Appendix: Task Configurations

Table of Contents

[Base Classes 3](#_Toc191365684)

[BlockDef 3](#_Toc191365685)

[TrialDef 4](#_Toc191365686)

[TaskDef 4](#_Toc191365687)

[Continuous Recognition 5](#_Toc191365688)

[BlockDef 5](#_Toc191365689)

[TrialDef 5](#_Toc191365690)

[TaskDef 6](#_Toc191365691)

[StimDef 6](#_Toc191365692)

[Effort Control 7](#_Toc191365693)

[BlockDef 7](#_Toc191365694)

[TrialDef 7](#_Toc191365695)

[TaskDef 7](#_Toc191365696)

[StimDef 7](#_Toc191365697)

[Flex Learning 8](#_Toc191365698)

[BlockDef 8](#_Toc191365699)

[TrialDef 8](#_Toc191365700)

[TaskDef 8](#_Toc191365701)

[StimDef 8](#_Toc191365702)

[Gaze Calibration 9](#_Toc191365703)

[TrialDef 9](#_Toc191365704)

[TaskDef 9](#_Toc191365705)

[Maze Game 10](#_Toc191365706)

[BlockDef 10](#_Toc191365707)

[TrialDef 10](#_Toc191365708)

[TaskDef 10](#_Toc191365709)

[MazeDef 11](#_Toc191365710)

[Touch-Hold-Release 11](#_Toc191365711)

[BlockDef 11](#_Toc191365712)

[TrialDef 12](#_Toc191365713)

[TaskDef 12](#_Toc191365714)

[Visual Search 14](#_Toc191365715)

[BlockDef 14](#_Toc191365716)

[TrialDef 14](#_Toc191365717)

[TaskDef 14](#_Toc191365718)

[StimDef 14](#_Toc191365719)

[What-When-Where 15](#_Toc191365720)

[BlockDef 15](#_Toc191365721)

[TrialDef 15](#_Toc191365722)

[TaskDef 15](#_Toc191365723)

[StimDef 15](#_Toc191365724)

[Working Memory 17](#_Toc191365725)

[BlockDef 17](#_Toc191365726)

[TrialDef 17](#_Toc191365727)

[TaskDef 17](#_Toc191365728)

[StimDef 17](#_Toc191365729)

This document begins by describing the configurable fields of M-USE’s base classes. After the base classes, it covers classes that have task-specific inheritances.

