**4.1.1** CLASS **ActionType**

Возможные действия игрока.

Игрок не может совершить новое действие, если в течение последних game.actionDetectionInterval - 1 игровых тиков он уже совершил максимально возможное для него количество действий. В начале игры это ограничение для каждого игрока равно game.baseActionCount. Ограничение увеличивается за каждый контролируемый игроком центр управления (FacilityType.CONTROL\_CENTER).

Большинство действий требует указания дополнительных параметров, являющихся полями объекта move. В случае, если эти параметры установлены некорректно либо указаны не все обязательные параметры, действие будет проигнорировано игровым симулятором. Любое действие, отличное от NONE, даже проигнорированное, будет учтено в счётчике действий игрока.

DECLARATION

public final class ActionType

**extends** Enum

FIELDS

* public static final ActionType NONE **–** Ничего не делать.
* public static final ActionType CLEAR\_AND\_SELECT
  + Пометить юнитов, соответствующих некоторым параметрам, как выделенных. При этом, со всех остальных юнитов выделение снимается. Юниты других игроков автоматически исключаются из выделения.
* public static final ActionType ADD\_TO\_SELECTION
  + Пометить юнитов, соответствующих некоторым параметрам, как выделенных. При этом, выделенные ранее юниты остаются выделенными. Юниты других игроков автоматически исключаются из выделения.
* public static final ActionType DESELECT
  + Снять выделение с юнитов, соответствующих некоторым параметрам.
* public static final ActionType ASSIGN
  + Установить для выделенных юнитов принадлежность к группе.
* public static final ActionType DISMISS
  + Убрать у выделенных юнитов принадлежность к группе.
* public static final ActionType DISBAND
  + Расформировать группу.
* public static final ActionType MOVE
  + Приказать выделенным юнитам меремещаться в указанном направлении.
* public static final ActionType ROTATE
  + Приказать выделенным юнитам поворачиваться относительно указанной точки.
* public static final ActionType SCALE
  + Масштабировать формацию выделенных юнитов относительно указанной точки.
* public static final ActionType SETUP\_VEHICLE\_PRODUCTION
  + Настроить производство нужного типа техники на заводе (FacilityType.VEHICLE\_FACTORY).
* public static final ActionType TACTICAL\_NUCLEAR\_STRIKE **–** Запросить тактический ядерный удар.

**4.1.6** CLASS **Move**

Стратегия игрока может управлять юнитами посредством установки свойств объекта данного класса.

DECLARATION

public class Move **extends** Object

METHODS

* *getAction* public ActionType **getAction**( )
  + **Returns** - Возвращает текущее действие игрока.
* *getAngle* public double **getAngle**( )
  + **Returns** - Возвращает текущий угол поворота.
* *getBottom* public double **getBottom**( )
  + **Returns** - Возвращает текущую нижнюю границу прямоугольной рамки, предназначенной для выделения юнитов.
* *getFacilityId* public long **getFacilityId**( )
  + **Returns** - Возвращает текущий идентификатор сооружения.
* *getFactor* public double **getFactor**( )
  + **Returns** - Возвращает текущий коэффициент масштабирования.
* *getGroup* public int **getGroup**( )
  + **Returns** - Возвращает текущую группу юнитов.
* *getLeft* public double **getLeft**( )
  + **Returns** - Возвращает текущую левую границу прямоугольной рамки, предназначенной для выделения юнитов.
* *getMaxAngularSpeed* public double **getMaxAngularSpeed**( )
  + **Returns** - Возвращает текущее абсолютное ограничение скорости поворота.
* *getMaxSpeed* public double **getMaxSpeed**( )
  + **Returns** - Возвращает текущее ограничение линейной скорости.
* *getRight* public double **getRight**( )
  + **Returns** - Возвращает текущую правую границу прямоугольной рамки, предназначенной для выделения юнитов.
* *getTop* public double **getTop**( )
  + **Returns** - Возвращает текущую верхнюю границу прямоугольной рамки, предназначенной для выделения юнитов.

