Mission Flow Node Scripting Reference

/t(W.I.P.)

A Mission Flow Node is a scripting unit used in Mechwarrior 5: Mercenaries, used by the ***Objective Chain Mission Director*** (if you are using your own custom mission director - use mission components directly without using the Mission Flow Node interface) in a ***scenario file***. The Objective Chain Mission Director converts Mission Flow Nodes from the scenario file into Mission Components at run-time.

Any event or objective in the game is specified through a Mission Flow Node.

This is a comprehensive list of all possible mission flow nodes, and what values they require.

Game Control / Scripted Events

- Container
- Timer
- Dispatch Waves To Attack Garrison
- Dispatch Waves To Attack Player Lance
- Dispatch Waves To Go To Target Location
- End Mission - Timed
- Ambush Limiter
- Artillery Control
- Play Audio
- Play Climactic Music
- Treasure/Loot Control
- Capture Garrison

Building-Related/

- Destroy Buildings
- Destroy All Buildings
- Defend Buildings
- Target Buildings In An Area
- Scan
- Destroy Garrison
- Defend Garrison
- Turn On Garrison Speech

Units-Related

- Destroy Units
- Destroy Units Of Type
- Destroy Units In An Area
- Target Units

Movement-Related

- Go To
- Evac - Go To Safety Zone
- Evac - Go To Extraction Point
- Go To Garrison
- Go To Units

One-offs

- Repair

Deprecated Nodes

- <DEPR> Dispatch: Waves - Attack Garrison
- <DEPR> Dispatch: Waves - Attack Player
- <DEPR> Dispatch: Waves - Go To New Base (Home Location)
- <DEPR> Dispatch: Reinforce 1st Dispatch Target
- <DEPR> Dispatch: Alert 1st Dispatch Target

Common Values For All Mission Flow Nodes

1. **Name:** The unique name of this mission flow node, used to identify this node. This is used by Mission Flow Connectivity Data as well as other mission flow nodes - so ***make sure to make this unique.***
2. **Start On Setup?:** Whether this Mission Flow Node starts on setup (i.e., starts at the beginning of the mission and not triggered by another flow node.)
3. **Location:** Location of this objective (this may or may not be applicable to each mission flow node type - check below). The location can be a locator specified in the Area Tile List, or custom encounter location specified in the "FindByActorTag" field.
4. **Node Type:** Whether this is an objective (what kind) or not. For example - scripted events will probably be non-objectives (eg. waves of enemies appearing, particular enemy flees to alert a garrison etc.), while other actions are objectives (eg. "destroy target building" will probably be an objective. Case in point - it might not be an objective - eg. if you want something to happen when a certain building is destroyed.)
5. **Parent Mission Flow Node:** If this is part of a larger objective, the parent name (name of its container) goes here. Leave it as None for default no parent.
6. **Definition Tags:** Defining the mission flow node and its details (Further info about definition tag under each kind of mission flow node),
7. **Timer Value:** After this amount of seconds from starting, this mission flow node will return failure - use for timed objectives. (0.0 is default and will be ignored.)
8. **Additional Mission Data:** This is different for different types of flow nodes. Details under each flow node type. **There are a few common parameters (see below), can be found in the list.**
9. **Reward/Briefing:** How much reward is awarded for success of this mission flow node.
10. **Description MainText:** Objective text (shows up in the shortened version of the HUD).
11. **Description SubText:** Objective Text in Details (shows up in the longer version of the HUD).
12. **Dispatch Data:** This is only applicable to Dispatch Mission Flow Nodes.

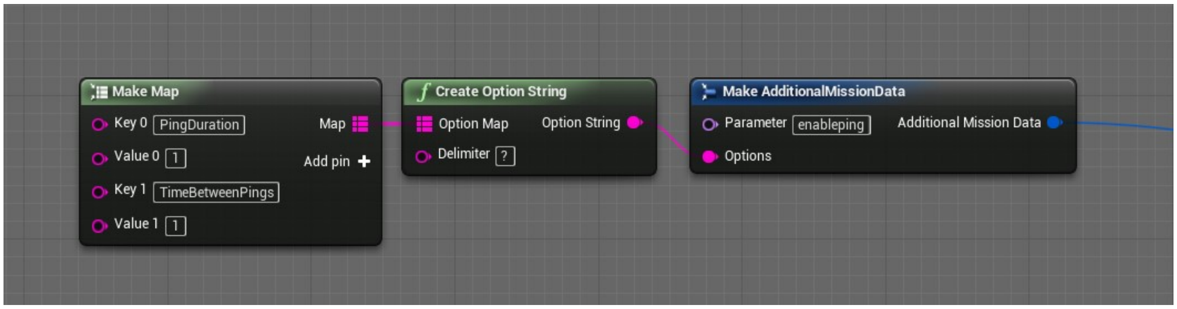
Y/N marks Yes/No for each of these in the sections below - as in, whether they're needed or not for that mission flow node.

