

About This Proposal

This is a proposal for the addition of a **Turn-Based Grid Battle System** for use in the ORK Framework for Unity. Ideally, it will provide enough base mechanics to implement a battle system similar to Strategy/Tactics RPGs such *Final Fantasy Tactics*, *Fire Emblem*, *Tactics Ogre*, and *Disgaea*. It would be an excellent addition to ORK, and also a good selling point as well.

GLOBAL GRID FEATURES

-Grid Settings added to the Editor settings > Battle Settings. This would include global settings for the Grid used in the game, such as:

-Individual Cell Size: How large will each Cell be represented as.

-Square or Hexagonal Grid

-Integration of Phase or Regular Turn-based Battles in Grid

Phase Type (Fire Emblem, where each Faction takes turns moving and attacking with all of their units at once) **or Standard Turn-based Type** (Final Fantasy Tactics, where a formula (i.e Speed stat) determines the turn order of every unit on the field).

-Cell Types: A separate Tab List of every Cell type being used in your project. Each Cell on a Grid will be spawned a prefab according to its Cell Type. You will be able to assign Prefabs and movement-specific properties to individual Cells here. A few examples:

-Movement Cell: An empty cell that Units can move through or stop on. You can assign Terrain-type for different Movement cells (Sand, Shallow Water, etc.), which can incur a movement penalty if the player tries to cross it.

-Obstacle/Wall Cell: A cell that represents a wall or object that cannot be traversed.

-Deployment Cell: Whenever a battle begins and the player is choosing which units/characters to bring, the Player will be free to place them in any formation or order on these designated cells on the grid. (Think of the Beginning Deployment phase of FFT or Fire Emblem).

SETTING UP THE GRID BATTLES

-**Grid**: Similar to setting up a Turn-based Battle/Real-time Battle, the player would be able to set up a 'Grid Battle' in the scene, as a new component to define the size of the grid, and where it will physically be placed in the scene.

-After the Grid has been created in-scene, You will be able to change each individual cell to the type of your choosing.

-**Setting up Starting Combatants** will be done in a similar way, assigning Spawners to specific cells for Enemies/Combatants that are already on the grid when the battle begins.

-**Events** can be placed over the grid, triggered by stepping on them. Depending on whether it's already possible via existing methods, there may need to be additional conditions on Starting Events, such as *'Unit needs to End Turn within Trigger/Cell'*.

NEW COMMANDS AND ACTIONS

Move Command

-Moves the selected combatant to a different cell on the grid.

-Selecting a Combatant and/or the Move command highlights the moveable cells on the grid (using the defined prefab.)

-The Move target on the grid can be selected by click/touch on a cell (Using a mouse or touch) and through input keys (i.e. selecting a cell through a cursor with horizontal/vertical keys)

-Abilities can be used before and after the move command.

-Combatants get updated move range values (e.g. using a formula) and there are move range bonuses in the bonus settings.

-Naturally, battle AI will be integrated with the Move command.

Turn End Orientation

-If enabled, after a combatant finishes its turn, they will be able to selection which way the combatant is facing.

-Player controlled combatants can select the orientation (e.g. click/touch or input keys)

-AI controlled combatants will turn to a target based on their AI settings (e.g. nearest target, last target, etc.)

Situational Custom Commands on the Battle Menu and Grid Queries

-In addition to the typical Move, Act, Wait, etc, you can give Units situational commands. (This may be somewhat possible via Target requirements but there may need to be additional conditions:)

-A method in which combatants will be able to use additional custom commands or trigger events, depending on fulfilling one or all of a set conditions, such as:

-Being within Specific Area/Cells

-Being directly adjacent to a specified Combatant or Cell. (or not!)

-Are any adjacent cells occupied/unoccupied by combatants? How many?

-Being a specific Class/Combatant

-Variable or Status Checks

-Example: A Player's combatant moves next to a recruitable enemy combatant, and fulfils the needed conditions, the TALK command appears on the player's Battle Menu.

You can choose to 'Talk' to the unit, starting an event, and possibly converting the enemy combatant to your own side.

-This would be used in a wide variety of situations common in Strategy RPGs, from pulling levers, to talking to allies, to taking over a fortress and ending the battle.

