duodecim

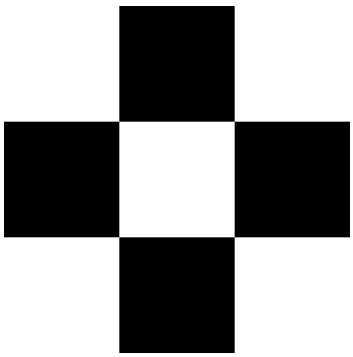
RPG Maker MV Tactics System

El Moussaoui Bilal
: @embxii_

https//:forums.rpgmakerweb.com/index.php?threads/tactics-system-ignis

26 juin 2018 | 0.1 Ignis

Thank you for downloading the Tactics System. This document is a user guide to quickly configure the Tactics System in a new project. Please read the instructions carefully before using this system. The Tactics System is a tactical battle system for RPG Maker MV inspired by Final Fantasy Tactics and the Fire Emblem series.



Basics

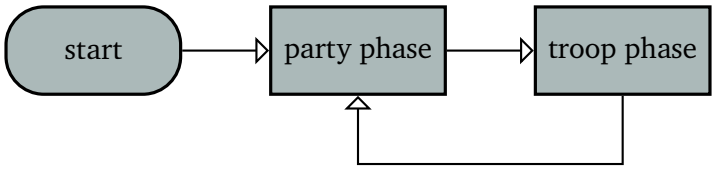
Prepare Yourself !

The **Tactics System** allows the creation of *tactical-RPG* with **RPG Maker MV** [1] software. Concessions were made to better adhere to the **RPG Maker** philosophy. The **Tactics System** has been designed to be **easy to use**. Just create a map with events defining the position of units and use a plugin command to launch a fight !

An example of how to use the system is provided in the forum. The **Tactics System** will be updated regularly, so please check the forum [2]regularly.

System Flow

The fight proceeds in the same way as an episode of the series Fire Emblem : we have the player phase and the enemy phase.



The agility statistics of a unit doesn't influence the order of action of the units. The player decides which unit moves first. The player phase ends when all player's units have completed their action.

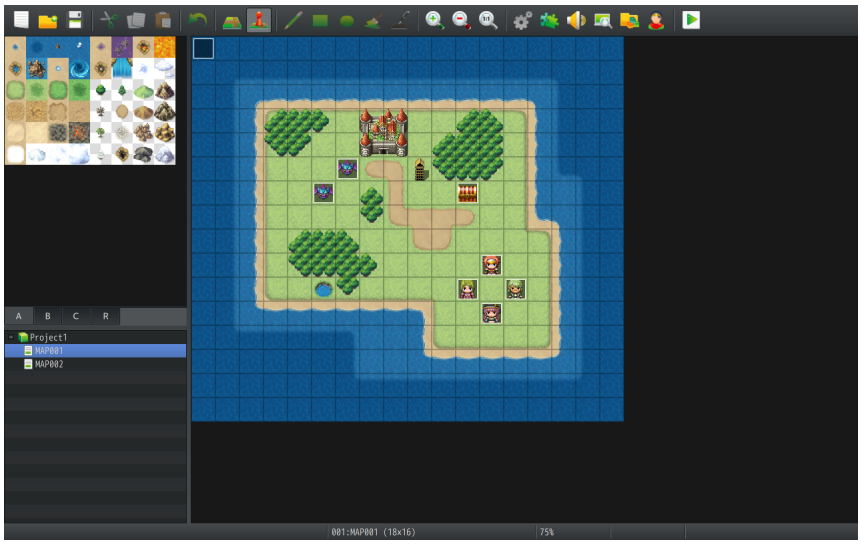
Condition of victory and defeat

The conditions of victory and defeat are simple : the player wins if all enemy's units are dead, the player loses if all his units are dead.

1. Version 1.5.2
2. https// :forums.rpgmakerweb.com/index.php ?threads/tactics-system-ignis

Battle Map

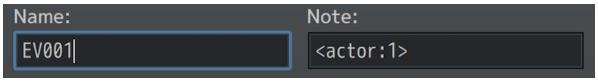
Player and enemy units are positioned with events. The command to start the fight is simple ! It consists of a plugin command contained in an event.



