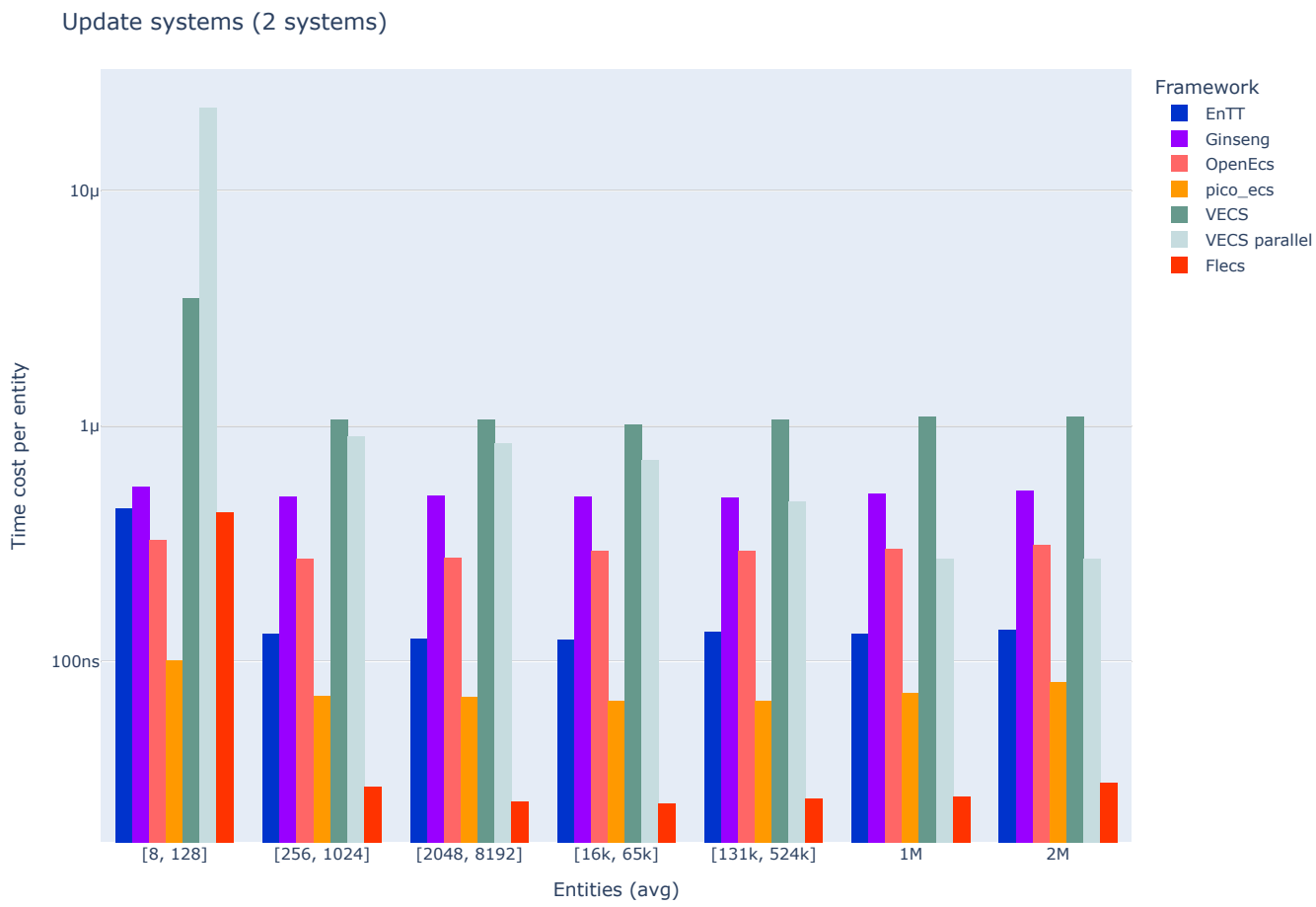
Results

TL;DR Results



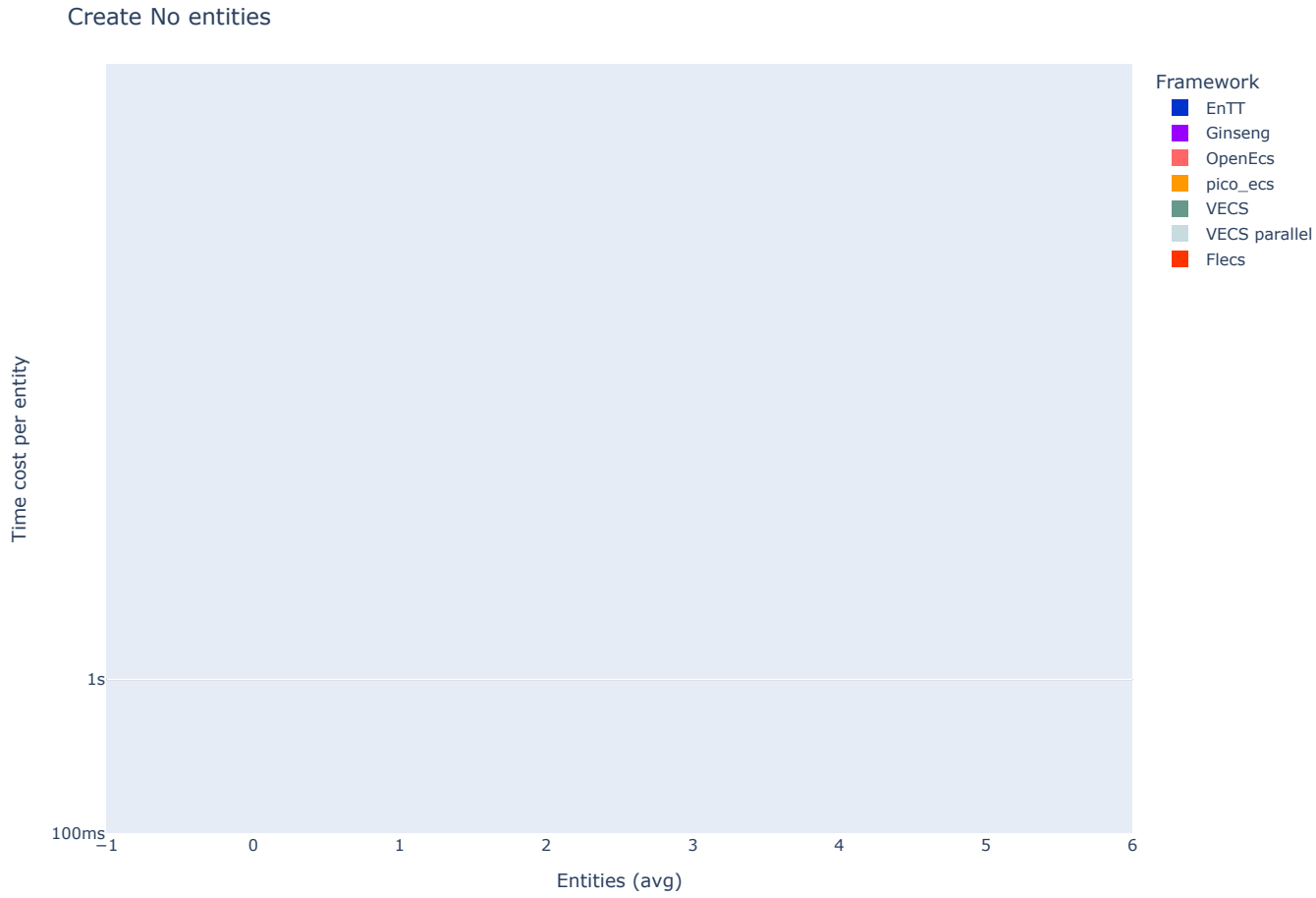
Graph shows cost per entity, tables shows total cost. lower is faster.

