Parameters in map_gen.ini
version 1.1.0

Ulrich Krüger

2017-Apr-08

General remarks

This document lists *all* parameters which are recognized by map_gen 1.1.0. Not each parameter is used in every scenario. If you use a parameter in a scenario which doesn't need it, the scenario simply ignores your parameter.

You will also find some 'hidden' parameters which I didn't mention before. Technically, they are in no way different to 'regular' parameters, and there is no reason why you shouldn't use them. I use them to fine-tune some scenarios without the need to recompile map_gen. Moreover, some scenarios rely on the default values of these 'hidden' parameters.

Some parameters control distances on the map. These parameters always use the old movement point definition, i. e. the distance of a tile to the four tiles with which it has an edge in common (north-east, south-east, south-west and north-west) is 100, while the distance to the four tiles touching one corner (north, south, east and west) is 150, no matter what the terrain is.

Parameter types

The 'Type' column in the table below tells the data type of the respective parameter. That is the type as it is defined in the source code.

U8 An unsigned 8-bit parameter

U16 An unsigned 16-bit parameter

BIT A boolean parameter. Despite the fact that the table lists 'FALSE' as default value, you cannot specify symbolic values in the configuration file, you have to specify numbers (decimal constants). This might change in future releases.

char∗ A string parameter

Default values

Each parameter has a default value which is used when the parameter is not given in the configuration file map_gen.ini. If you know what you're doing, you may omit a parameter when you want its default value. But be careful: some scenarios change that value to a scenario-specific default value (which is not listed here). Therefore it is safest to write the parameter values you want into the configuration file.

Default values are used to supply parameters to scenarios when I guessed those parameters might irritate the user.

Min/Max values

Each integer parameter has an allowed range defined by the minimum and maximum value. Specifying a value outside this range causes a syntax error. Specifying a value *inside* this range doesn't guarantee that the respective scenario likes that value; i. e. there might be additional restrictions introduced by the scenario. Moreover, when a parameter is used in more than one scenario, these additional restrictions might be different.

String parameters have dummy min/max values which are zero. There is no range checking for string parameters.

Parameter listing

Name	Type	Default value	Minimum value	Maximum value	Description
allow_lakes_near_poles	BIT	FALSE	0	1	In the GreatPlains scenario, lakes are usually not generated in the northern and southern mountain stripes. This is to generate more mountains on small maps, so that all special resources can be placed. When you set allow_lakes_near_poles=1, lakes might also appear in the mountains. It looks nice, and it should be no problem on large maps.
basic_probability_desert	U16	0	0	10000	Used by the Navigation_reqired scenario which changes the default value to 50. Also used by the Micronesia scenario which changes the default to zero. When generating random land tiles, this parameter controls the probability for choosing 'desert'. Random land tile generation depends on the tile's environment in city radius. E. g., a desert will never be placed right beside an arctic tile. Generally, higher values mean a higher probability. Please note this is not a percentage value. Furthermore, this value will be mangled according to the tile's environment. Setting this parameter to zero does *not* guarantee that no desert will be placed: If there are already some desert tiles inside the city radius, the probability is increased by the algorithm before the tile is chosen. Hence the name, *basic_*probability, since the algorithm uses this parameter as a basis for its calculations.
basic_probability_grassland	U16	0	0	10000	For a description see "basic_probability_desert". Used in all scenarios which use random tile generation which depends on the tile's environment in city radius. These are: Navigation_required, Micronesia, Hard_fight, Big_river and Arctic.
basic_probability_mountains	U16	0	0	10000	Analogous to basic_probability_grassland.
basic_probability_prairie	U16	0	0	10000	Used by the Navigation_reqired scenario which changes the default value to 100. Also used by the Micronesia scenario which changes the default to zero. For a description see "basic_probability_desert".

Name	Type	Default value	Minimum value	Maximum value	Description
basic_probability_tundra	U16	0	0	10000	Used by the Navigation_reqired scenario which changes the default value to 20. Also used by the Micronesia scenario which changes the default to zero. For a description see "basic_probability_desert".
big_island_dist	U16	200	100	10000	The mutual distance of the islands for computer opponents and the human player in the Navigation_required scenario.
big_island_max_size	U16	300	1	9600	Used by the Arctic scenario for the generation of 'southern land'. Southern land is an aggregation of islands which are allowed to merge together. This parameter determines the upper limit (in tiles) for the size of these islands. The default value is changed to 20 by the Arctic scenario.
big_island_min_size	U16	200	1	9600	Lower limit for island size, see big_island_max_size. The default value is changed to 2 by the Arctic scenario.
comp_island_max_size	U16	300	1	9600	When randomly choosing the size of an island for computer opponents in the Navigation_required scenario, this is the maximum value (in tiles).
comp_island_min_size	U16	200	1	9600	When randomly choosing the size of an island for computer opponents in the Navigation_required scenario, this is the minimum value (in tiles).
comp_opponents	U8	3	1	14	The number of starting positions for the computer opponents.
comp_opponents_area1	U8	3	0	15	In the Big_river scenario, this parameter determines how many computer opponents will be placed north of the river.
comp_opponents_area2	U8	3	0	15	In the Big_river scenario, this parameter determines how many computer opponents will be placed south of the river.
forest_max_size	U16	8	1	9600	When placing random clusters of forest, this parameter determines their maximum size (in tiles). Used in the Great_plains scenario. Please note that clusters may merge together, i.e. one may observe bigger clusters on the generated map; these are two or more clusters merged together.
forest_min_size	U16	2	1	9600	When placing random clusters of forest, this parameter determines their minimum size (in tiles). Used in the Great_plains scenario. This is no strict lower limit; think of it as a recommendation you give to the algorithm.

