

Actions do not have independent descriptions in separate files, but are present in descriptions of other game elements, for example, in job states.

Actions are used to affect the state of the game objects or to trigger an event.

Description structure

The action description always includes its type from the list below in the ActionType parameter, as well as a set of additional parameters depending on it.

- [Message](#)
- [Message2](#)
- [PlayerUpgradeAdd](#)
- [PlayerCargoAdd](#)
- [PlayerCargoRemove](#)
- [PlayerFundAdd](#)
- PlayerFundRemove
- SetWeather
- CityFundAdd
- CityFundRemove
- CityChangePhase
- CitySetOpen
- CreateSimpleEntity
- CreateNPC
- CreateCargoBox
- RemoveEntity

Message

Sending a message to the player through the intercom.

```
{  
  
    "ActionType": "Message",  
  
    "Name": "ExampleMessageText",  
  
    "Sound": "ExampleMessageSound",  
  
}
```

Name - the name of the message text from the localization

Sound - (optional) the name of the sound, if not specified - take Name.

Message2

Sending an improved message to the player through the intercom.

```
{  
  
    "ActionType": "Message2",  
  
    "Name": "ExampleMessageName",  
  
    "PersonName": "ExamplePersonName",  
  
}
```

Name - the name of the message description (for details, see Message Descriptions).

PersonName - the name of the author's description of the identity of the message (for details, see Personality Descriptions).

PlayerUpgradeAdd

Adding an upgrade to the player's ship.

```
{  
  
    "ActionType": "PlayerUpgradeAdd",  
  
    "Name": "ExampleUpgradeName",  
  
}
```

Name - the name of the upgrade description (for details, see Upgrade Descriptions).

PlayerCargoAdd

Adding goods to the hold of the player's ship.

```
{  
  
  "ActionType": "PlayerCargoAdd",  
  
  "Name": "ExampleProductName",  
  
}
```

Name - the name of the product description (more in Product Descriptions).

PlayerCargoRemove

Removal of goods from the hold of the player's ship.

```
{  
  
  "ActionType": "PlayerCargoRemove",  
  
  "Name": "ExampleCargoName",  
  
}
```

Name - the name of the product description (more in Product Descriptions).

PlayerFundAdd

Adding currency to the player.

```
{  
  
  "ActionType": "PlayerFundAdd",  
  
  "Name": "ExampleFundName",  
  
  "Count": 100,  
  
}
```

Name - the name of the currency.

Count - the amount of currency.

PlayerFundRemove

Removing the currency from the player.

```
{  
  
  "ActionType": "PlayerFundRemove",  
  
  "Name": "ExampleFundName",  
  
  "Count": 100,  
  
}
```

Name - the name of the currency.

Count - the amount of currency.

SetWeather

Set the weather.

```
{  
  
  "ActionType": "SetWeather",  
  
  "Name": "ExampleWeatherName",  
  
}
```

Name - the name of the weather description (more in the Weather Descriptions).

CityFundAdd

Adding the currency of the city (base).

```
{  
  
  "ActionType": "CityFundAdd",  
  
  "Name": "ExampleFundName",  
  
  "Count": 100,  
  
  "CityName": "ExampleBaseName",  
  
}
```

Name - the name of the currency.

Count - the amount of currency.

CityName - the name of the base.

CityFundRemove

Removal of currency from the city (base).

```
{  
  
  "ActionType": "CityFundRemove",  
  
  "Name": "ExampleFundName",  
  
  "Count": 100,  
  
  "CityName": "ExampleBaseName",  
  
}
```

Name - the name of the currency.

Count - the amount of currency.

CityName is the name of the base.

CityChangePhase

Setting the phase of the construction of the city (base).

```
{  
  
  "ActionType": "CityChangePhase",  
  
  "Name": "ExampleCityName",  
  
  "Count": 2,  
  
}
```

Name - the name of the description of the base (more in the base Descriptions).

Count is the number of the phase.

CitySetOpen

Opening of the city (base).

```
{  
  
  "ActionType": "CitySetOpen",  
  
  "Name": "ExampleCityName",  
  
}
```

Name - the name of the description of the base (more in the base Descriptions).

CreateSimpleEntity

Creating a game object.

```
{  
  
  "ActionType": "CreateSimpleEntity",  
  
  "Name": "ExampleEntityName",  
  
  "TargetDescr": "ExampleTargetName",  
  
  "PrefabName": "ExamplePrefabName",  
  
  "NeedCorrectPosition": true,  
  
}
```

Name - the unique name that will be assigned to the object.

TargetDescr - name of the goal description (details in the Descriptions of goals), which determines the position of the object being created.

PrefabName - the path to the prefab of the object.

NeedCorrectPosition - (optional) whether you need to adjust the position and set the object to the ground.

CreateNPC

Creation of a character in the world.

```
{  
  
    "ActionType": "CreateNPC",  
  
    "Name": "ExampleNPCName",  
  
    "TargetDescr": "ExampleTargetName"  
  
    "DescrName": "ExampePassengerName",  
  
}
```

Name - the unique name that will be assigned to the object.

TargetDescr - name of the goal description (details in the Descriptions of goals), which determines the position of the object being created.

DescrName - the name of the character description (more in Character Descriptions *).

CreateCargoBox

Creation of a box with the goods.

```

{

    "ActionType": "CreateCargoBox",

    "Name": "ExampleNPCName",

    "TargetDescr": "ExampleTargetName"

    "DescrName": "ExampleCargoboxName",

    "Velocity": 0,

    "RotateImpulse": 0,

    "AngleDirection": [0, 0, 0],

    "KillUnderGround": true,

    "UseGravity": true,

}

```

Name - the unique name that will be assigned to the object.

TargetDescr - name of the goal description (details in the Descriptions of goals), which determines the position of the object being created.

DescrName - the name of the container description (for details see Container Descriptions).

Velocity - (optional)

RotateImpulse - (optional)

AngleDirection - (optional)

KillUnderGround - (optional)

UseGravity - (optional)

RemoveEntity

Deleting an object.

```
{  
  
    "ActionType": "RemoveEntity",  
  
    "Name": "ExampleEntityName",  
  
}
```

Name - the name of the object.

SetBelongContainer

Setting the ownership (ownership) of the container to the player, world or city.

If the container belongs to the world or the player, it can be taken, it can break.

```
{  
  
    "ActionType": "SetBelongContainer",  
  
    "Name": "ExampleContainerName",  
  
    "CargoBelongType": "PLAYER",  
  
}
```

Name - the name of the object.

CargoBelongType - type of ownership: CITY - to the city, PLAYER - to the player, WORLD - to the world.