None

None

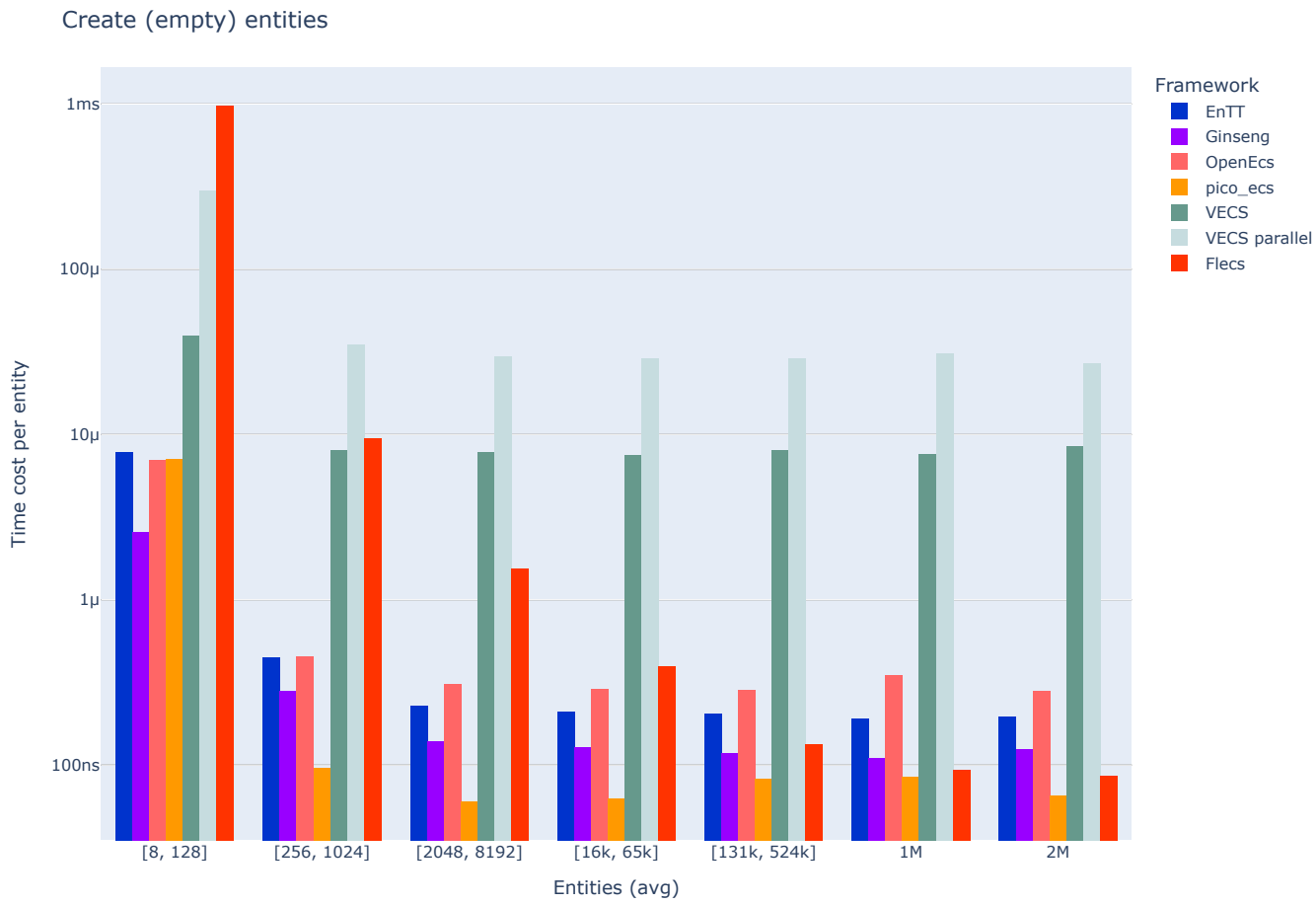
Benchmarks

Create No entities



Graph shows cost per entity, tables shows total cost. lower is faster.

Create empty entities



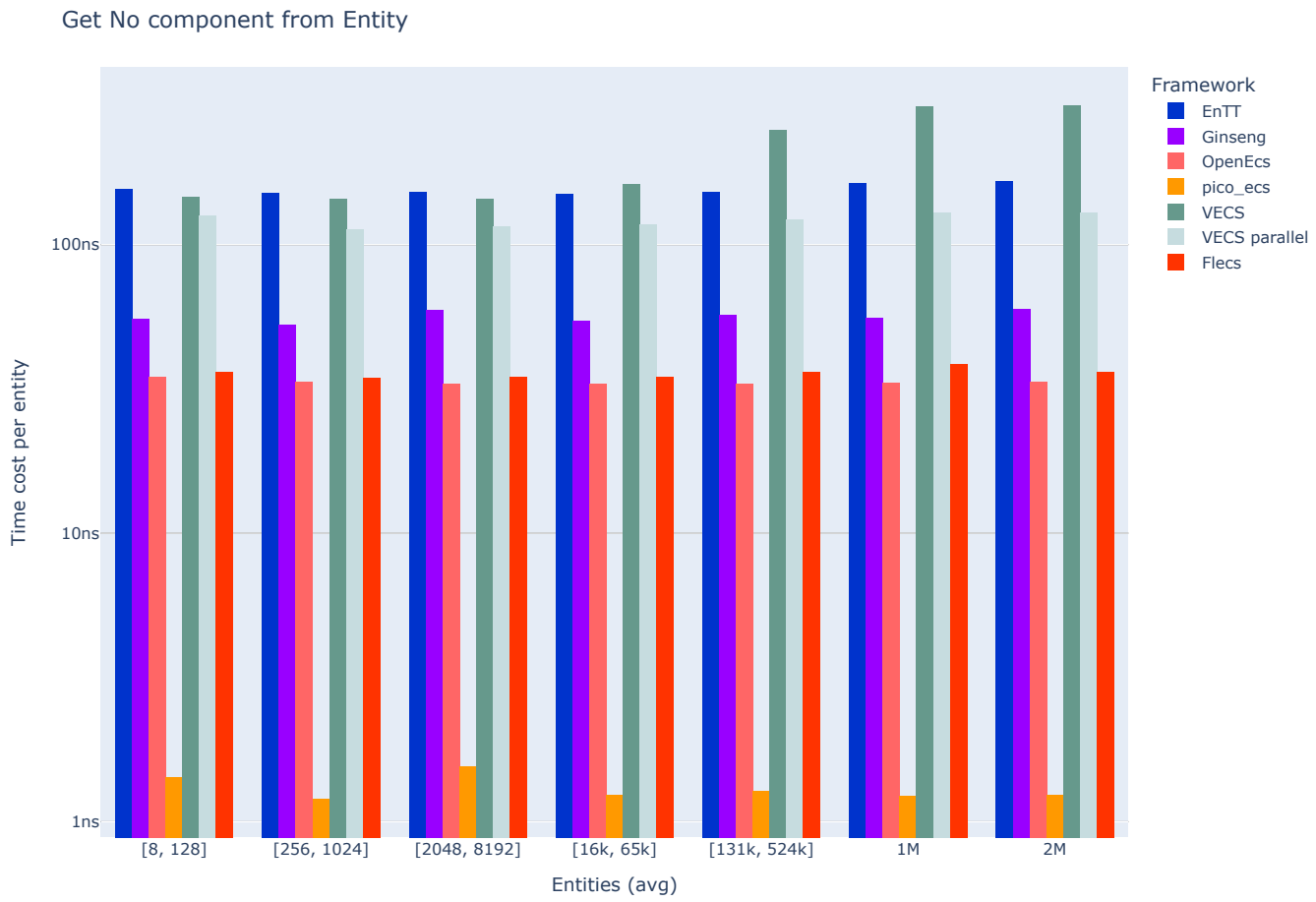
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Create 256 (empty) entities	138us	96us	129us	36us	2040us	9202us	4163us
Create ~1K (empty) entities	432us	216us	387us	60us	8118us	32049us	4203us
Create ~4K (empty) entities	1004us	543us	1345us	258us	34219us	117995us	6317us
Create ~16K (empty) entities	3217us	2136us	4674us	899us	117833us	478623us	10242us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Create ~65K (empty) entities	14ms	8ms	19ms	4ms	521ms	1857ms	13ms
Create 262K (empty) entities	54ms	30ms	73ms	22ms	2071ms	7604ms	33ms

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Create ~1M (empty) entities	198ms	114ms	363ms	88ms	8002ms	32263ms	98ms
Create ~2M (empty) entities	412ms	260ms	589ms	137ms	17669ms	56707ms	180ms