*getVehicleId* public long **getVehicleId**( )

* + **Returns** - Возвращает текущий идентификатор техники.
* *getVehicleType* public VehicleType **getVehicleType**( )

**– Returns** - Возвращает текущий тип техники.

* *getX* public double **getX**( )
  + **Returns** - Возвращает текущую абсциссу точки или вектора.
* *getY* public double **getY**( )
  + **Returns** - Возвращает текущую ординату точки или вектора.
* *setAction* public void **setAction**( ActionType **action** )
  + **Usage**

∗ Устанавливает действие игрока.

* *setAngle* public void **setAngle**( double **angle** )
  + **Usage**

∗ Задаёт угол поворота.

Является обязательным параметром для действия ActionType.ROTATE и задаёт угол поворота относительно точки (x, y). Положительные значения соответствуют повороту по часовой стрелке.

Корректными значениями являются вещественные числа от -PI до PI включительно.

* *setBottom* public void **setBottom**( double **bottom** )
  + **Usage**

∗ Устанавливает нижнюю границу прямоугольной рамки для выделения юнитов.

Является обязательным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT, если не установлена группа юнитов. В противном случае граница будет проигнорирована.

Корректными значениями являются вещественные числа от top до game.worldHeight включительно.

* *setFacilityId* public void **setFacilityId**( long **facilityId** )
  + **Usage**

∗ Устанавливает идентификатор сооружения.

Является обязательным параметром для действия

ActionType.SETUP\_VEHICLE\_PRODUCTION. Если сооружение с данным идентификатором отсутствует в игре, не является заводом по производству техники

(FacilityType.VEHICLE\_FACTORY) или принадлежит другому игроку, то действие будет проигнорировано.

* *setFactor* public void **setFactor**( double **factor** )
  + **Usage**

∗ Задаёт коэффициент масштабирования.

Является обязательным параметром для действия ActionType.SCALE и задаёт коэффициент масштабирования формации юнитов относительно точки (x, y). При значениях коэффициента больше 1.0 происходит расширение формации, при значениях меньше 1.0— сжатие.

Корректными значениями являются вещественные числа от 0.1 до 10.0 включительно.

* *setGroup* public void **setGroup**( int **group** )
  + **Usage**

∗ Устанавливает группу юнитов для различных действий.

Является опциональным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT. Если для этих действий группа юнитов установлена, то параметр vehicleType, а также параметры прямоугольной рамки left, top, right и bottom будут проигнорированы.

Является обязательным параметром для действий ActionType.ASSIGN,

ActionType.DISMISS и ActionType.DISBAND. Для действия ActionType.DISBAND является единственным учитываемым параметром.

Корректными значениями являются целые числа от 1 до game.maxUnitGroup включительно.

* *setLeft* public void **setLeft**( double **left** )
  + **Usage**

∗ Устанавливает левую границу прямоугольной рамки для выделения юнитов.

Является обязательным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT, если не установлена группа юнитов. В противном случае граница будет проигнорирована.

Корректными значениями являются вещественные числа от 0.0 до right включительно.

* *setMaxAngularSpeed* public void **setMaxAngularSpeed**( double **maxAngularSpeed** )
  + **Usage**

∗ Устанавливает абсолютное ограничение скорости поворота в радианах за тик.

Является опциональным параметром для действия ActionType.ROTATE. Если для этого действия установлено ограничение скорости поворота, то параметр maxSpeed будет проигнорирован.

Корректными значениями являются вещественные числа в интервале от 0.0 до PI включительно. При этом, 0.0 означает, что ограничение отсутствует.

* *setMaxSpeed* public void **setMaxSpeed**( double **maxSpeed** )
  + **Usage**

∗ Устанавливает абсолютное ограничение линейной скорости.

Является опциональным параметром для действий ActionType.MOVE, ActionType.ROTATE и ActionType.SCALE. Если для действия ActionType.ROTATE установлено ограничение скорости поворота, то этот параметр будет проигнорирован.