Units

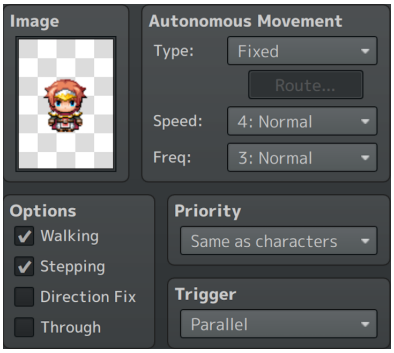
Actor

We need to bind the events to the actors in the database. The bond of the event to an actor is realized with the following note : `<actor:id>`



The identifier represents that of the actor in the database.

The image representing the actor is not directly associated with it. You must manually display the correct image in the event. This choice of design is motivated to be able to see the unit used directly on the map.



Warning, an error occurs if the actor is not part of the party or if several events are bound to the same actor !

Enemy

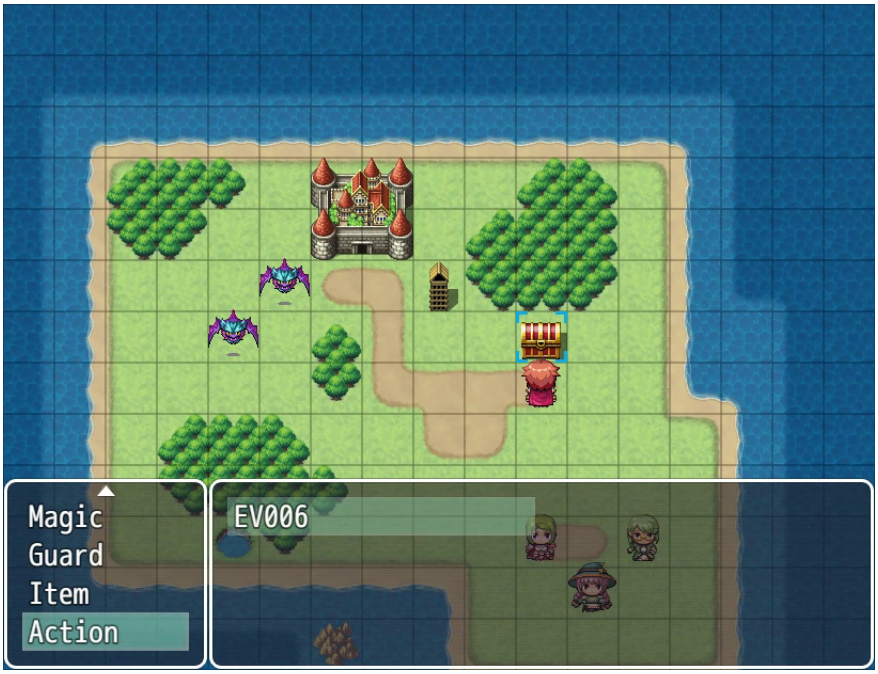
The configuration of the enemies follows the same logic as those of the actors. Except the command in the note section is : `<enemy:id>`.

Start Battle

To start a fight, use the following command : `startBattleTS`. This command is blocking. At the end of the combat, the system resumes the flow of the event used. In the example below, the player will be transferred to the map `MAP002` after the fight. All other events are being updated in parallel.

Action

Events are configured as classic events. The **trigger button** of an event allows units to activate it when they are at an adjacent cell. So you can easily configure switches, doors, chests, characters and more.



The event name that appears in the target window is the name of the event.

DATABASE

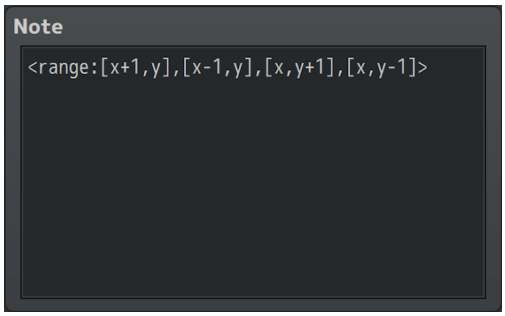
Movement

Movement defines a unit's ability to move. It is easily defined by the `<move:int>` command in the actors or enemies note section.

Note, that the player's units can through each other to reach a destination.

Range

The **Tactics System** allows your units to use skills and objects. The configuration of a skill or object is done as in the basic system. A new attribute is the scope of the action. It is defined as `<range:coordinates>` in the note section of a skill or object.



The coordinates x and y indicate the current position of the unit. In the image, the range defined above indicates the four adjacent cells of the unit. The cell at the top left is defined by `[x-1, y-1]` and the cell at the bottom right is defined by `[x+1, y+1]`.