## Base Classes

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| BlockCount | int | A unique number that represents the block number of the specified block, corresponds with the TrialDef BlockCount. |
| BlockName | string | A unique string used to label different blocks. |
| ContextName | string | Refers to the filename of the PNG texture in the resources folder used during the trial. |
| MinTrials | int | The minimum number of trials in the block. |
| MaxTrials | int | The maximum number of trials in the block. |
| NumTrials | int | The exact number of trials in the block. |
| MinMaxTrials | int[] | An integer array specifying the minimum and maximum number of trials in the block.  \*Note: e.g., [5, 10] means a minimum of 5 trials in the block and a maximum of 10 trials in the block. |
| RandomMinMaxTrials | int[] | An integer array specifying the range for the random selection of the maximum number of trials in a block. \*Note: e.g., [5, 10] means a random number between 5 and 10 trials will run, and then the block will be forced to switch. Block switch can occur before this max number if criterion is met using BlockEndType, BlockEndThreshold, BlockEndWindow and the minimum number of trials is completed, in this example, 5 trials. |
| BlockEndType | string | A strategy defining when to end a block. Options include: CurrentTrialPerformance, SimpleThreshold, ThresholdAndPeak, or ThresholdOrAsymptote. |
| BlockEndThreshold | float | A specified value used in conjunction with the BlockEndType to determine when to conclude a block. |
| BlockEndWindow | int | The number of most recent trials evaluated against the block end threshold. |
| NumPulses | int | The number of pulses transmitted to the SyncBox when a pulse reward is given. |
| PulseSize | int | The magnitude of each pulse sent from the SyncBox for reward. |
| SliderInitialValue | int | The initial position or value of a slider used within the block. |
| NumInitialTokens | int | The initial tokens in the token bar at the start of the block. |
| TokenBarCapacity | int | The number of tokens that the token bar can hold.  Note: Ensure the value is less than 10 for CR |
| TrialDefs | List<TrialDef> | A list of trial definitions for the block. |
| RandomNumGenerator | int | Random number generator, used to select random number of max trials in the MinMaxTrials range. |
| TrialDefSelectionStyle | string | If set to “adaptive,” adaptive procedure takes place; if set to “default,” default procedure takes place. |
| MaxDiffLevel | int | The highest difficulty level of all trials in a block. For default blocks, you may this to -1. |
| AvgDiffLevel | int | Used as a seed to randomize the initial trial. For default blocks, you may set this to -1. |
| DiffLevelJitter | int | Defines range of randomization for initial trial. The first trial will have a difficultyLevel somewhere in the range of AvgDifficultyLevel - DiffLevelJitter, AvgDifficultyLevel + DiffLevelJitter. For default blocks, you may set this to -1. |
| NumReversalsUntilTerm | int | Defines the number of reversals needed to terminate a block. A reversal is defined as correctly completing a trial to incorrectly completing the next one, or vice versa. |
| MinTrialsBeforeTermProcedure | int | This is the user-defined minimum number of trials completed before the termination procedure takes place and reversals start being counted. |

### TrialDef

See **BlockDef**, in addition to the variables listed below.

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| TrialCountInBlock | int | The count of trials within a block. |
| TrialCountInTask | int | The count of trials within a task. |
| TrialID | string | A unique string identifier for a trial. |
| DifficultyLevel | int | Defines a trial’s difficulty level. Higher numbers correspond to higher difficulty. For trials in default blocks, you may set this to -1 |
| PosStep | int | This determines the number of DifficultyLevels up the staircase goes after a lower effort choice. For trials in default blocks, you may set this to -1 |
| NegStep | int | This determines the number of DifficultyLevels down the staircase goes after a higher effort choice. For trials in default blocks, you may set this to -1 |

### TaskDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| StimFacingCamera | bool | A boolean indicating if the stimuli should automatically orient to face the camera. |
| ShadowType | string | A string indicating the type of shadow the stimulus produces. Available options are: "None", "Soft", and "Hard". |
| NeutralITI | bool | A boolean denoting if a neutral texture should be presented during the task's Inter-Trial Interval. The "NeutralITI" texture from the resources is utilized for this purpose. |

Below are the listed configurable fields for every task implemented in M-USE, along with their type and description. All tasks’ **BlockDef**, **TrialDef**, **TaskDef,** and **StimDef** inherit from the base class detailed above.