Get No component from Entity



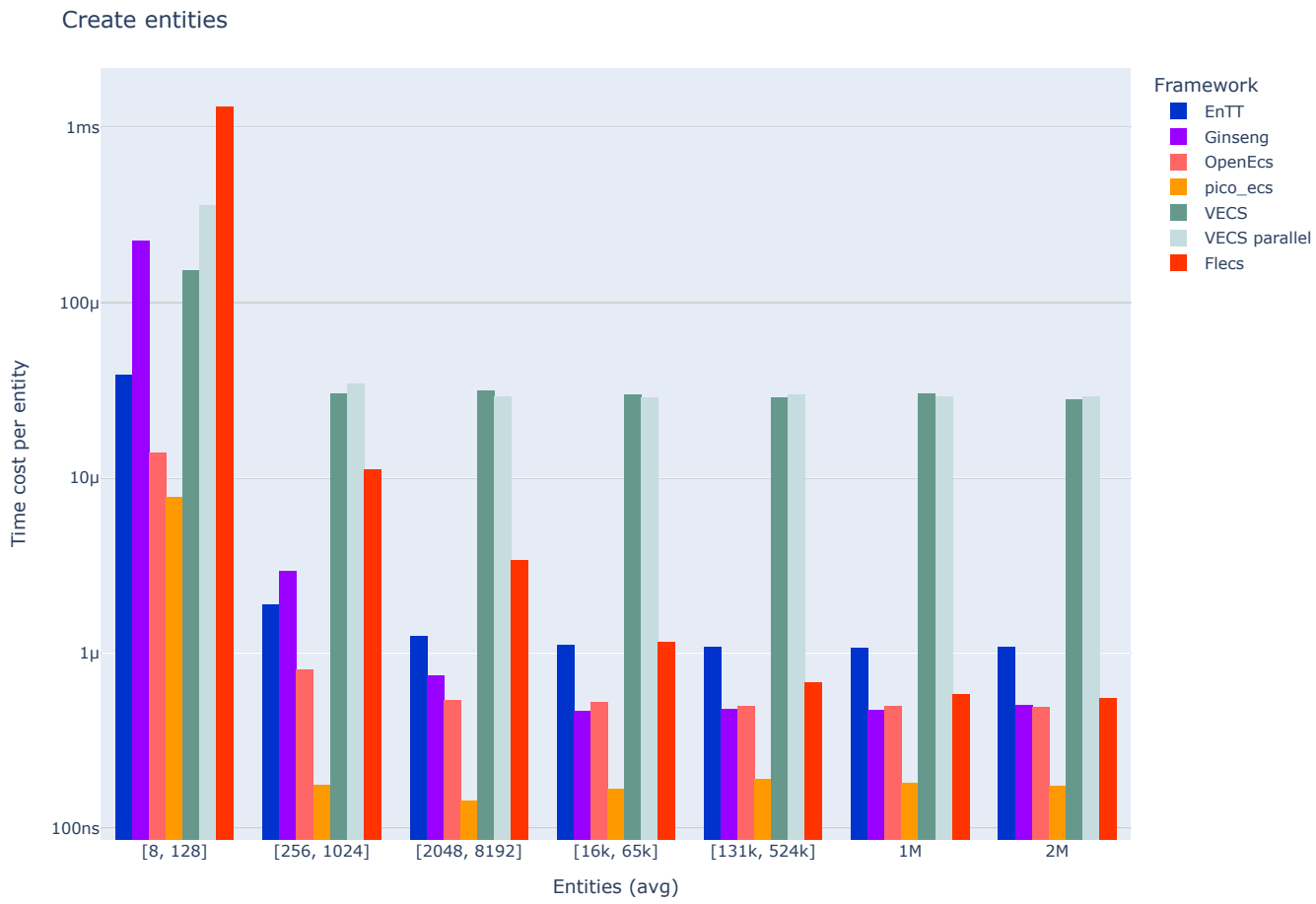
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Get No component in 256 entities	37us	13us	8us	0us	38us	28us	8us
Get No component in ~1K entities	153us	54us	34us	1us	144us	114us	35us
Get No component in ~4K entities	627us	228us	135us	7us	579us	473us	143us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Get No component in ~16K entities	2448us	891us	540us	21us	2326us	1897us	572us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Get No component in ~65K entities	9ms	3ms	2ms	0ms	12ms	7ms	2ms
Get No component in 262K entities	41ms	15ms	8ms	0ms	79ms	32ms	10ms
Get No component in ~1M entities	172ms	58ms	34ms	1ms	317ms	135ms	40ms
Get No component in ~2M entities	349ms	125ms	70ms	2ms	634ms	270ms	75ms

Create entities

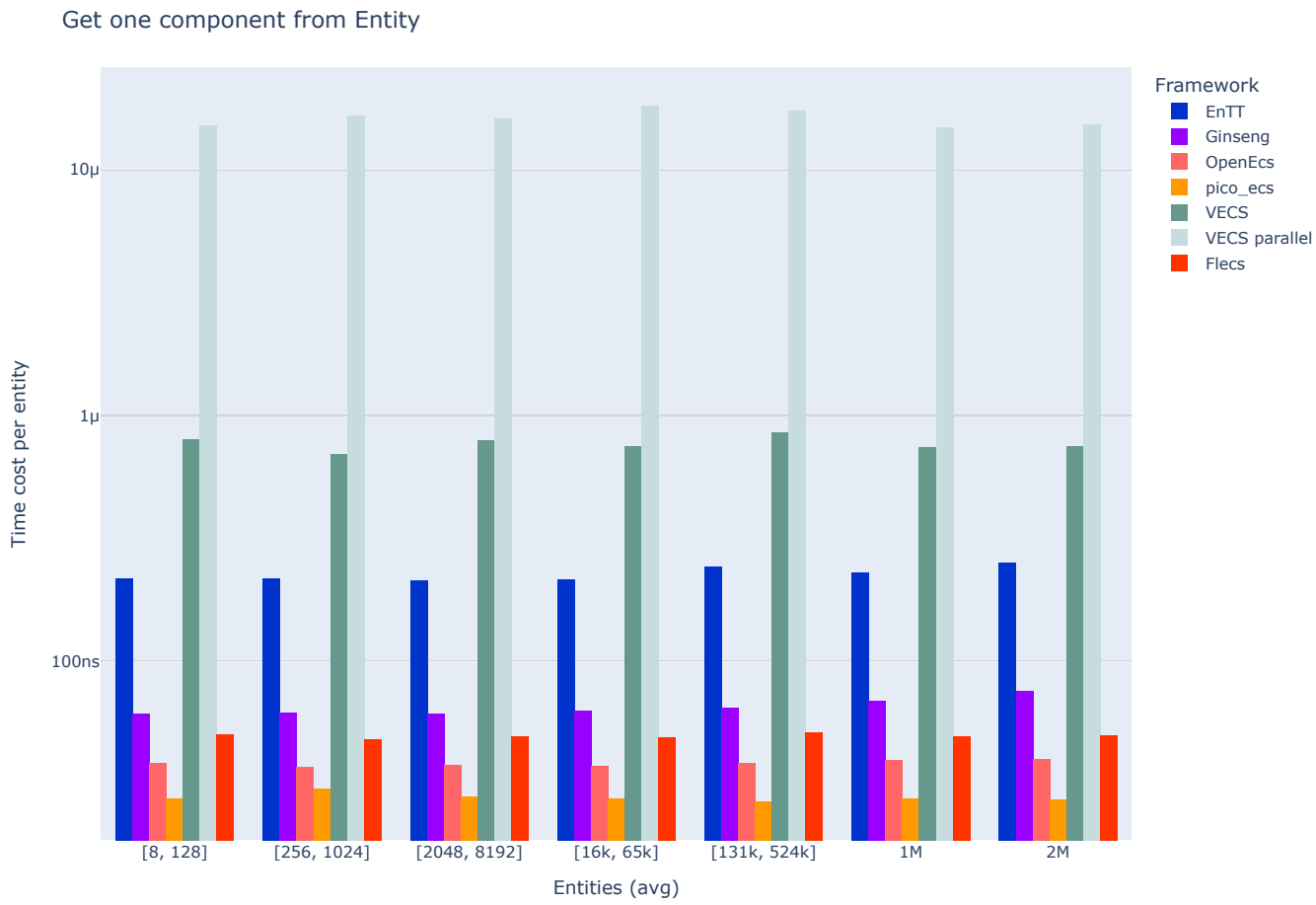


Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Create 256 entities with two Components	608us	1271us	249us	58us	7969us	10036us	4626us
Create ~1K entities with two Components	1524us	1435us	683us	141us	30175us	31829us	6940us
Create ~4K entities with two Components	4749us	2754us	2194us	600us	118743us	122636us	11854us
Create ~16K entities with two Components	17749us	7657us	8086us	2322us	490753us	465320us	23742us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Create ~65K entities with two Components	76ms	31ms	38ms	12ms	1914ms	1988ms	58ms
Create 262K entities with two Components	287ms	125ms	129ms	51ms	7478ms	7834ms	175ms
Create ~1M entities with two Components	1125ms	498ms	523ms	190ms	31643ms	30690ms	609ms
Create ~2M entities with two Components	2286ms	1058ms	1032ms	366ms	59042ms	61366ms	1169ms