Spawning and Moving Combatants to different Nodes via Events

For example, if there were a cutscene mid-battle, where a combatant moved to a different Cell or appeared on a new cell. Likewise, Spawning new combatants mid-battle would probably need to be oriented or directed into cells.

NEW UI/MENU COMPONENTS

Set-up Phase/Starting Formation

In most Tactics RPGs, when you begin a battle, you select from a list of Allies/Units to bring into battle, and place them down into a starting formation on the grid before the battle begins.

-This is a pretty important aspect of the genre and a UI/menu components would need to be implemented with this in mind.

Targeting Confirmation Prompt

There would need to be an optional confirmation prompt/menu upon targeting a Combatant or Cell with an ability. In most Tactics RPGs, it quickly compares Stats between the two Combatants, [along with other important information, like a Hit Chance % and estimated damage.](#)

GRID CURSOR AND HIGHLIGHTING SPECIFICS (The Telegraph System)

Highlighting Grid via Events

There would need to be a method where you can highlight specific cells via events. (This entire area is about to get hit by a meteor so avoid this, go to this spot, etc)

Highlight Colors and Prefabs

You will be able to select either the Colors, or what the Prefab of the Highlight object will be for Movement Highlights, and Ability/Action Highlights. This allows you to create a 'Highlight Animation' (such as a moving gradient/sprite animation) and set it as a prefab to appear over highlighted cells.

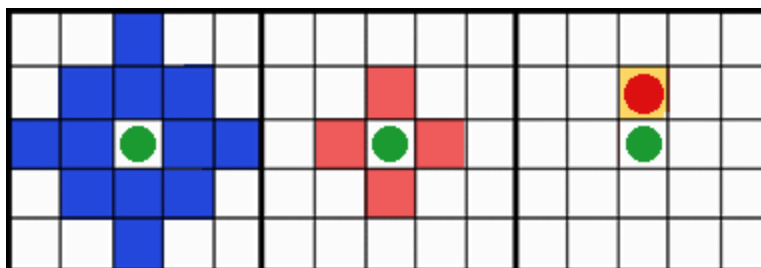
-The Cell that the cursor is currently over will be optionally highlighted.

-Three types of Highlights:

Movement Range (Blue)

Ability Range (Red)

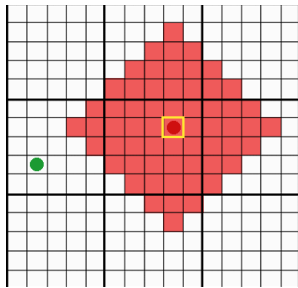
Target Selected/Will be affected by ability (Target/s's grid color will change to Yellow on confirmation prompt/UI for additional clarity. etc)



Examining Combatants and their Movement Range (and Effective Attack Range)

Being able to move the cursor around the map, and display properties of any given Combatant by selecting them and opening up a Menu (examining them, seeing what they have equipped, their name, their Stats, etc) or any important information about a particular cell (terrain type, etc).

Most importantly, placing your cursor over Enemy combatants can optionally highlight their ***EFFECTIVE*** attack range on their next turn (i.e optimal Movement + Attack range) with their currently equipped items. (and option for just movement range) This mechanic is common, and essential in helping Players gauge where it is safe to move without being at threat of attack.



Move Command

As mentioned previously, using the move command on a controllable combatant will highlight all cells that the combatant is capable of moving to.

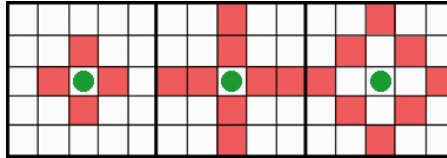
ABILITIES ON THE GRID

Ability Range

Abilities will have several new options for Grid Battles.

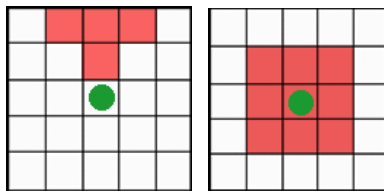
From what Distance can the User target target another cell? There are two kinds: Basic and Shape.

BASIC: Minimum and Maximum range that can be determined by set values or formulas. Would be set to 0 if the ability is Self-targeting only. This is good for simple, algorithmic ranges in four directions.