## Continuous Recognition

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| ShakeStim | bool | A boolean that defines whether the stimuli will wobble in a small circle during the trial.  \*Note: Primarily used for added difficulty. |
| FindAllStim | bool | A boolean for whether or not the block can be completed by finding all stim. \*Default: Block is ended by completing all trials in the TrialDef. |
| UseStarfield | bool | A boolean that assigns the visibility of the Starfield (snow-like particle system).  \*Note: Primarily used for added difficulty. |
| ManuallySpecifyLocations | bool | A boolean for manually specifying stimulus locations in the BlockDef config file. Setting to true indicates that stimuli locations are specified in the BlockStimLocations field. |
| RewardMag | int | The token gain from a correct selection. |
| BlockStimIndices | int[] | An array of integers that indicate the stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| NumObjectsMinMax | int[] | The number of stim in the first and last trial. Total number of trials is calculated as (Num 2 – Num 1) + 1.  \*Note: Ensure the value for the second number is 24 or less. |
| InitialStimRatio | int[] | The percentage of PC, New, and PNC Stim.  \*Note: A ratio of [2, 1, 1] means 50% PC Stim, 25% New Stim and 25% PNC Stim. |
| BlockStimLocations | Vector3[] | The stimuli locations if you set the ManuallySpecifyLocations field to true. |
| X\_Locations | float[] | The stimuli X locations used to generate the stimulus locations. |
| Y\_Locations | float[] | The stimuli Y locations used to generate the stimulus locations. |
| X\_FbLocations | float[] | The feedback X locations used to generate feedback locations. |
| Y\_FbLocations | float[] | The feedback Y locations used to generate feedback locations. |
| slopeOfRewardIncreaseOverTrials | float | A coefficient describing the rate at which number of reward pulses increases in relation to trial number. If set to 0, reward pulses stay constant at 1 throughout trials; if set to 0.5, a reward pulse is added every ~4 trials; if set to 1, a reward pulse is added every ~3 trials; if set to 1.5, a reward pulse is added every ~2 trials; etc. |
| \*MaxTrials | int | The maximum number of trials in a block. Calculation depends on the type of block end: FindAllStim or CompleteAllTrials.  \*Note: Not a configurable variable. Calculated in script. |

### TrialDef

See **BlockDef**.

### TaskDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| \*TrialStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear.  \*Note: Not a configurable variable. Calculated in script. |
| \*NumTrialStims | int | The number of stimuli loaded for the trial.  \*Note: Not a configurable variable. Calculated in script. |

### StimDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| PreviouslyChosen | bool | A boolean that assigns if the stimulus has already been chosen during the block. |

## Effort Control

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| NumTrials | int | The number of trials in the block. |
| NumClicksLeft | int | The number of outlines for the balloon on the left. |
| NumClicksRight | int | The number of outlines for the balloon on the right. |
| NumCoinsLeft | int | The number of tokens shown above the balloon on the left. |
| NumCoinsRight | int | The number of tokens shown above the balloon on the right. |
| NumPulsesLeft | int | The number of pulses delivered for completed left balloon inflation. |
| NumPulsesRight | int | The number of pulses delivered for completed right balloon inflation. |
| PulseSizeLeft | int | The size of the pulses delivered for completed left balloon inflation. |
| PulseSizeRight | int | The size of the pulses delivered for completed right balloon inflation. |
| ClicksPerOutline | int | The number of clicks needed to inflate the balloon to the next outline |

### TrialDef

See **BlockDef**

### TaskDef

No additional variables used.

### StimDef

No additional variables used.

## Flex Learning

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| TrialStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, following the order in the TrialStimIndices. |
| TrialStimIndices | int[] | An array of integers that indicate the stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| TrialStimTokenReward | int[] | An array of integers that indicate the number of token value of each stimulus, assigned based on the order in the TrialStimIndices. |
| ProbabilisticTrialStimTokenReward | Reward[][] | An array indicating the likelihood of obtaining varying reward amounts for each stimulus, assigned based on the order in the TrialStimIndices. |
| ProbabilisticNumPulses | Reward[] | An array indicating the likelihood of receiving varied reward amounts when a reward pulse is issued. |
| RandomizedLocations | bool | A boolean indicating if the positions in TrialStimLocations are randomly allocated or are assigned as listed. |
| TokensWithStimOn | bool | A boolean that determines if the stimuli and associated feedback remain visible on screen while token rewards are being given.  \*Default: false. |

### TrialDef

See **BlockDef**.

### TaskDef

No additional variables used.

### StimDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| IsTarget | bool | A boolean that determines if the stimulus is the target object, based on a positive reward assignment. |

## Gaze Calibration

### TrialDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| CalibPointsInset | float[] | A float array that defines how far the calibration targets are inset from the screen edges as a proportion of the x and y axes (for example, [0.15, 0.15] places each point 15% away from the screen boundary in both directions). |
| MaxCircleScale | float | A float indicating the initial size of the calibration point before the shrinking animation. |
| MinCircleScale | float | A float indicating the final size of the calibration point after the shrinking animation. |
| ShrinkDuration | float | A float indicating the minimum duration of fixation (when the calibration point shrinks) before calibration begins. |
| NumPulses | int | The number of pulses transmitted to the SyncBox when a pulse reward is given. |
| PulseSize | int | The magnitude of each pulse sent from the SyncBox for reward. |

### TaskDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| RewardStructure | string | A string that determines how rewards are distributed throughout the gaze calibration. Either “OnCompletion,” “PerPoint,” or “None.” |

## Maze Game

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| MazeName | string | A unique string identifier for the maze. |
| MazeDims | Vector2 | A 2D vector that indicates the dimensions of the maze in terms of width and height. |
| MazeStart | string | A string identifier indicating the starting position within the maze (e.g., A1 represents the bottom left of the maze). |
| MazeFinish | string | A string identifier indicating the finishing or end position within the maze (e.g., A1 represents the bottom left of the maze). |
| MazeNumSquares | int | The total number of squares along the maze path. |
| MazeNumTurns | int | The total number of turns along the maze path. |
| ViewPath | bool | A boolean indicating if the chosen path through the maze should be visualized or highlighted for the participant. |
| RewardRatio | int | An integer indicating the fixed number of correct responses required before a reward is given. |
| ErrorPenalty | bool | A boolean indicating whether a penalty is applied for incorrect responses or errors. |
| FlashingTileRatio | int | Indicates the frequency of the tile flashing.  \*Note: The pattern always begins with the start tile flashing. A value of 0 indicates that there are no tiles along the path that are flashing. A value of 1 indicates that the next correct tile along the path will flash. A value of 2 indicates that every other tile will flash, etc.. |
| DefaultTileColor | float[] | The RGB values of the default color for the maze tiles.  \*Note: Values range from [0,1], divide integer values by 255 |

### TrialDef

See **BlockDef.**

### TaskDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| MazePosition | Vector3 | The 3D coordinates for the center of the maze within the scene. |
| MazeBackgroundTexture | string | The texture applied as the maze background. |
| SpaceBetweenTiles | float | The distance maintained between individual tiles in the maze. |
| TileSize | float | The size of individual tiles in the maze. |
| TileTexture | string | The texture or pattern applied to each maze tile. |
| StartColor | float[] | The RGB values of the color of the first tile in the hidden maze.  \*Note: Values range from [0,1], divide integer values by 255 |
| FinishColor | float[] | The RGB values of the color of the final tile in the hidden maze.  \*Note: Values range from [0,1], divide integer values by 255 |
| CorrectColor | float[] | The RGB values of the color displayed given a correct tile selection.  \*Note: Values range from [0,1], divide integer values by 255 |
| LastCorrectColor | float[] | The color displayed given the re-selection of the current position within the sequence and for the blinking of the next correct tile in the sequence.  \*Note: Values range from [0,1], divide integer values by 255 |
| IncorrectRuleAbidingColor | float[] | The color displayed given an incorrect selection that still adheres to established rules.  \*Note: Values range from [0,1], divide integer values by 255 |
| IncorrectRuleBreakingColor | float[] | The color displayed given an incorrect selection that breaks or violates established rules.  \*Note: Values range from [0,1], divide integer values by 255 |
| NumBlinks | int | The number of times a tile blinks or flashes. Used in GuidedMazeSelection or when indicating last correct tile following a perseverative error. |
| UsingFixedRatioReward | bool | Indicates if rewards are given based on a fixed ratio, which is assigned in the BlockDef. |

### MazeDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| mName | string | A unique string identifier for the maze. |
| mDims | Vector2 | A 2D vector that indicates the dimensions of the maze in terms of width and height.  \*Note: Use of mDims implies the generation of a rectangular maze. |
| mCustomDims | List<int> | A list of integers that describes the number of elements in each row. This format allows for varying number of elements in each row of the maze. |
| mStart | string | A string identifier indicating the starting position within the maze. \*Note: e.g., A1 represents the bottom left of the maze. |
| mFinish | string | A string identifier indicating the finishing or end position within the maze. \*Note: e.g., A1 represents the bottom left of the maze. |
| mNumSquares | int | The total number of squares along the maze path. |
| mNumTurns | int | The total number of turns along the maze path. |
| mPath | List<string> | The designated hidden tile path. \*Note: If mPath is not included in the MazeDef, Free Play will be activated where the player can select any path from start to finish. |
| mString | string | The JSON string representation of the maze that is loaded. |

## Touch-Hold-Release

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| ShowNegFB | bool | A boolean that determines the presentation of negative visual feedback when the subject holds too long, holds too short, or moves their touch beyond the bounds of the object. |
| PerfWindowEndTrials | int | The number of most recent trials evaluated against the block end threshold. |
| PerfThresholdEndTrials | float | The necessary performance percentage needed to end the block. |
| AvoidObjectDuration | float | The duration of the AvoidObject State. The units are in seconds. |
| SelectObjectDuration | float | The duration of the SelectObject state. The units are in seconds. |
| TimeoutDuration | float | The duration of the timeout for clicking incorrectly. The units are in seconds. |
| ItiDuration | float | The duration of the ITI state. The units are in seconds. |
| MinTouchDuration | float | The minimum touch duration required to be considered a successful object selection. |
| MaxTouchDuration | float | The maximum touch duration that is the threshold of a successful object selection. |
| ObjectSize | float | The size of the object. |
| ObjectSizeMin | float | The minimum size of the object. |
| ObjectSizeMax | float | The maximum size of the object. |
| PositionX | int | The x position of the object. |
| PositionX\_Min | int | The minimum x position of the object. |
| PositionX\_Max | int | The maximum x position of the object. |
| PositionY | int | The y position of the object. |
| PositionY\_Min | int | The minimum y position of the object. |
| PositionY\_Max | int | The maximum y position of the object. |
| RewardTouch | bool | A boolean that determines if reward is sent for touching an object for the appropriate duration.  \*Note: Ensure only one of RewardTouch or RewardRelease are set to true. |
| RewardRelease | bool | A boolean that determines if reward is sent upon selecting the blue object for the appropriate duration.  \*Note: Ensure only one of RewardTouch or RewardRelease are set to true. |
| NumTouchPulses | int | The number of pulses transmitted to the SyncBox when a pulse reward is given for object touch. |
| NumReleasePulses | int | The number of pulses transmitted to the SyncBox when a pulse reward is given for object release. |
| RandomObjectSize | bool | A boolean that randomizes the object’s size. |
| RandomObjectPosition | bool | A boolean that randomizes the object’s position. |
| TimeToAutoEndTrialSec | int | The number of seconds before the trial automatically terminates.. |
| TouchToRewardDelay | float | The duration between the initiation of a touch and the sending of reward. |
| ReleaseToRewardDelay | float | The duration between the release of a touch and the sending of reward. |

### TrialDef

See **BlockDef**.

### TaskDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| StartWithSelectObjectState | bool | A boolean for whether the trial skips the AvoidObject State and begins with the SelectObject state. |

## Visual Search

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| TrialStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, following the order in the TrialStimIndices. |
| TrialStimIndices | int[] | An array of integers that indicate the stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| TrialStimTokenReward | int[] | An array of integers that indicate number of tokens that are added or subtracted from the token bar if that stimulus is selected |
| ProbabilisticTrialStimTokenReward | Reward[][] | An array indicating the likelihood of obtaining varying reward amounts for each stimulus, assigned based on the order in the TrialStimIndices. |
| ProbabilisticNumPulses | Reward[] | An array indicating the likelihood of receiving varied reward amounts when a reward pulse is issued. |
| RandomizedLocations | bool | A boolean indicating if the positions in TrialStimLocations are randomly allocated or are used as listed. |
| TokensWithStimOn | bool | A boolean that determines if the stimuli and associated feedback remain visible on screen while token rewards appear.  \*Default: false. |

### TrialDef

See **BlockDef**.