Get one (non-const) component from Entity



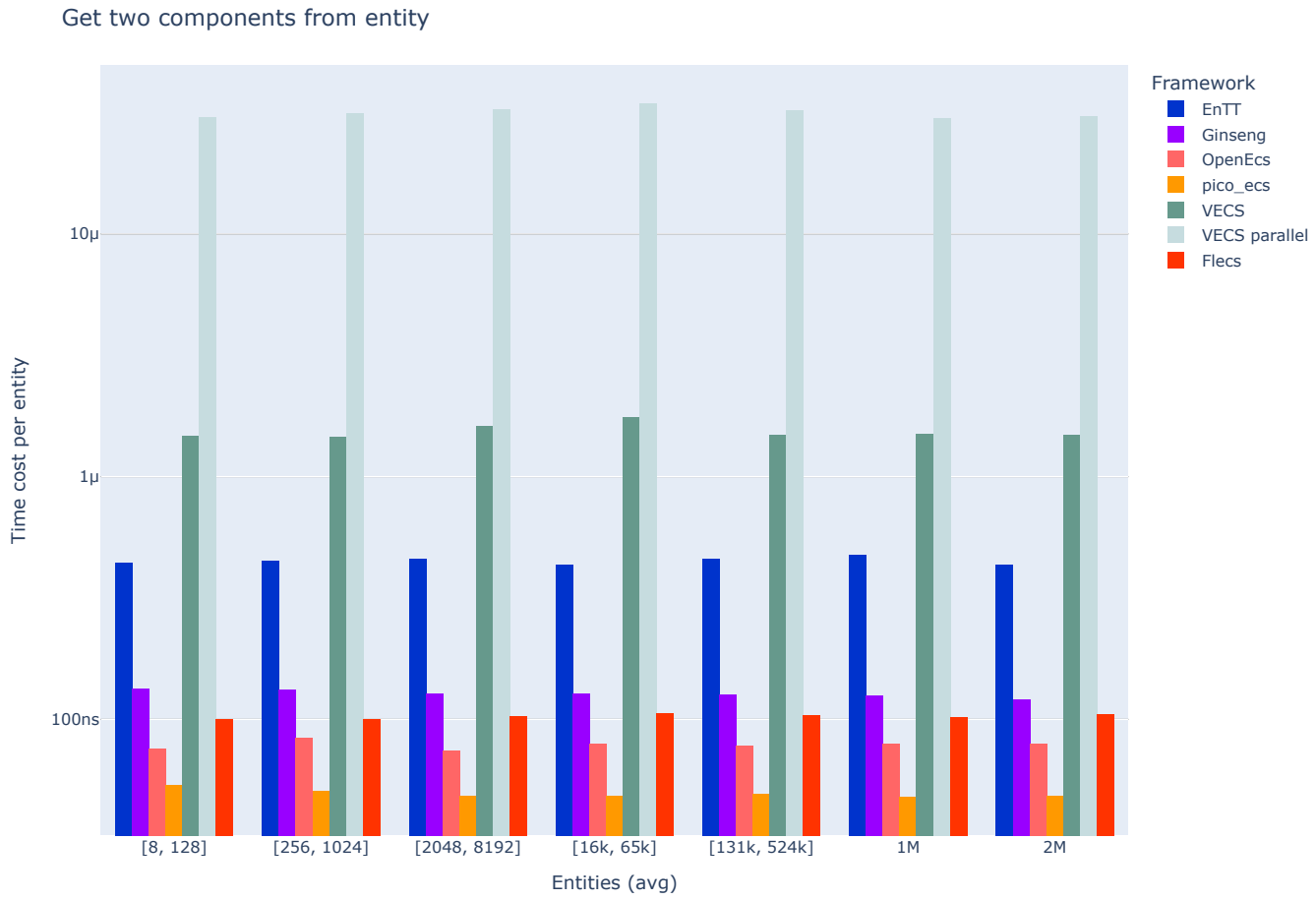
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack one component in 256 entities	57us	15us	9us	6us	189us	3777us	12us
Unpack one component in ~1K entities	222us	62us	38us	36us	745us	19229us	49us
Unpack one component in ~4K entities	879us	248us	157us	112us	2992us	62597us	201us
Unpack one component in ~16K entities	3625us	1010us	615us	462us	11916us	251696us	794us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack one component in ~65K entities	13ms	4ms	2ms	1ms	50ms	1494ms	3ms
Unpack one component in 262K entities	80ms	16ms	9ms	6ms	192ms	4483ms	12ms

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack one component in ~1M entities	240ms	72ms	41ms	28ms	776ms	15551ms	52ms
Unpack one component in ~2M entities	530ms	159ms	84ms	57ms	1570ms	31952ms	104ms

Get two components from entity



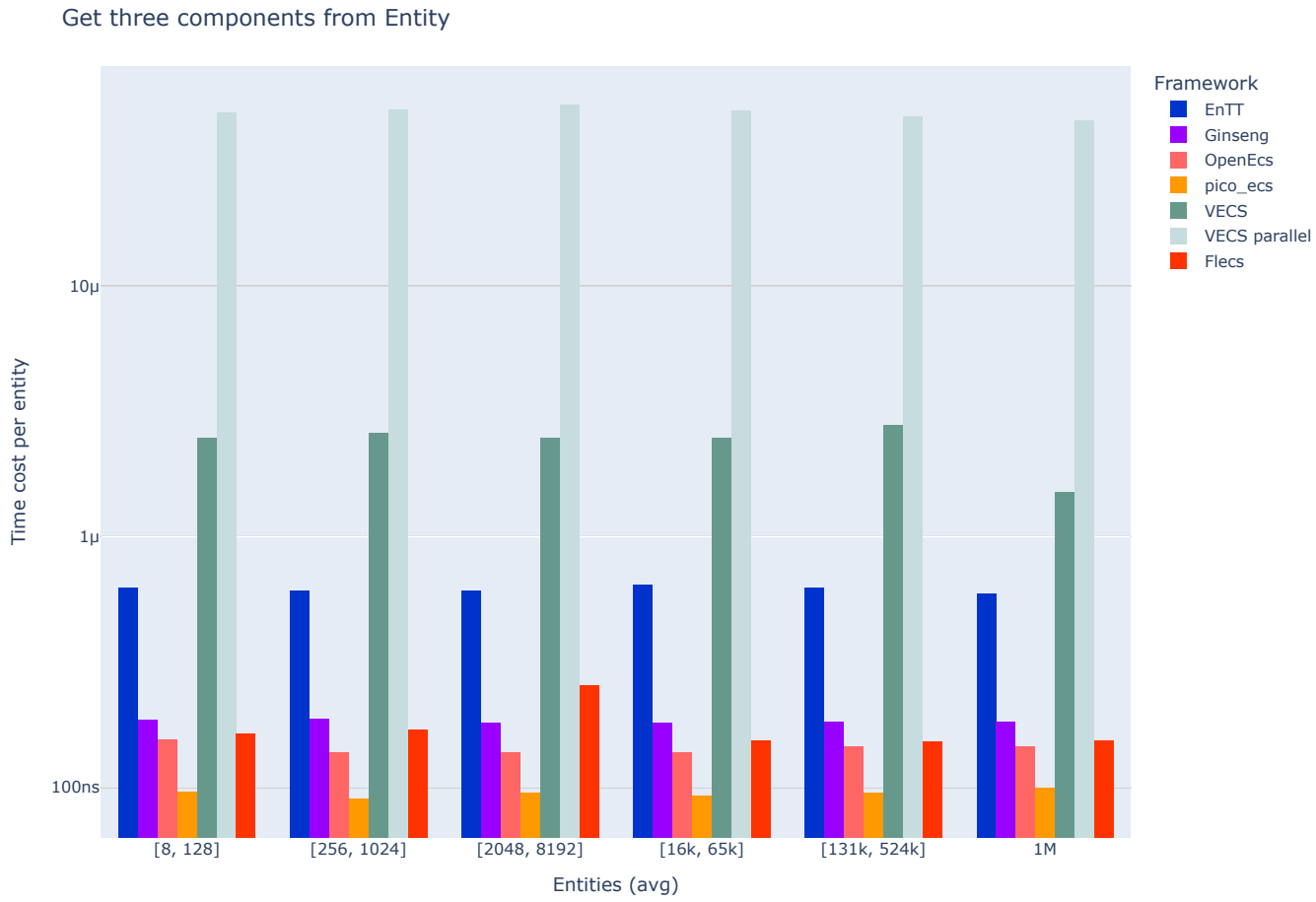
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack two components in 256 entities	112us	33us	22us	13us	376us	7834us	24us
Unpack two components in ~1K entities	480us	132us	78us	49us	1481us	32345us	105us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack two components in ~4K entities	1770us	529us	316us	197us	5998us	133402us	413us
Unpack two components in ~16K entities	7090us	2088us	1238us	797us	27563us	564405us	1639us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack two components in ~65K entities	28ms	8ms	5ms	3ms	99ms	2205ms	7ms
Unpack two components in 262K entities	122ms	33ms	21ms	13ms	385ms	8571ms	26ms
Unpack two components in ~1M entities	499ms	131ms	83ms	50ms	1570ms	31431ms	106ms
Unpack two components in ~2M entities	908ms	253ms	166ms	101ms	3124ms	64126ms	220ms

