

Error: Found 2 powers (GERMANY, TURKEY) trying to control same province (SPA/SC) with same control type (I VS I).

setController

src/gui/utils/province.js:57

```
54 |         throw new Error(`Invalid province control type (${controlType}), expected
    |         'C', 'I' or 'U'.`);
55 |         if (this.controller && this.controller !== controller) {
56 |             const controlTypeComparison = this.compareControlType(controlType,
    |             this.controlType);
> 57 |             if (controlTypeComparison === 0)
    |             ^ 58 |                 throw new Error(`Found 2 powers (${this.controller},
    |                 ${controller}) trying to control same province ` +
59 |                 `(${this.name}) with same control type (${controlType} VS
    |                 ${this.controlType}).`);
60 |             if (controlTypeComparison > 0)
```

View compiled

__set_controller

src/gui/utils/province.js:48

```
45 |     __set_controller(controller, controlType) {
46 |         this.controller = controller;
47 |         this.controlType = controlType;
> 48 |         for (let coast of Object.values(this.coasts))
    |         ^ 49 |             coast.setController(controller, controlType);
50 |     }
51 | }
```

View compiled

setController

src/gui/utils/province.js:62

```
59 |         `(${this.name}) with same control type (${controlType} VS
    |         ${this.controlType}).`);
60 |         if (controlTypeComparison > 0)
61 |             this.__set_controller(controller, controlType);
> 62 |         } else
    |         ^ 63 |             this.__set_controller(controller, controlType);
64 |     }
65 | }
```

View compiled

MapData

src/gui/utils/map_data.js:44

```
41 |         this.getProvince(center).setController(power.name, 'C');
42 |     }
43 |     for (let loc of power.influence) {
> 44 |         this.getProvince(loc).setController(power.name, 'I');
    |         ^ 45 |     }
46 |     for (let unit of power.units) {
47 |         this.__add_unit(unit, power.name);
```

View compiled

renderMapForMessages

src/gui/pages/content_game.jsx:992

```
989 | <div id="messages-map" key="messages-map">
990 |   <Map game={gameEngine}
991 |     showAbbreviations={this.state.showAbbreviations}
> 992 |     mapData={new MapData(this.getMapInfo(gameEngine.map_name),
gameEngine)}
    | ^ 993 |     onError={this.getPage().error}
994 |     orders={showOrders &&
gameEngine.order_history.contains(gameEngine.phase) &&
gameEngine.order_history.get(gameEngine.phase) || null}
995 |     onHover={showOrders ? this.displayLocationOrders : null}
```

View compiled

renderTabMessages

src/gui/pages/content_game.jsx:1158

```
1155 | <Row>
1156 |   <div className={'col-xl'}>
1157 |     {this.state.historyCurrentOrders && (
> 1158 |       <div className={'history-current-
orders'}>{this.state.historyCurrentOrders.join(', ')}</div>
    | ^ 1159 |     )}
1160 |     {this.renderMapForMessages(engine, this.state.historyShowOrders)}
1161 |   </div>
```

View compiled

render

src/gui/pages/content_game.jsx:1348

```
1345 |     afterTitle={navAfterTitle}
1346 |     username={page.channel.username}
1347 |     navigation={navigation}/>
> 1348 | <Tabs menu={tabNames} titles={tabTitles} onChange={this.onChangeMainTab}
active={mainTab}>
    | ^ 1349 |   { /* Tab Phase history. */}
1350 |   {(hasTabPhaseHistory && mainTab === 'phase_history' &&
this.renderTabResults(mainTab === 'phase_history', engine)) || ''}
1351 |   {mainTab === 'messages' && this.renderTabMessages(mainTab ===
'messages', engine, currentPowerName)}
```

View compiled

 16 stack frames were collapsed.

setState

src/gui/pages/page.jsx:77

```
74 | }
75 |
76 | setState(state) {
> 77 |   return new Promise(resolve => super.setState(state, resolve));
    | ^ 78 | }
79 |
80 | onReconnectionError(error) {
```

View compiled