### TaskDef

No additional variables used.

### StimDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| IsTarget | bool | A boolean that determines if the stimulus is the target object, based on a positive reward assignment. |

## What-When-Where

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| CorrectObjectTouchOrder | int[] | Assigns the order of selection for the block sequence, indexing through the stimuli defined in SearchStimIndices. |
| SearchStimIndices | int[] | An array of integers that indicate the search stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| SearchStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, assigning according to the order in the SearchStimsIndices. |
| DistractorStimIndices | int[] | An array of integers that indicate the distractor stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| DistractorStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, assigning according to the order in the DistractorStimsIndices. |
| RandomizedLocations | bool | If set to true, the SearchStimLocations and DistractorStimLocations are pooled and assigned. |
| LeaveFeedbackOn | bool | A boolean that indicates whether halo feedback remains visible after selecting an object. |
| SliderGain | int[] | An integer array that assigns the slider value gain, given the index in the in the sequence. |
| SliderLoss | int[] | An integer array that assigns the slider value loss, given the index in the in the sequence. |
| ErrorThreshold | int | An integer that indicates the number of consecutive errors the participant can make before imposing a delay of the start button presentation. |
| MaxTrialErrors | int | If the BlockEndType is CurrentTrialErrorCount or CurrentTrialPercentError, the player is limited to MaxTrialErrors in a single trial before the trial will terminate. |
| GuidedSequenceLearning | bool | A boolean that guides sequence learning by flashing a yellow halo around the next correct stimulus in the sequence. |

\*Note: If the BlockEndType is CurrentTrialErrorCount or CurrentTrialPercentError, the player can make several errors in a single trial and the block will terminate once they have completed the minimum number of trials in the block and the error count or percent error is below or equal to the BlockEndThreshold on the most recent trial.

### TrialDef

See **BlockDef.**

### TaskDef

No additional variables used.

### StimDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| IsCurrentTarget | bool | A boolean that indicates whether the stimulus is the target. |
| IsDistractor | bool | A boolean that indicates whether the stimulus is a distractor. |

## Working Memory

### BlockDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| SampleStimLocation | Vector3 | A 3D vector specifying the location of the sample stim at the start of the trial. |
| SearchStimIndices | int[] | An array of integers that indicate the search stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| SearchStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, assigning according to the order in the SearchStimIndices. |
| PostSampleDistractorStimIndices | int[] | An array of integers that indicate the post-sample distractor stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| PostSampleDistractorStimLocations | Vector3[] | An array of 3D vectors specifying where each stimulus will appear, following the order in the PostSampleDistractorStimIndices. |
| TrialStimIndices | int[] | An array of integers that indicate the stimuli loaded for the block, using the index that corresponds to the StimIndex of the StimDef config file. |
| SearchStimTokenReward | int[] | An array of integers that indicate number of tokens that are added or subtracted from the token bar if that stimulus is selected.  Note: Target is assigned by having a reward greater than 0, only one stimulus in the trial can have a positive token reward. |
| ProbabilisticSearchStimTokenReward | Reward[][] | An array indicating the likelihood of obtaining varying reward amounts for each stimulus, assigned based on the order in the TrialStimIndices. |

### TrialDef

See **BlockDef**.

### TaskDef

No additional variables used.

### StimDef

|  |  |  |
| --- | --- | --- |
| *Variable Name* | *Type* | *Description* |
| IsTarget | bool | A boolean that determines if the stimulus is the target object, based on a positive reward assignment. |