Get three components from entity

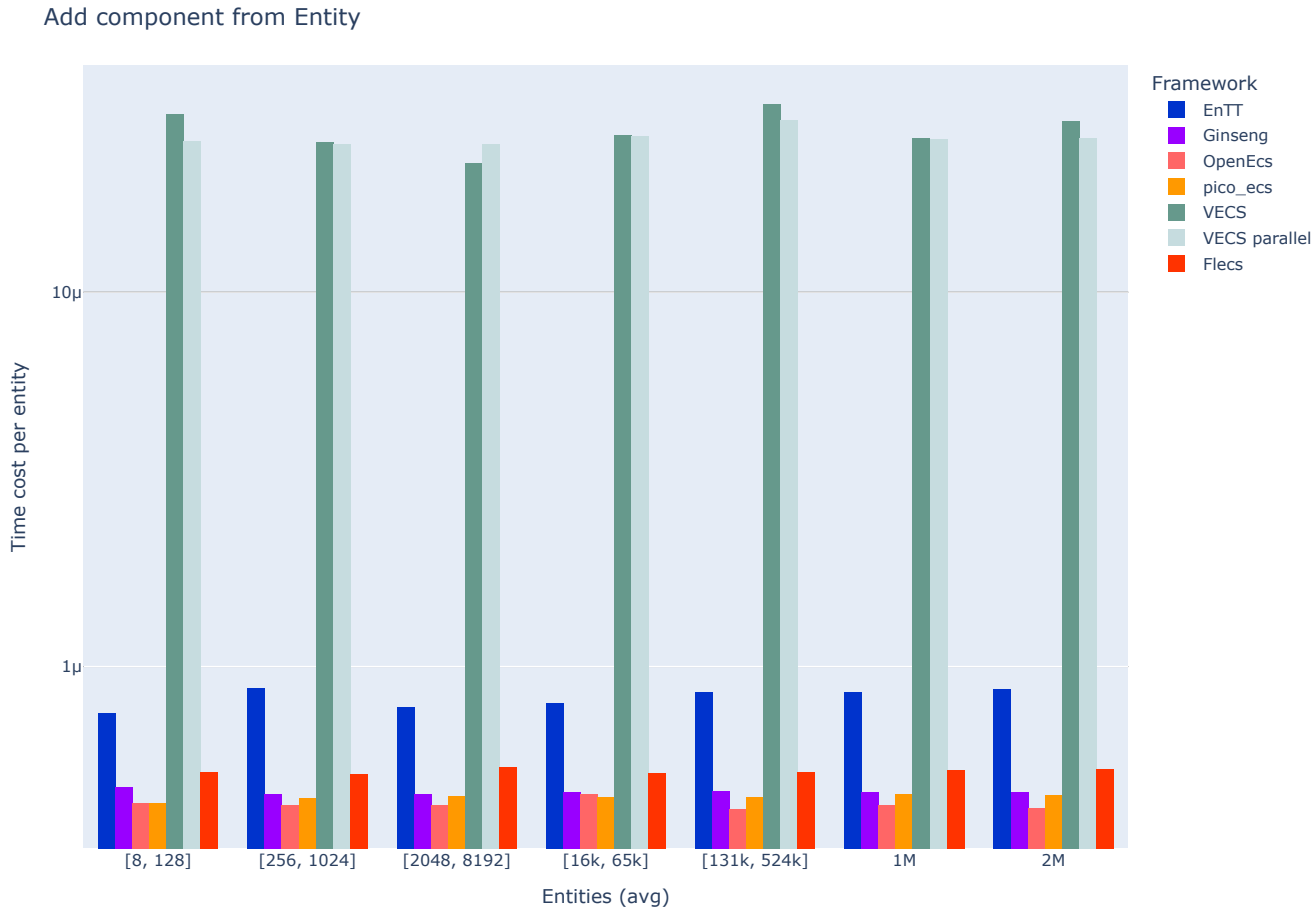


Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack three components in 512 entities	313us	98us	70us	45us	1494us	24973us	76us
Unpack three components in ~2K entities	1312us	364us	281us	201us	5014us	114285us	573us
Unpack three components in ~8K entities	5012us	1518us	1132us	766us	20882us	424499us	1812us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Unpack three components in ~32K entities	20ms	5ms	4ms	3ms	81ms	1659ms	5ms
Unpack three components in 131K entities	82ms	24ms	19ms	12ms	334ms	6717ms	19ms
Unpack three components in 524K entities	333ms	95ms	77ms	50ms	1328ms	23916ms	81ms
Unpack three components in ~1M entities	619ms	191ms	153ms	104ms	1586ms	48040ms	161ms