Name	Type	Default value	Minimum value	Maximum value	Description
forest_percentage	U8	1	0	100	When placing random clusters of forest, this parameter determines their cumulated amount as a percentage of the total numbers of tiles on the map. Used in the Great_plains scenario.
glacier_percentage	U8	2	1	30	Used in the "Mountains" scenarios. This parameter determines the cumulated area coverd by arctic tiles as a percentage of the total number of tiles on the map.
grassland_max_size	U16	8	1	9600	Used in the "Great_Plains" and "Desert" scenarios. When placing grassland clusters, this parameter determines the maximum size (in tiles) of the clusters.
grassland_min_size	U16	2	1	9600	Used in the "Great_Plains" and "Desert" scenarios. When placing grassland clusters, this parameter determines the minimum size (in tiles) of the clusters.
grassland_percentage	U8	2	0	99	Used in the "Great_Plains" and "Desert" scenarios. When placing grassland clusters, this parameter determines the amount of grassland tiles as percentage of total tiles on the map.
human_island_max_size	U16	35	1	9600	When randomly choosing the size of the island for the human player in the Navigation_required scenario, this is the maximum value.
human_island_min_size	U16	35	1	9600	When randomly choosing the size of the island for the human player in the Navigation_required scenario, this is the minimum value.
human_start_pos	U8	1	0	3	This parameter determines where the human player's starting position will be. In the Mountains scenario, setting human_start_pos=0 gives the human player a similar starting position as the computer players, that is, not in a 'faraway valley'. No other value is allowed in the Mountains scenario. In the "Big river" scenario, human_start_pos=1 places the human player on the north side, 2 on the south side, and 3 picks randomly either north or south side. With human_start_pos=0 no special starting position is chosen for the human player.
island_dist	U16	900	100	10000	Used in the Volcano scenario. This parameter determines the minimum mutual distance of the volcano islands.

Name	Type	Default value	Minimum value	Maximum value	Description
island_max_size	U16	35	1	9600	Used in the Volcano scenario. This parameter determines the maximum size (in tiles) of the volcano islands when randomly choosing their size.
island_min_size	U16	35	1	9600	Used in the Volcano scenario. This parameter determines the minimum size (in tiles) of the volcano islands when randomly choosing their size.
island_percentage	U8	10	1	100	Used in the Volcano scenario. This parameter determines the cumulated land mass of the volcano islands as a percentage of the total number of tiles on the map.
lake_dist	U16	1000	0	10000	When clusters of water (= lakes) are placed randomly, this is the minimum mutual distance of the lakes. Setting lake_dist=0 allows the lakes to merge together.
lake_percentage	U8	10	0	100	When clusters of water (= lakes) are placed randomly, this is the amount of lake tiles as a percentage of the total number of tiles on the map.
land_percentage	U8	10	0	100	Used in the Micronesia scenario. This parameter determines the total landmass (=islandmass) as a percentage of the total number of tiles on the map.
mapfile	char*	"map_gen.cevo map"	0	0	The name of the map file to be generated. CAUTION: A file with the same name as the one being generated is overwritten without notice!
map_size	U8	100	35	230	The size of the map being generated; analog to the 'size' parameter of C-Evo's built-in map generator. Map_size may be 35, 50, 70, 100, 150 or 230.
max_lake_size	U16	10	2	9600	When clusters of water (= lakes) are placed randomly, this is the maximum size (in tiles) of the lakes.
max_no_of_comp_islands	U16	4	2	100	Used in the Navigation_required scenario. When randomly choosing the number of islands suitable for computer opponents, this is the upper limit. This parameter must be greater than or equal to comp_opponents.

