

```
depth_clustering::Abstract  
Sender< std::unordered  
_map< uint16_t, Cloud >  
>::AddClient
```

```
depth_clustering::Abstract  
Sender< std::unordered  
_map< uint16_t, Cloud >  
>::RemoveClient
```

```
depth_clustering::Abstract  
Sender< std::unordered  
_map< uint16_t, Cloud >  
>::ShareDataWithAllClients
```

```
depth_clustering::Identifiable::id
```

```
graph LR; A["depth_clustering::AbstractSender< std::unordered_map< uint16_t, Cloud > >::AddClient"] --> D["depth_clustering::Identifiable::id"]; B["depth_clustering::AbstractSender< std::unordered_map< uint16_t, Cloud > >::RemoveClient"] --> D; C["depth_clustering::AbstractSender< std::unordered_map< uint16_t, Cloud > >::ShareDataWithAllClients"] --> D;
```

The diagram illustrates a design pattern where three different abstract sender methods (AddClient, RemoveClient, and ShareDataWithAllClients) all interact with a common identifier (id) of the Identifiable interface. The identifier is represented by a grey box, while the sender methods are in white boxes. Blue arrows indicate the flow of interaction from each sender method to the identifier.