(For example, a Combatant with a sword can strike only adjacent cells. (Range of 1-1) A Combatant with a Spear can strike up to two cells in any direction. (Range of 1-2) But a Combatant with a Bow can't attack any adjacent cells, limited only to a (2-2) range of cells.

SHAPE: If a range's shape is more unorthodox, then they can paint a shape in the editor to show the range of the ability. For instance, if the range is in a cone shape, or a 3x3 box with the User at the center.



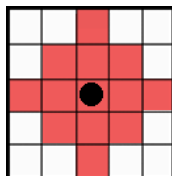
-Direction: Is the Ability limited to a single direction or can it be used in all four directions?

-AREA OF EFFECT SHAPES:

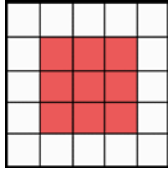
When an Ability affects more than a single Cell, it has an Area of Effect Shape.

You will be able to paint individual cells in an editor to define Ability Grid shapes, which can be used by abilities/items to determine which cells around a target will be affected. Could be extended to perhaps include formulaic or algorithmic targeting as well.

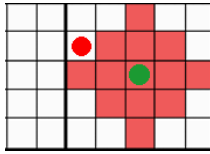
For example:



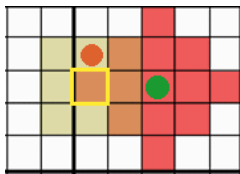
We are making a 'Fireball' Ability. First, we determine the Range. 0-2 Cells.



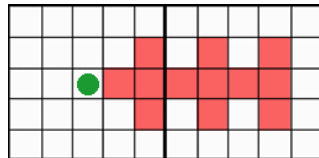
Next, we determine the Area of Effect by painting the shape. A 3x3 box.



Here, we have a scenario where Green has fireball, but the Range of Fireball cannot hit Red. However...



By targeting an empty cell, the Area of Effect of Fireball, the 3x3 Box, hits Red.



Using Range and Area of Effect, you can create interesting and varied ability types.

Ability Range Highlighting

Selecting an ability (Attack, Magic, Items) will highlight the range of the ability on the grid.

Ability Range Target Selection

Different abilities may affect more than one cell.

Ability Targeting: Empty Cells vs Combatant

Abilities can require that they have a valid Combatant target in order to be available for use. If disabled, then an ability can attempt to use an ability on an empty grid.

Some Examples:

-A Combatant cannot use his ability unless there is an actual combatant within range of the ability.

-A Combatant is free to cast a spell/ability on any Cell within range, even empty ones, as long as the Area of Effect shape hits at least one combatant.

-A Combatant can cast a spell/ability on any Cell within range, even if it's empty. (In this case, the Cell MUST be empty for the player to lay down a Hidden Trap on the empty cell.)

GRID-CENTRIC ABILITIES

Abilities can optionally move players around on the grid.

Some examples:

'Shove' Ability can push a combatant backward (opposite the direction of the ability) by one cell.

'Rescue' Spell teleports a friendly combatant to a cell adjacent to the spellcaster.

'Cut-Through Attack' moves the user directly behind the target he attacks.

'Swap' Spell swaps places on the grid with the user and an enemy combatant.

etc. Lots of possibilities for interesting stuff.

2D SUPPORT

Support for 2D (and hopefully 2DTK!) sprites.

Changing Animations/Prefab/Appearance after a turn is complete.

Adding an effect that visually indicates that a player's turn is over. For instance, the player's combatants becoming Grayed out once their turn has been played.

ADDITIONAL IDEAS

Support for Large Combatants that take up more than 1 Cell.

Some larger combatants (say, an enormous minotaur that takes up 2x2 Cells or a Mecha that could take up 2x1 Cells of space on the Grid.

Height Mechanics

Having a map filled with cells of varying heights is a common sight in 3D Tactics games.

- Having a height variable per Cell to signify 'height'.
- Changing combatants' Y position depending on the height of the current tile.
- Adding the current Cell Height as a variable for damage calculations.
- Adding Height to Range calculations.
- Obstacle Calculation: 'If there's a wall between an enemy and my unit with a height of 10, and the highest my Bow and Arrow can go is 5, my arrow will strike the wall instead.'