Common Additional Mission Data

- "failontimer" - The mission component will fail after this time (if it's not already succeeded/aborted/failed).
- "succeedontimer" - The mission component will succeed after this time (if it's not already succeeded/aborted/failed).
- "abortontimer" - The mission component will abort after this time (if it's not already succeeded/aborted/failed).
- "highlighttext" - The text to show on as the highlight text (if applicable). *Highlight Text is the text shown on bars/additional text on the top part of the HUD etc.*

Creating an Option String

Use MakeMap to create Option/Value pairs and feed to CreateOptionString with a '?' delimiter character.



Game Control / Scripted Events

Container

A container is used to contain multiple objectives inside it as part of a bigger objective. It only returns success if ALL of its children mission flow nodes return success, and fail if ANY of its underlying mission flow nodes fail.

1. **Name:** Y
2. **Start On Setup?:** Y

3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** N
6. **Definition Tags:** *Objectives.Type.Container*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "shownumberofsubobjectives", "showNumberOfSubObjectives", "ShowNumberOfSubObjectives" (Shows x/x number of subobjectives)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Timer

A timer returns success after a specified amount of time.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Timer*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "Timer", "timer", "Timer Value", "timer value", "Timer value"
       Value: Amount of seconds after which this flow node will return successful
    2. *Name*: "ShowBar" - display the timer progress bar
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

End Mission - Timed

End mission after a specific time. (Default is success.) *If you have an audio playing Success Audio, the mission component will play the audio before ending the game.*

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.EndMission.Timed*
7. **Timer Value:** Y (DO NOT USE this as the timer value! This value only fails the mission component after that amount of time.)
8. **Additional Mission Data:** **(Optional)**
    1. *Name:* "Timer", "timer", "Timer Value", "timer value", "Timer value"
       Value: Amount of seconds from when this mission component begins - after which the game will end.
    2. *Name:* "failure", "Failure" - Fails the mission.
    3. *Name:* "fadetime", "fadeTime", "FadeTime"
       Value: Fade out time.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Dispatch Waves To Destroy Garrison

Sends waves of units to destroy a garrison. The difference with the above is: if a hotdrop locator is specified as a dispatch locator in this one, this will create units dynamically around it.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y/N (See Additional Mission Data)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch.WavesToDestroyGarrison*
7. **Timer Value:** Y

8. **Additional Mission Data:**
    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional - use only one)
        1. "waitfordeathY", "waitforpreviouswavedeathY" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died. Y = percentage of units dead for this to trigger (100% by default). **Eg.** If you want units to spawn after 75% of the previous wave has died, the text is "waitfordeath75".
        2. "waitfordeath", "waitforpreviouswavedeath" specifies this wave will spawn after *(value)* time after the previous wave has completely been destroyed.
        3. "spawnwhenremainingY" specifies whether this wave should spawn after *(value)* time only when Y units are remaining from previous waves. **Eg.** If you want units to spawn after if 3 units from all previous waves are remaining, the text is "spawnwhenremaining3".
        4. "spawnafterdeathsY" specifies whether this wave should spawn after *(value)* time only when Y units have been destroyed in previous waves. **Eg.** If you want units to spawn after 3 units from past waves have been destroyed, the text is "spawnafterdeaths3".

       Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).
    2. *Name:* "spawnnearlocation", "spawnNearLocation", "SpawnNearLocation" (Will spawn the wave near a specified Location in #3. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
       Value: The max pathing distance (in meters) it will check around the location for hotdrop . If it can't find any, it'll search the whole map for any hotdrop encounters to use.
       Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    3. *Name:* "spawnnearlance", "spawnNearLance", "SpawnNearLance" (Will spawn the wave near player lance. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
       Value: The max pathing distance (in meters) it will check around the lance for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
       Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    4. *Name:* "minSpawnDistance", "MinSpawnDistance", "minspawndistance" (Minimum distance from host player for choosing a drop location. Drops within this distance will be avoided for spawning. Use ONLY with "spawnnearlance"/"spawnnearlocation").
       Value: The minimum spawn distance (in meters).
    5. *Name:* "chancetoignoremindistance", "chanceToIgnoreMinDistance", "ChanceToIgnoreMinDistance" (Chance per wave to ignore minimum distance for hotdrop spawn).
       Value: Chance (0 - 1)
    6. *Name:* "teamid", "teamID", "TeamID" (Will spawn units with this team ID. By default, it spawns hostile team IDs. If used in conjunction with "spawnnearlance" or "spawnnearlocation", it will only search for reinforcement encounters with this team ID - unless overridden in *Options*.)
       Value: Team ID.
       Options: (Optional) "ignoretileteamid" (Will ignore the tile's team ID, find all reinforcement encounters based on aforementioned rules, and set their team IDs to *value* when spawning.) NOTE: This part only works if used in conjunction with "spawnnearlocation" or "spawnnearlance".
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* N
       Dispatch Targets: List of target encounters (locators) to send units in waves to destroy.
       Support Sources: List of hotdrop locators where this dispatch behavior will get its units from. It will create them on the fly.
       Dispatch Type: N

Dispatch Waves To Attack Player Lance

Sends waves of units to attack player lance. ***Try to avoid using this except for special cases because it makes enemy units aware of the entire lance even if they are halfway across the map.*** The difference with the above is: if a hotdrop locator is specified as a dispatch locator in this one, this will create units dynamically around it.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y/N (Read Additional Mission Data)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch.WavesToAttackPlayerLance*
7. **Timer Value:** Y
8. **Additional Mission Data:**

    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional - use only one)
        1. "waitfordeathY", "waitforpreviouswavedeathY" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died. Y = percentage of units dead for this to trigger (100% by default). **Eg.** If you want units to spawn after 75% of the previous wave has died, the text is "waitfordeath75".
        2. "waitfordeath", "waitforpreviouswavedeath" specifies this wave will spawn after *(value)* time after the previous wave has completely been destroyed.
        3. "spawnwhenremainingY" specifies whether this wave should spawn after *(value)* time only when Y units are remaining from previous waves. **Eg.** If you want units to spawn after if 3 units from all previous waves are remaining, the text is "spawnwhenremaining3".
        4. "spawnafterdeathsY" specifies whether this wave should spawn after *(value)* time only when Y units have been destroyed in previous waves. **Eg.** If you want units to spawn after 3 units from past waves have been destroyed, the text is "spawnafterdeaths3".

       Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).
    2. *Name:* "spawnnearlocation", "spawnNearLocation", "SpawnNearLocation" (Will spawn the wave near a specified Location in #3. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
       Value: The max pathing distance (in meters) it will check around the location for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
       Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    3. *Name:* "spawnnearlance", "spawnNearLance", "SpawnNearLance" (Will spawn the wave near player lance. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
       Value: The max pathing distance (in meters) it will check around the lance for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
       Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    4. *Name:* "minSpawnDistance", "MinSpawnDistance", "minspawndistance" (Minimum distance from host player for choosing a drop location. Drops within this distance will be avoided for spawning. Use ONLY with "spawnnearlance"/"spawnnearlocation").
       Value: The minimum spawn distance (in meters).
    5. *Name:* "chancetoignoremindistance", "chanceToIgnoreMinDistance", "ChanceToIgnoreMinDistance" (Chance per wave to ignore minimum distance for hotdrop spawn).
       Value: Chance (0 - 1)
    6. *Name:* "teamid", "teamID", "TeamID" (Will spawn units with this team ID. By default, it spawns hostile team IDs. If used in conjunction with "spawnnearlance" or "spawnnearlocation", it will only search for reinforcement encounters with this team ID - unless overridden in *Options*.)
       Value: Team ID.
       Options: (Optional) "ignoretileteamid" (Will ignore the tile's team ID, find all reinforcement encounters based on aforementioned rules, and set their team IDs to *value* when spawning.) NOTE: This part only works if used in conjunction with "spawnnearlocation" or "spawnnearlance".

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* N
       Dispatch Targets: N
       Support Sources: List of hotdrop locators where this dispatch behavior will get its units from. It will create them on the fly.
       Dispatch Type: N

Dispatch Waves To Go To Target Location

Sends waves of units to go to a location. The difference with the above is: if a hotdrop locator is specified as a dispatch locator in this one, this will create units dynamically around it.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y/N (Read Additional Mission Data)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch.WavesToGoTargetLocation*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional - use only one)
        1. "waitfordeathY", "waitforpreviouswavedeathY" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died. Y = percentage of units dead for this to trigger (100% by default). **Eg.** If you want units to spawn after 75% of the previous wave has died, the text is "waitfordeath75".
        2. "waitfordeath", "waitforpreviouswavedeath" specifies this wave will spawn after *(value)* time after the previous wave has completely been destroyed.

