

Description	Code sample	Short description	Tag	Component	Enableable Tag	Enableable Component
No structural changes: Just enable/disable components	<code>state.EntityManager.SetComponentEnabled<Foo>(_disabledFooQuery, true);</code>	Enable component	N/A	N/A	0.03	0.02
EntityManager and EntityQuery	<code>state.EntityManager.AddComponent<Foo>(_noFooQuery);</code>	EntityManager+Query	2.37	6.35	6.47	6.33
EntityManager and a NativeArray of Entities	<code>state.EntityManager.AddComponent<Foo>(_entities);</code>	EntityManager+Array	68.18	69.11	60.88	65.34
EntityCommandBuffer and EntityQuery	<code>var ecb = new EntityCommandBuffer(Allocator.TempJob); ecb.AddComponent<Foo>(_noFooQuery); ecb.Playback(state.EntityManager); ecb.Dispose();</code>	ECB+Query	63.25	63.72	60.85	60.77
EntityCommandBuffer and a NativeArray of Entities	<code>var ecb = new EntityCommandBuffer(Allocator.TempJob); ecb.AddComponent<Foo>(_entities); ecb.Playback(state.EntityManager); ecb.Dispose();</code>	ECB+Array	64.53	69.33	65.98	65.73
IJobChunk.ScheduleParallel() with an EntityCommandBuffer.AsParallelWriter()	<pre>[BurstCompile] partial struct AddFooJob : IJobChunk { public EntityCommandBuffer.ParallelWriter Writer; public EntityTypeHandle EntityTypeHandle; public void Execute(in ArchetypeChunk chunk, int unfilteredChunkIndex, bool useEnabledMask, in v128 chunkEnabledMask) { Writer.AddComponent<Foo>(unfilteredChunkIndex, chunk.GetNativeArray(EntityTypeHandle)); } }</pre>	ECB+ParallelWriter(IJobChunk)	36.37	36.7	41.04	40.94
IJobEntity.ScheduleParallel() with an EntityCommandBuffer.AsParallelWriter()	<pre>[BurstCompile] partial struct AddFooJob : IJobEntity { public EntityCommandBuffer.ParallelWriter Writer; public void Execute(Entity entity, [ChunkIndexInQuery] int chunkIndexInQuery) { Writer.AddComponent<Foo>(chunkIndexInQuery, entity); } }</pre>	ECB+ParallelWriter(IJobEntity)	372.34	371.86	354.64	361.76