Add component



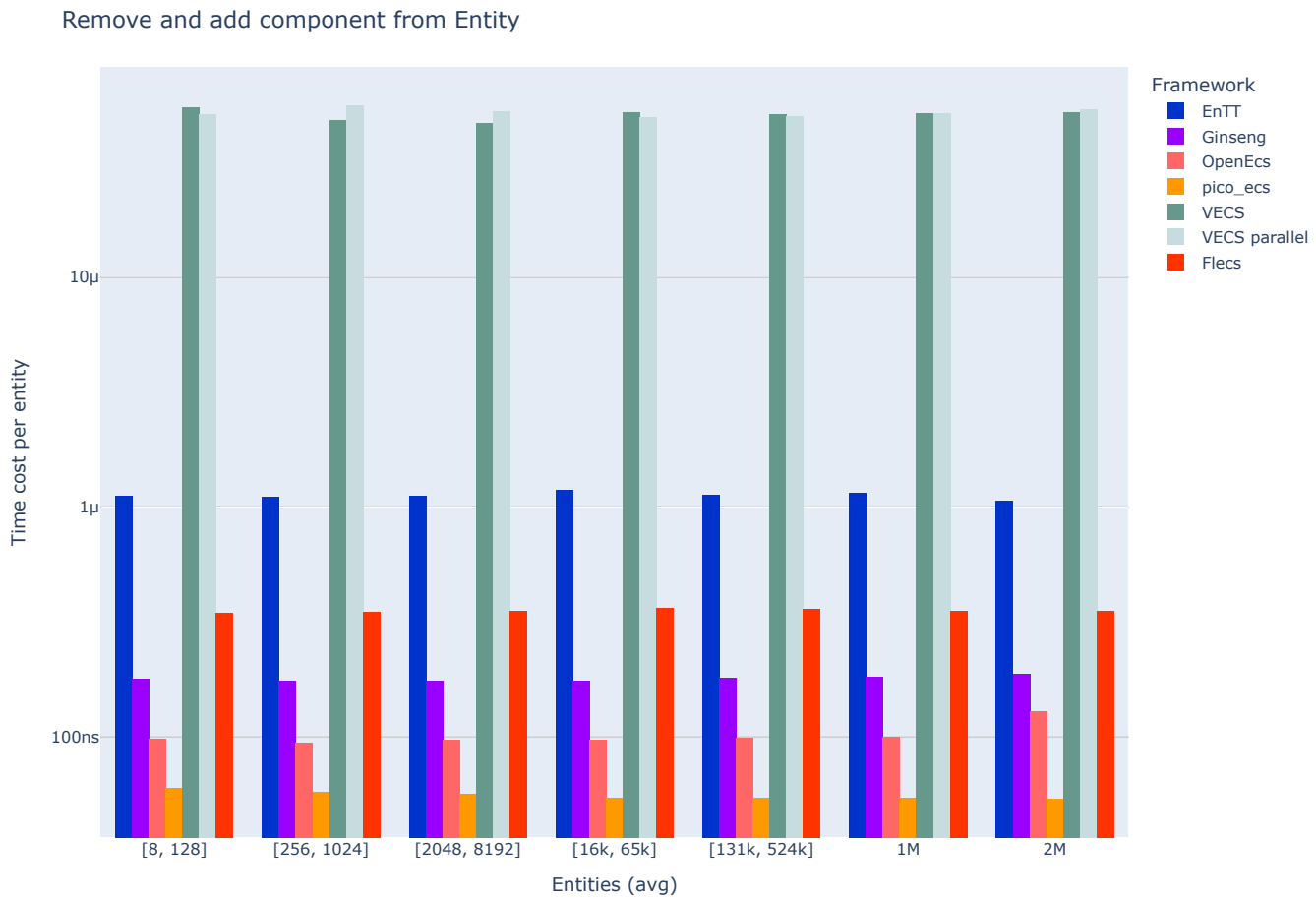
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Add a Component in 256 entities	219us	116us	106us	112us	6891us	6332us	129us
Add a Component in ~1K entities	824us	462us	457us	457us	24471us	25003us	528us
Add a Component in ~4K entities	3193us	1870us	1716us	1835us	84244us	104016us	2099us
Add a Component in ~16K entities	13116us	7463us	8216us	7303us	420495us	407932us	8612us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Add a Component in ~65K entities	52ms	30ms	27ms	29ms	1933ms	1779ms	34ms
Add a Component in 262K entities	220ms	122ms	108ms	116ms	8943ms	6891ms	136ms

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Add a Component in ~1M entities	892ms	483ms	445ms	476ms	26981ms	26876ms	554ms
Add a Component in ~2M entities	1818ms	967ms	875ms	947ms	59983ms	53872ms	1111ms

Remove and add component



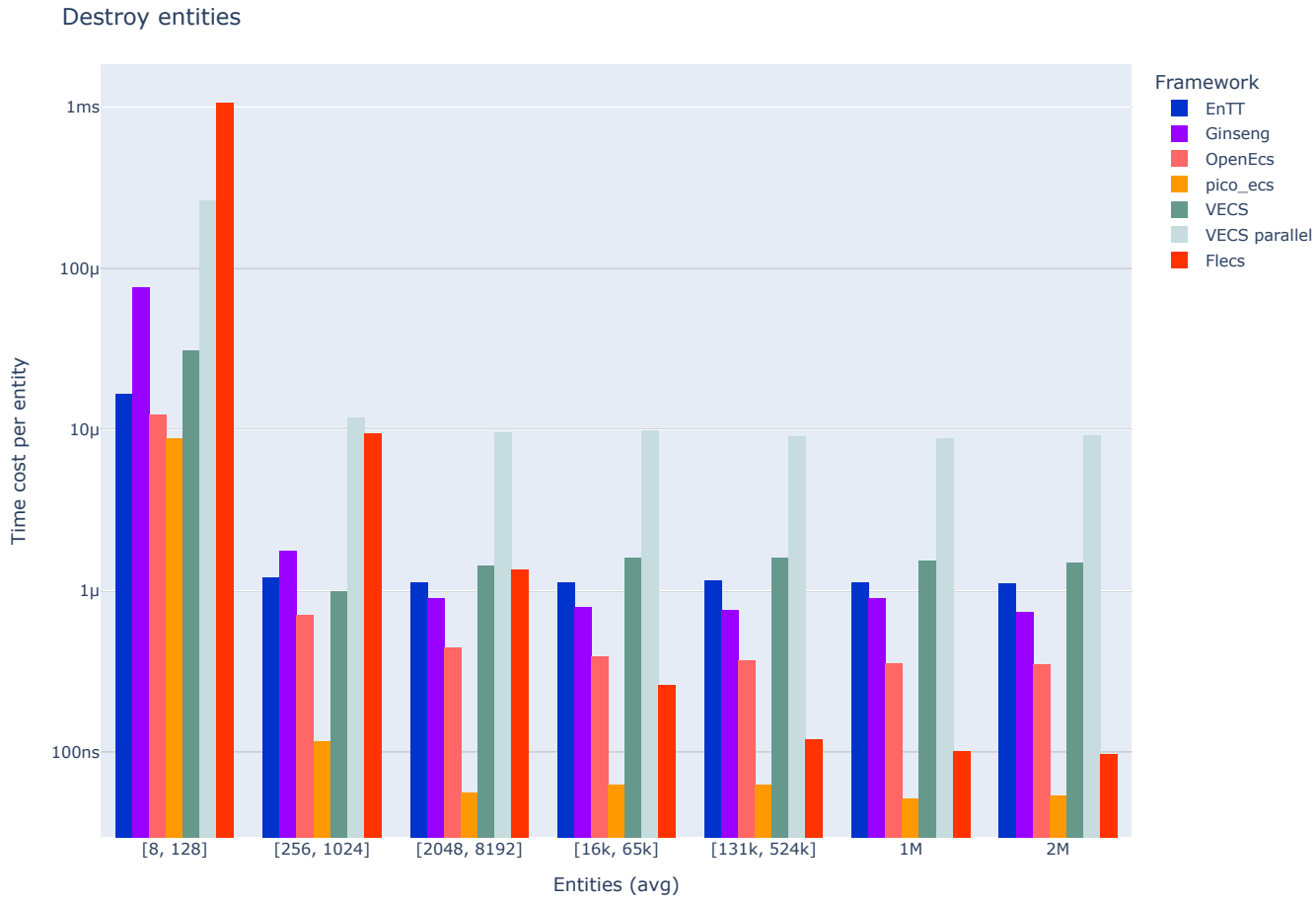
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Remove and Add a Component in 256 entities	281us	44us	23us	14us	11548us	14315us	89us
Remove and Add a Component in ~1K entities	1142us	180us	96us	58us	55714us	55741us	358us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Remove and Add a Component in ~4K entities	4577us	712us	397us	226us	189751us	220701us	1432us
Remove and Add a Component in ~16K entities	20501us	2861us	1574us	889us	826273us	843599us	6124us

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Remove and Add a Component in ~65K entities	74ms	11ms	6ms	3ms	3283ms	3173ms	23ms
Remove and Add a Component in 262K entities	298ms	47ms	25ms	14ms	13216ms	13060ms	91ms
Remove and Add a Component in ~1M entities	1210ms	190ms	104ms	56ms	54337ms	54204ms	370ms
Remove and Add a Component in ~2M entities	2225ms	395ms	270ms	112ms	109836ms	112381ms	737ms

