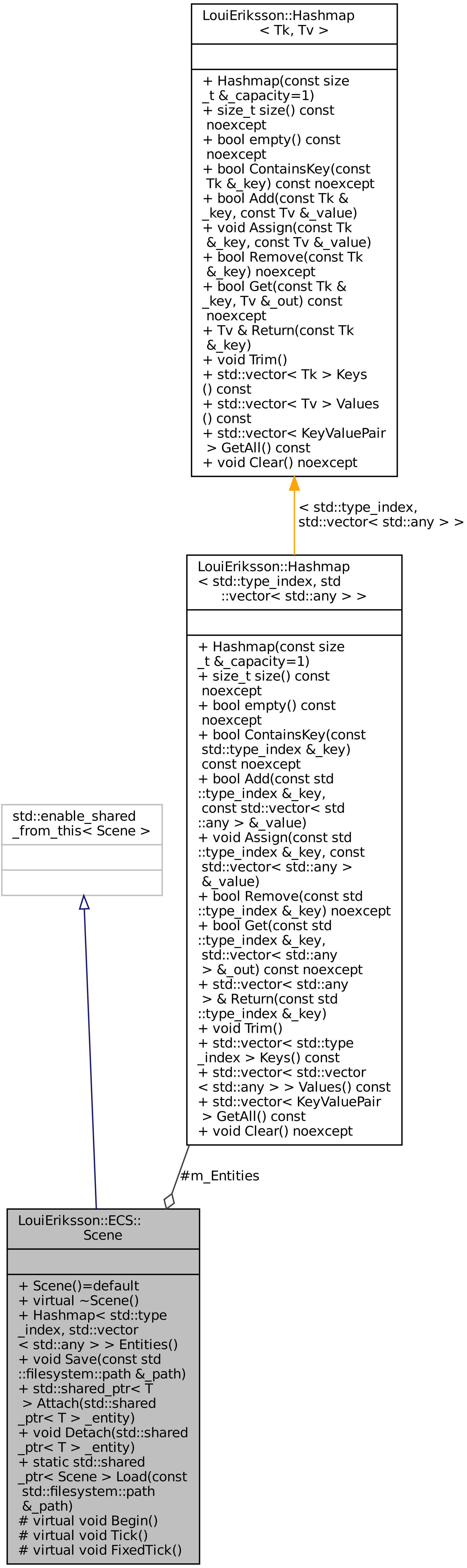
LouiEriksson::Hashmap
< Tk, Tv >
+ Hashmap(const size
_t &_capacity=1)
+ size_t size() const
noexcept
+ bool empty() const
noexcept
+ bool ContainsKey(const
Tk &_key) const noexcept
+ bool Add(const Tk &
_key, const Tv &_value)
+ void Assign(const Tk
&_key, const Tv &_value)
+ bool Remove(const Tk
&_key) noexcept
+ bool Get(const Tk &
_key, Tv &_out) const
noexcept
+ Tv & Return(const Tk
&_key)
+ void Trim()
+ std::vector< Tk > Keys
() const
+ std::vector< Tv > Values
() const
+ std::vector< KeyValuePair
> GetAll() const
+ void Clear() noexcept
< std::type_index,
std::vector< std::any > >
LouiEriksson::Hashmap
< std::type_index, std
::vector< std::any > >
+ Hashmap(const size
_t &_capacity=1)
+ size_t size() const
noexcept
+ bool empty() const
noexcept
+ bool ContainsKey(const
std::type_index &_key)
const noexcept
+ bool Add(const std
::type_index &_key,
const std::vector< std
::any > &_value)
+ void Assign(const std
::type_index &_key, const
std::vector< std::any >
&_value)
+ bool Remove(const std
::type_index &_key) noexcept
+ bool Get(const std
::type_index &_key,
std::vector< std::any
> &_out) const noexcept
+ std::vector< std::any
> & Return(const std
::type_index &_key)
+ void Trim()
+ std::vector< std::type
_index > Keys() const
+ std::vector< std::vector
< std::any > > Values() const
+ std::vector< KeyValuePair
> GetAll() const
+ void Clear() noexcept
std::enable_shared
_from_this< Scene >
#m_Entities
LouiEriksson::ECS::
Scene
+ Scene()=default
+ virtual ~Scene()
+ Hashmap< std::type
_index, std::vector
< std::any > > Entities()
+ void Save(const std
::filesystem::path &_path)
+ std::shared_ptr< T
> Attach(std::shared
_ptr< T > _entity)
+ void Detach(std::shared
_ptr< T > _entity)
+ static std::shared
_ptr< Scene > Load(const
std::filesystem::path
&_path)
# virtual void Begin()
# virtual void Tick()
# virtual void FixedTick()