

events_1::EventEmitter

```
classDiagram
    class Operation {
        + constructor()
        + cancel()
        + getOperation()
        + _unpackResponse()
        + promise()
        - _listenForEvents()
        - startPolling_()
    }
    class events_1_EventEmitter {
    }
    Operation --|> events_1_EventEmitter
```

The diagram illustrates a class hierarchy. At the bottom is the 'Operation' class, which is shaded gray and contains a list of methods: a constructor, 'cancel()', 'getOperation()', '_unpackResponse()', 'promise()', and two private methods '_listenForEvents()' and 'startPolling_()'. Above it is the 'events_1::EventEmitter' class, which is white. A blue arrow with an open triangle head points from the 'Operation' class to the 'events_1::EventEmitter' class, indicating that 'Operation' inherits from 'events_1::EventEmitter'.

Operation

- + constructor()
- + cancel()
- + getOperation()
- + _unpackResponse()
- + promise()
- _listenForEvents()
- startPolling_()