Destroy entities



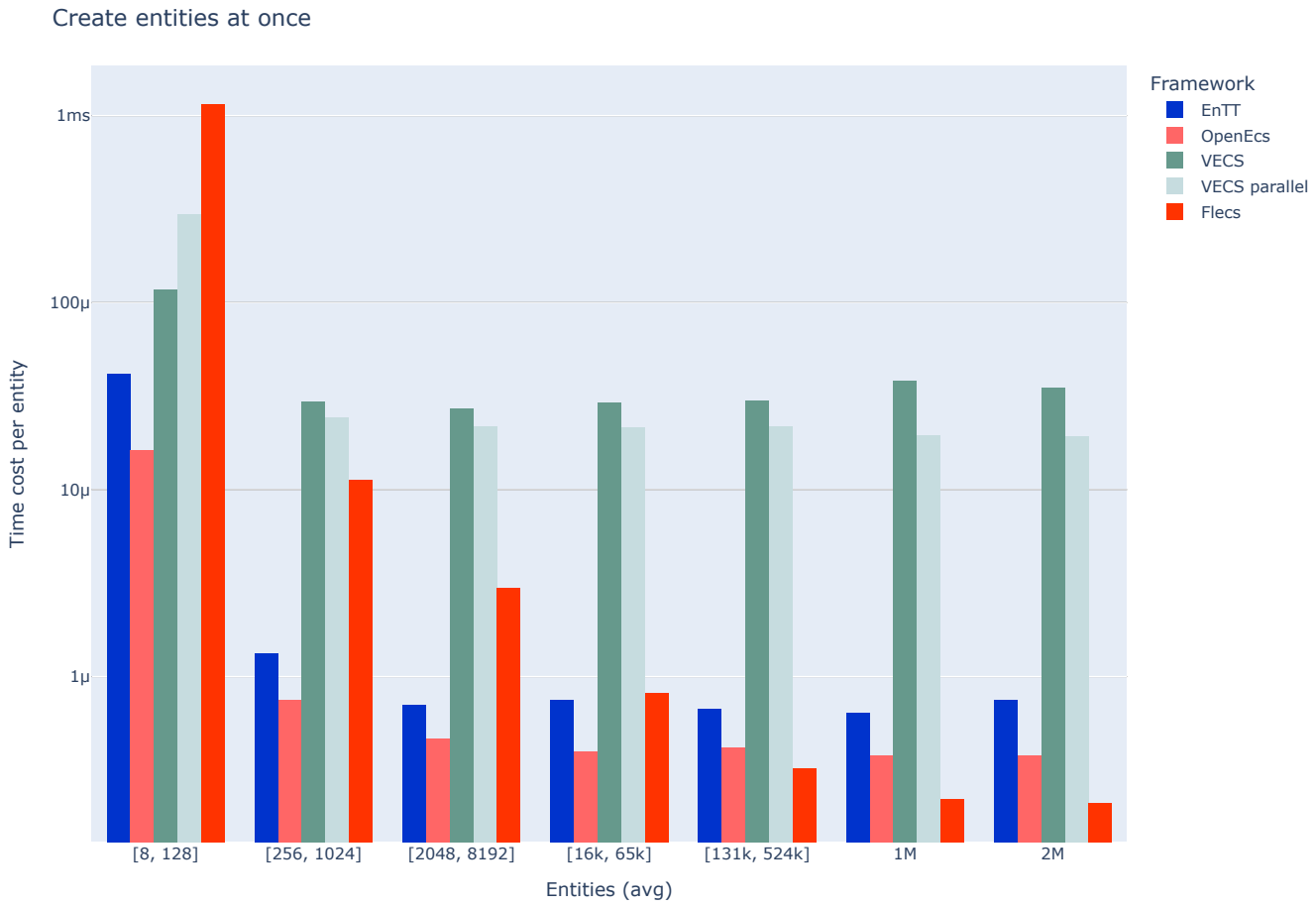
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	**Ginseng**	**OpenEcs**	**pico_ecs**	**VECS**	**VECS parallel**	**Flecs**
Destroy 256 entities with two components	321us	640us	239us	43us	265us	3652us	4066us
Destroy ~1K entities with two components	1190us	1221us	518us	76us	886us	10002us	4309us
Destroy ~4K entities with two components	4578us	3627us	1801us	222us	6322us	37981us	4988us
Destroy ~16K entities with two components	18542us	13188us	6357us	810us	22577us	164776us	6266us

	EnTT	**Ginseng**	**OpenEcs**	**pico_ecs**	**VECS**	**VECS parallel**	**Flecs**
Destroy ~65K entities with two components	72ms	51ms	26ms	4ms	108ms	619ms	11ms
Destroy 262K entities with two components	299ms	200ms	95ms	18ms	429ms	2389ms	29ms

	EnTT	Ginseng	OpenEcs	pico_ecs	VECS	VECS parallel	Flecs
Destroy ~1M entities with two components	1182ms	945ms	369ms	54ms	1599ms	9290ms	106ms
Destroy ~2M entities with two components	2342ms	1540ms	737ms	111ms	3123ms	19311ms	202ms

Create entities at once



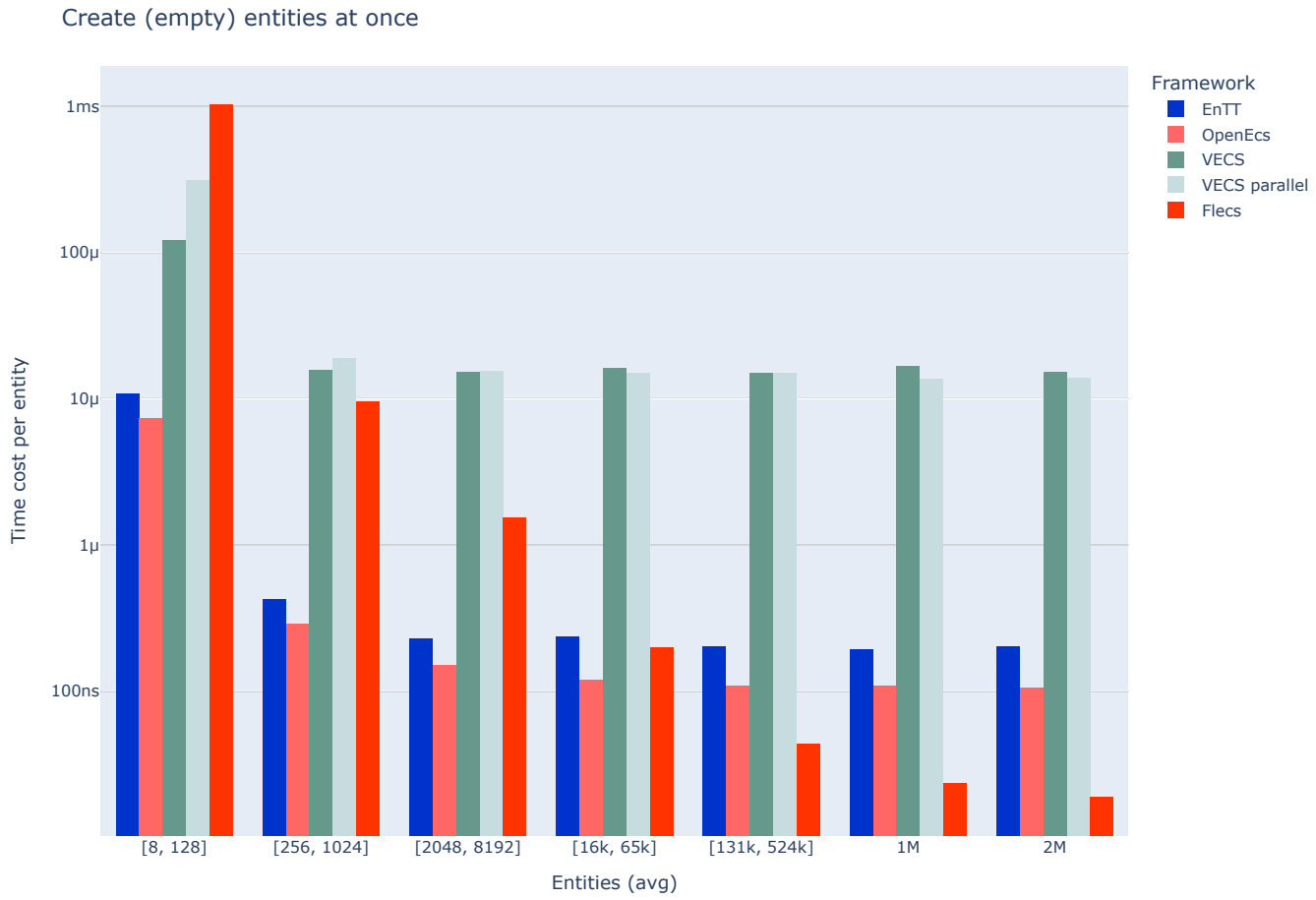
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	OpenEcs	VECS	VECS parallel	Flecs
Create 256 entities with two components at once	467us	235us	7543us	6873us	4619us
Create ~1K entities with two components at once	945us	613us	29812us	22665us	6870us
Create ~4K entities with two components at once	2840us	1880us	95439us	90455us	10313us