Корректными значениями являются вещественные неотрицательные числа. При этом, 0.0 означает, что ограничение отсутствует.

* *setRight* public void **setRight**( double **right** )
  + **Usage**

∗ Устанавливает правую границу прямоугольной рамки для выделения юнитов.

Является обязательным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT, если не установлена группа юнитов. В противном случае граница будет проигнорирована.

Корректными значениями являются вещественные числа от left до game.worldWidth включительно.

* *setTop* public void **setTop**( double **top** )
  + **Usage**

∗ Устанавливает верхнюю границу прямоугольной рамки для выделения юнитов.

Является обязательным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT, если не установлена группа юнитов. В противном случае граница будет проигнорирована.

Корректными значениями являются вещественные числа от 0.0 до bottom включительно.

* *setVehicleId* public void **setVehicleId**( long **vehicleId** )
  + **Usage**

∗ Устанавливает идентификатор техники.

Является обязательным параметром для действия ActionType.TACTICAL\_NUCLEAR\_STRIKE. Если юнит с данным идентификатором отсутствует в игре, принадлежит другому игроку или цель удара находится вне зоны видимости этого юнита, то действие будет проигнорировано.

* *setVehicleType* public void **setVehicleType**( VehicleType **vehicleType** )
  + **Usage**

∗ Устанавливает тип техники.

Является опциональным параметром для действий ActionType.CLEAR\_AND\_SELECT, ActionType.ADD\_TO\_SELECTION и ActionType.DESELECT. Указанные действия будут применены только к технике выбранного типа. Параметр будет проигнорирован, если установлена группа юнитов.

Является опциональным параметром для действия

ActionType.SETUP\_VEHICLE\_PRODUCTION. Завод будет настроен на производство техники данного типа. При этом, прогресс производства будет обнулён. Если данный параметр не установлен, то производство техники на заводе будет остановлено.

* *setX* public void **setX**( double **x** )

**– Usage**

∗ Устанавливает абсциссу точки или вектора.

Является обязательным параметром для действия ActionType.MOVE и задаёт целевую величину смещения юнитов вдоль оси абсцисс.

Является обязательным параметром для действия ActionType.ROTATE и задаёт абсциссу точки, относительно которой необходимо совершить поворот.

Является обязательным параметром для действия ActionType.SCALE и задаёт абсциссу точки, относительно которой необходимо совершить масштабирование.

Является обязательным параметром для действия ActionType.TACTICAL\_NUCLEAR\_STRIKE и задаёт абсциссу цели тактического ядерного удара.

Корректными значениями для действия ActionType.MOVE являются вещественные числа от -game.worldWidth до game.worldWidth включительно. Корректными значениями для действий ActionType.ROTATE и ActionType.SCALE являются вещественные числа от -game.worldWidth до 2.0 \* game.worldWidth включительно. Корректными значениями для действия ActionType.TACTICAL\_NUCLEAR\_STRIKE являются вещественные числа от 0.0 до game.worldWidth включительно.

• *setY* public void **setY**( double **y** )

**– Usage**

∗ Устанавливает ординату точки или вектора.

Является обязательным параметром для действия ActionType.MOVE и задаёт целевую величину смещения юнитов вдоль оси ординат.

Является обязательным параметром для действия ActionType.ROTATE и задаёт ординату точки, относительно которой необходимо совершить поворот.

Является обязательным параметром для действия ActionType.SCALE и задаёт ординату точки, относительно которой необходимо совершить масштабирование.

Является обязательным параметром для действия ActionType.TACTICAL\_NUCLEAR\_STRIKE и задаёт ординату цели тактического ядерного удара.

Корректными значениями для действия ActionType.MOVE являются вещественные числа от -game.worldHeight до game.worldHeight включительно. Корректными значениями для действий ActionType.ROTATE и ActionType.SCALE являются вещественные числа от -game.worldHeight до 2.0 \* game.worldHeight включительно. Корректными значениями для действия ActionType.TACTICAL\_NUCLEAR\_STRIKE являются вещественные числа от 0.0 до game.worldHeight включительно.