Name	Type	Default value	Minimum value	Maximum value	Description
max_no_of_glaciers	U16	5	1	100	Used in the "Mountains" scenario. This parameter limits the number of glaciers. Glaciers are clusters of arctic tiles. The actual number of glaciers is a random number between 1 and max_no_of_glaciers. Since glaciers can merge together (there is no minimum distance between glaciers) the number of glaciers you can count on the map might even be less; when two glaciers merge together they look like one single cluster.
max_no_of_small_islands	U16	8	0	100	Used in the Navigation_reqired scenario. When randomly choosing the number of small islands, this is the upper limit.
MHT_percentage	U8	2	0	100	Used in the Arctic scenario. This parameter determines the amount of mountain, hills and tundra tiles (MHT) in the northern and mid stripe, as a percentage of the total number of tiles in the northern and mid stripe.
min_lake_size	U16	10	1	9600	When clusters of water (= lakes) are placed randomly, this is the minimum size (in tiles) of the lakes. Not all scenarios keep strictly to this value; think of it as a recommendation you give to the algorithm.
min_no_of_comp_islands	U16	3	2	100	Used in the Navigation_required scenario. When randomly choosing the number of islands suitable for computer opponents, this is the lower limit. This parameter must be greater than or equal to comp_opponents.
min_no_of_small_islands	U16	4	0	100	Used in the Navigation_reqired scenario. When randomly choosing the number of small islands, this is the lower limit.
mountain_stripe	U8	10	1	20	Used in the Great_plains scenario. Near the top and bottom mountain stripes (see mountain_stripe_uncond), there is an area with a mix of mountains and prairie. This parameter determines its width. If this parameter is not given in map_gen.ini, the scenario uses its own default value of 12.
mountain_stripe_uncond	U8	10	1	20	Used in the Great_plains scenario. The top and bottom lines of the map are all mountain tiles. This parameter determines the width of the two stripes (in lines). If this parameter is not given in map_gen.ini, the scenario uses its own default value of 2.

Name	Type	Default value	Minimum value	Maximum value	Description
northern_harbors	U8	8	1	30	Used in the Arctic scenario. This is the number of places — located in the mid stripe — suitable for building a city with access to the open sea. The human player's starting position is at one of these places.
northern_percentage	U8	10	1	98	Used in the Arctic scenario. This parameter determines the height of the northern stripe, expressed as a percentage of the map's height.
no_southern_land	BIT	FALSE	0	1	Used in the Arctic scenario. Setting no_southern_land=1 suppresses the generation of southern land. All starting positions are moved to 'Northern harbors'. All special resources are placed in the northern stripe.
prairie_max_size	U16	8	1	9600	When placing random clusters of prairie, this parameter determines their maximum size (in tiles). Used in the Desert scenario. Please note that clusters may merge together, i.e. one may observe bigger clusters on the generated map; these are two or more clusters merged together.
prairie_min_size	U16	2	1	9600	When placing random clusters of prairie, this parameter determines their minimum size (in tiles). Used in the Desert scenario. This is no strict lower limit; think of it as a recommendation you give to the algorithm.
prairie_percentage	U8	2	0	99	When placing random clusters of prairie, this parameter determines their cumulated amount as a percentage of the total numbers of tiles on the map. Used in the Desert scenario.
river_visibility	U16	100	1	10000	This parameter influences how many tiles will be tiles with rivers. Map_gen uses an elevation/rain/erosion model to add rivers to the map. In this model, certain quantities of water flow thru each land tile. Only tiles with a throughput of river_visibility or more will become river tiles. Please note that not all scenarios implement rivers.
small_island_dist	U16	800	100	10000	In the Navigation_required scenario, this parameter determines the mutual distance of the small islands as well as their distance to the player's islands ('big islands').

Name	Type	Default value	Minimum value	Maximum value	Description
small_island_max_size	U16	11	1	9600	When randomly choosing the size of 'small islands' in the Navigation_required scenario, this is the upper limit (in tiles).
small_island_min_size	U16	2	1	9600	When randomly choosing the size of 'small islands' in the Navigation_required scenario, this is the lower limit (in tiles).
southern_percentage	U8	10	1	98	Used in the Arctic scenario. This parameter determines the amount of land in the southern stripe as a percentage of the total number of tiles *in the southern stripe*.
starting_pos_dist	U16	1000	100	10000	This parameter sets the guaranteed minimum distance between the player's starting positions. It is the minimum distance from center tile to center tile, i. e. 550 will guarantee there is no overlap of city radius areas.
startpos_min_rating	U8	41	8	255	This parameter sets the level for what is considered to be an acceptable starting position. Increasing this parameter demands better starting positions from the algorithm. Don't try it! Increasing this parameter will *not* result in better starting positions (when run again with the same seed value, i. e. when generating an identical map). The algorithm always picks the best positions available, and then checks their rating against startpos_min_rating. So increasing startpos_min_rating will not give you *better* starting positions, but *fewer* candidates. The default value guarantees a reasonable amount of resources: three or four excess food and three or more materials. Decreasing it will allow worse positions, hence *more* positions. Consider decreasing startpos_min_rating when you get the "can place only x starting positions" error message but don't want to change any other parameter. When I generate maps for myself, I either use the default value or set startpos_min_rating=8. This is the lowest value which guarantees a city (size 4) will not starve.

Name	Type	Default value	Minimum value	Maximum value	Description
water_width	U16	1000	0	10000	In the "Hard fight" scenario, this parameter determines the width of the water ring around each of the three islands. In scenarios with 'Big rivers' ("Big river" and "Great plains"), it determines the width of the 'Big River'. Set it to 650 or higher if you want Navigation to be required to cross the river.