        3. "spawnwhenremainingY" specifies whether this wave should spawn after *(value)* time only when Y units are remaining from previous waves. **Eg.** If you want units to spawn after if 3 units from all previous waves are remaining, the text is "spawnwhenremaining3".
        4. "spawnafterdeathsY" specifies whether this wave should spawn after *(value)* time only when Y units have been destroyed in previous waves. **Eg.** If you want units to spawn after 3 units from past waves have been destroyed, the text is "spawnafterdeaths3".

    Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).

    2. *Name:* "spawnnearlocation", "spawnNearLocation", "SpawnNearLocation" (Will spawn the wave near a specified Location in #3. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
    Value: The max pathing distance (in meters) it will check around the location for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
    Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    3. *Name:* "spawnnearlance", "spawnNearLance", "SpawnNearLance" (Will spawn the wave near player lance. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
    Value: The max pathing distance (in meters) it will check around the lance for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
    Options: (Optional) "limitX" where X = the number of reinforcement encounters to be selected (default 3), sorted on pathing distance (closest first).
    4. *Name:* "minSpawnDistance", "MinSpawnDistance", "minspawndistance" (Minimum distance from host player for choosing a drop location. Drops within this distance will be avoided for spawning. Use ONLY with "spawnnearlance"/"spawnnearlocation").
    Value: The minimum spawn distance (in meters).
    5. *Name:* "chancetoignoremindistance", "chanceToIgnoreMinDistance", "ChanceToIgnoreMinDistance" (Chance per wave to ignore minimum distance for hotdrop spawn).
    Value: Chance (0 - 1)
    6. *Name:* "teamid", "teamID", "TeamID" (Will spawn units with this team ID. By default, it spawns hostile team IDs. If used in conjunction with "spawnnearlance" or "spawnnearlocation", it will only search for reinforcement encounters with this team ID - unless overridden in *Options*.)
    Value: Team ID.
    Options: (Optional) "ignoretileteamid" (Will ignore the tile's team ID, find all reinforcement encounters based on aforementioned rules, and set their team IDs to *value* when spawning.) NOTE: This part only works if used in conjunction with "spawnnearlocation" or "spawnnearlance".

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* N
    Dispatch Targets: The target locator. (If more than one locator is listed, only the first one will be considered.)
    Support Sources: List of hotdrop locators where this dispatch behavior will get its units from. It will create them on the fly.
    Dispatch Type: N

Ambush Limiter

Limits the number of ambushes in the mission (if there are any ambush encounters).

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.LimitAmbushes*
7. **Timer Value:** N
8. **Additional Mission Data:**
    1. *Name:* "number", "Number"
    Value: Number of ambushes.
    2. Name: "DelayStartDialogue"
    Start Dialogue will be delayed until the the first patrol/ambush is triggered
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Artillery Control

Limits the number of ambushes in the mission (if there are any ambush encounters).

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (if uselocatorastarget is used)
4. **Node Type:** Y

5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Special.ControlArtillery*
7. **Timer Value:** N
8. **Additional Mission Data:**
    1. *Name:* "useplayerastarget", "usePlayerAsTarget", "UsePlayerAsTarget" uses the player as target around which artillery drop points are formed
    2. *Name:* "uselocatorastarget", "useLocatorAsTarget", "UseLocatorAsTarget"
       Value: The outer radius (in meters) from the locator (specified in location) - this will determine spread of bombings. The inner radius is 10m, and default outer radius is 150m.
    3. *Name:* "delaybetweenfiring", "delayBetweenFiring", "DelayBetweenFiring"
       Value: Delay between two artillery fires
    4. *Name:* "delaybetweengroupfiring", "delayBetweenGroupFiring", "DelayBetweenGroupFiring"
       Value: Delay between two sets of artillery fires (one set can consist of many artilleries)
    5. *Name:* "ignoredispatchtargets", "ignoreDispatchTargets", "IgnoreDispatchTargets" - Artillery won't fire on garrisons pointed in Dispatch Targets in Dispatch Data
    6. *Name:* "usesupportsourcesonly" - Only artilleries in the garrisons specified in "Support Sources" in Dispatch Data will be used.
    7. *Name:* "pingartilleryleader" - Pings the artillery leader when a group fires.
       Value: Length of time the ping lasts (default 3 seconds)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data: Y** (If "ignoredispatchtargets" or "usesupportsourcesonly" is used)

Play Start Audio

Play Start Audio. This returns successful when the audio has completed (Normally the Start Audio would normally play - but this is for cases when you want to listen to an audio ending before doing something else).