	EnTT	OpenEcs	VECS	VECS parallel	Flecs
Create ~16K entities with two components at once	13390us	6584us	468029us	348659us	19607us

	EnTT	OpenEcs	VECS	VECS parallel	Flecs
Create ~65K entities with two components at once	49ms	25ms	1989ms	1425ms	34ms
Create 262K entities with two components at once	176ms	102ms	7589ms	5629ms	82ms
Create ~1M entities with two components at once	667ms	397ms	40147ms	20416ms	232ms
Create ~2M entities with two components at once	1566ms	797ms	73818ms	40418ms	441ms

Create empty entities at once



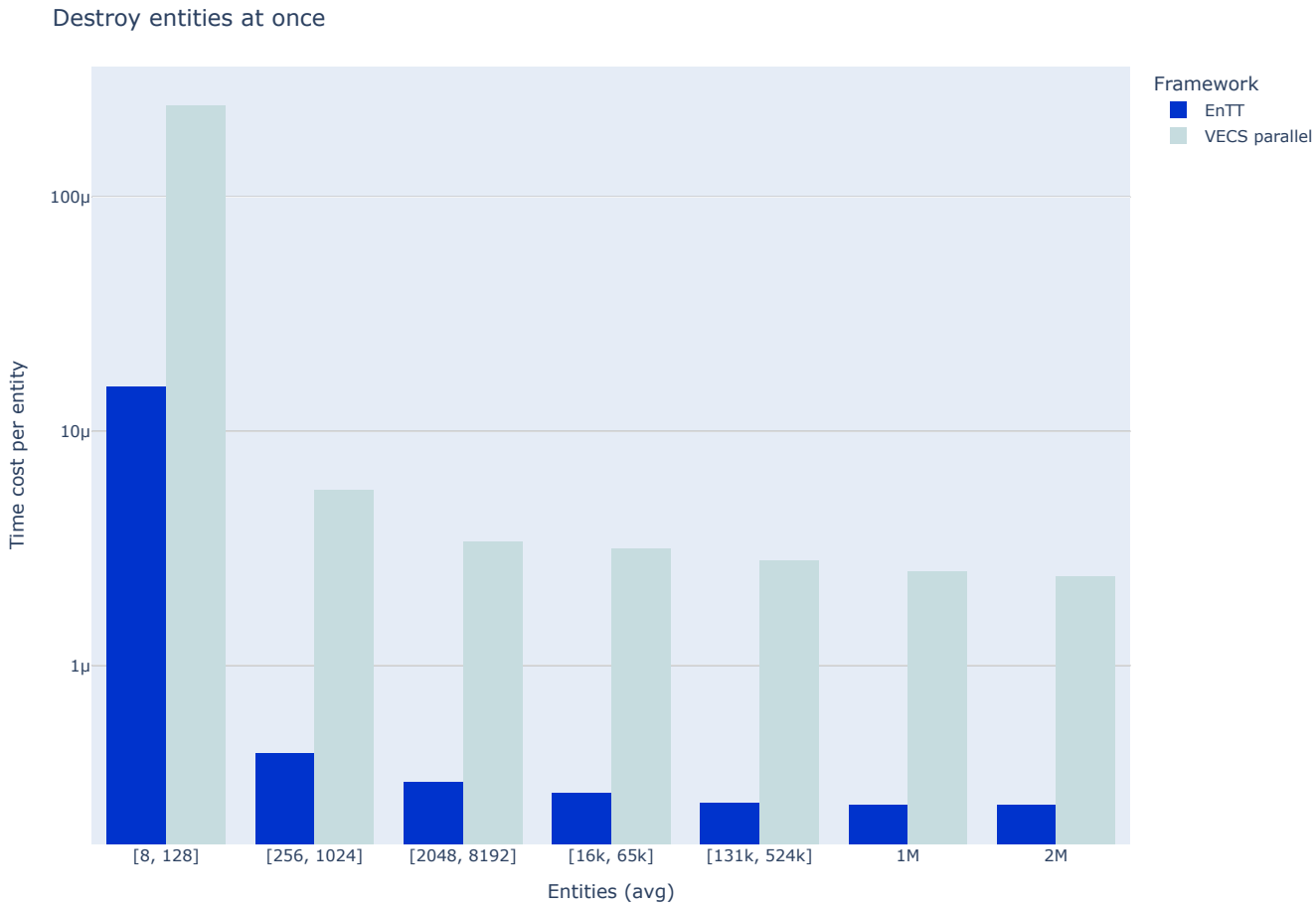
Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	OpenEcs	VECS	VECS parallel	Flecs

	EnTT	OpenEcs	VECS	VECS parallel	Flecs
Create 256 (empty) entities at once	144us	92us	4102us	5292us	4251us
Create ~1K (empty) entities at once	303us	226us	15740us	18302us	4154us
Create ~4K (empty) entities at once	918us	596us	62268us	63033us	6001us
Create ~16K (empty) entities at once	3455us	2107us	296771us	245514us	6202us

	EnTT	OpenEcs	VECS	VECS parallel	Flecs
Create ~65K (empty) entities at once	18ms	7ms	976ms	977ms	5ms
Create 262K (empty) entities at once	52ms	28ms	3923ms	3963ms	10ms
Create ~1M (empty) entities at once	203ms	114ms	17514ms	14466ms	24ms
Create ~2M (empty) entities at once	423ms	223ms	31851ms	29053ms	39ms

Destroy entities at once



Graph shows cost per entity, tables shows total cost. lower is faster.

	EnTT	VECS parallel
Destroy 256 entities with two components at once	134us	1962us
Destroy ~1K entities with two components at once	348us	3965us

	EnTT	VECS parallel
Destroy ~4K entities with two components at once	1381us	14006us
Destroy ~16K entities with two components at once	4876us	56349us

	EnTT	VECS parallel
Destroy ~65K entities with two components at once	19ms	193ms
Destroy 262K entities with two components at once	69ms	721ms
Destroy ~1M entities with two components at once	268ms	2653ms
Destroy ~2M entities with two components at once	532ms	5064ms

Candidates

EntityX by @alecthomas

> Entity Component Systems (ECS) are a form of decomposition that completely decouples entity logic and data from the entity "objects" themselves. The Evolve your Hierarchy article provides a solid overview of EC systems and why you should use them.

Version: 1.1.2 (Apr 2023)

EnTT by @skypjack

> EnTT is a header-only, tiny and easy to use library for game programming and much more written in modern C++.

Version: v3.13.2

Ginseng by @apples

> Ginseng is an entity-component-system (ECS) library designed for use in games.

The main advantage over similar libraries is that the component types do not need to be listed or registered. Component types are detected dynamically.

Any function-like object can be used as a system. The function's parameters are used to determine the required components.

Version: 1.1 (Dec 2021)

mustache by @kirillochnev

> A fast, modern C++ Entity Component System

Version: 0.2 (Feb 2024)

OpenEcs by @Gronis

> Open Ecs is an Entity Component System that uses metaprogramming, cache coherency, and other useful tricks to maximize performance and configurability. It is written in c++11 without further dependencies.

Version: 0.1.101 (Apr 2017)

Flecs by @SanderMertens

> Flecs is a fast and lightweight Entity Component System that lets you build games and simulations with millions of entities.

Version: v4.0.1

pico_ecs by @empyreanx

> A collection of cross-platform single header libraries written in C. Pure and simple ECS.

Version: 2.3 (Sep 2023)

gaia-ecs by @richardbiely

> Gaia-ECS is a fast and easy-to-use ECS framework.

Version: v0.8.6

VECS by @hlavacshelmut

> The Vienna Entity Component System (VECS) is a C++20 based ECS for game engines.

Version: 0.1

VECS parallel by @hlavacshelmut @hoelzlisabella

> The Vienna Entity Component System (VECS) is a C++20 based ECS for game engines.

Version: 0.1

Environment

- **OS:** Windows
- **CPU:** 2.11GHz @ 8Cores
- **RAM:** 15.78GB