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.PlayStartDialogueAndComplete*
7. **Timer Value:** N
8. **Additional Mission Data:** N
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N
13. **Dialogue Script:** Y
14. **Override Icon:** N

Play Climactic Music

Play climactic music associated with current music set.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.PlayClimacticMusic*
7. **Timer Value:** N
8. **Additional Mission Data:** N
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N
13. **Dialogue Script:** Y
14. **Override Icon:** N

Treasure/Loot Control

Places random loot crates in the mission

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y

5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Special.TreasureControl*
7. **Timer Value:** N
8. **Additional Mission Data:**
    1. *Name:* "lootamount" - Amount of loot that will be spawned.
    2. *Name:* "ignoredispatchtargets" - Loot won't be spawned in garrisons pointed in Dispatch Targets in Dispatch Data
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data: Y** (If "ignoredispatchtargets" or "usesupportsourcesonly" is used)

Capture Garrison

Capture a garrison by standing in the marked area and keeping all enemies out of the area

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter this building is in)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Capture.Garrison*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "CaptureDuration"
       Value - Time in seconds (between 1 and 600) required to stay in region
    2. Name: "HighlightText" - *Highlight Text is the text shown on bars/additional text on the top part of the HUD etc*
       OptionString: The text to show on as the highlight text (if applicable).
    3. *Name:* "RemoveBar" *If present, do not use the capture bar indicator from the HUD*
    4. *Name: "NoGroundMarker" If present, do no create the on-ground marker indicating the capture zone*
    5. *Name: "EnablePing " Turn on the garrison's radar*
       OptionString - "PingDuration=seconds?TimeBetweenPings=seconds"
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Building-Related Objectives

Destroy Buildings

Destroy a building trigger - this returns successful when specified building(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter this building is in)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.Building*
    1. Add any additional building filtering tag here of the *Object.Building.<Details>* type. Eg. If you want the encounter to choose all guard towers as targets, put *Object.Building.Military.GuardTower* here. The encounter will only choose from its target buildings list that are marked with *Object.Interactions.Destructible.* For specific missions, if you don't want an encounter to choose a certain building, do not add it under target buildings in the encounter (or do not have the *Object.Interactions.Destructible* tag on it.)
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "amount", "Amount"
       Value - Amount of buildings to consider from the filtered buildings (chosen at random). Eg. If the encounter returns 4 guard towers and the amount is set to 1 - it will choose one of those towers at random as the target.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Destroy All Buildings

Destroy buildings irrespective of whichever garrison they're in. The buildings **MUST** be in the list of important buildings under encounters.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.OverallBuildings*
    1. Add any additional building filtering tag here of the *Object.Building.<Details>* type. Eg. If you want the encounter to choose all guard towers as targets, put *Object.Building.Military.GuardTower* here. The encounter will only choose from its target buildings list that are marked with *Object.Interactions.Destructible.* For specific missions, if you don't want an encounter to choose a certain building, do not add it under target buildings in the encounter (or do not have the *Object.Interactions.Destructible* tag on it.)
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "amount", "Amount"
       Value - The number of buildings to destroy (exclusive, doesn't work with "percentage").
    2. *Name:* "percentage", "Percentage"
       Value - The percentage of buildings to destroy (exclusive, doesn't work with "amount").
    3. *Name:* "nomarker", "noMarker", "NoMarker" (if this key is found, units aren't individually marked.)
    4. *Name:* "checkfriendlies", "checkFriendlies", "CheckFriendlies"
       (Checks for only friendly buildings. *Considers only hostile buildings by default.*)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Defend Buildings

Defend a building - this returns failed when specified building(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter this building is in)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Defend.Building*
    1. Add any additional building filtering tag here of the *Object.Building.<Details>* type. Eg. If you want the encounter to choose all guard towers as targets, put *Object.Building.Military.GuardTower* here. The encounter will only choose from its target buildings list that are marked with *Object.Interactions.Destructible.* For specific missions, if you don't want an encounter to choose a certain building, do not add it under target buildings in the encounter (or do not have the *Object.Interactions.Destructible* tag on it.)
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "amount", "Amount"
       Value - Amount of buildings to consider from the filtered buildings (chosen at random). Eg. If the encounter returns 4 guard towers and the amount is set to 1 - it will choose one of those towers at random as the target to defend.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Destroy Buildings In An Area

Destroy percentage of all assets in an area (garrison) - this returns success when the specified percentage of building(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.Area*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "percentage", "Percentage"
       Value - Percentage of buildings to destroy in this encounter. (Note that the encounter chooses from ALL its buildings here.)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Scan

Scan building(s) - this returns success when the specified building(s) are scanned.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Scan*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "quantity", "Quantity"
       Value - The number of actors that need to be scanned in order for the component to succeed.
    2. *Name:* "incorrect", "Incorrect"
       Value - The number of "false" scannable actors. These will mark as scannable, but will not contribute to the objective progress.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Destroy Garrison

Destroy percentage of all assets in an area (garrison) - this returns success when the specified percentage of garrison has been destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.Garrison*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "percentage", "Percentage"
       Value - Percentage of buildings to destroy in this garrison. (All destructible buildings in an encounter's boundary is considered.)
    2. *Name:* "removebar", "removeBar", "RemoveBar" (Does not show the healthbar of garrison).
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Defend Garrison

Defend percentage of all assets in a garrison - this returns failure when garrison health goes below specified percentage.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Defend.Garrison*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "percentage", "Percentage"
       Value - Percentage of buildings to defend in this garrison. (All destructible buildings in an encounter's boundary is considered.)
    2. *Name:* "enableping", "enablePing", "EnablePing" (Enable pinging of enemies - their locations will be displayed for a while every couple of seconds while this mission component is active.)
    3. *Name:* "removebar", "removeBar", "RemoveBar" (Does not show the healthbar of garrison).
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Turn On Garrison Speech

Turn on default garrison speech (off by default). Garrison commanders will talk based on whether they're friendly or not.

1. **Name:** Y
2. **Start On Setup?:** Y

3. **Location:** Y (Locator of the encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.TurnOnGarrisonSpeech*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "percentage", "Percentage"
       Value - Percentage of buildings to defend in this garrison. (All destructible buildings in an encounter's boundary is considered.)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Unit-Related Objectives

Destroy Units

Destroy units trigger - this returns successful when specified unit(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.Unit*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "units", "Units"
       Guids - The unit guids to be destroyed for this to trigger success.
    2. *Name:* "nomarker", "noMarker", "NoMarker" (if this key is found, units aren't individually marked.)
    3. *Name:* "percentage", "Percentage"
       Value - Percentage of units to check for (default is 100%).
    4. "markinrange", "markInRange", "MarkInRange"
       Value - Range (in meters) around the units - if the player enters this range only then will the target be marked.
    5. *Name:* "enableping", "enablePing", "EnablePing" (Enable pinging of enemies - their locations will be displayed for a while every couple of seconds while this mission component is active.)
    6. *Name:* "CheckSpawning", "checkspawning" - Check that the units have spawned - if they have not, process them as if they were destroyed
       Value - Time (in seconds) to delay the check after the start of the component. Min - 10 seconds, Max - 60 seconds.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Destroy Units Of Type

Destroy units trigger - this returns successful when specified unit(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.UnitsOfType*
    1. Add any unit filtering tags here of *UnitType.<Type>.* So if we're checking for only tanks, put *UnitType.Vehicle* as a filtering tag.
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "amount", "Amount"
       Value - The number of units to check for.
    2. *Name:* "checkfriendlies", "checkFriendlies", "CheckFriendlies"
       (Checks for only friendly units. *Considers only hostile units by default.*)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Destroy Units In Area/Encounter

Destroy percentage of all units in a garrison/encounter - this returns successful when amount of unit(s) are destroyed.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Location of encounter)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Destroy.UnitsInArea*
    1. Add any additional unit filtering tag here of the *UnitType.<Details>* type. Eg. If you want to destroy only all mechs in the garrison, put *UnitType.Mechs*.
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "percentage", "Percentage"
       Value - Percentage of units to destroy in this encounter.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Target Units

This returns successful when specified unit(s) are targeted by a player.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Target.Unit*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "units", "Units"
       Guids - The unit guids to be destroyed for this to trigger success.
    2. *Name:* "nomarker", "noMarker", "NoMarker" (if this key is found, units aren't individually marked.)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Movement-Related Objectives

Go To

Go to a waypoint. The area trigger is a box of 200 x 200 x 90 m by default, unless otherwise specified to be of different dimensions (or a sphere).

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Travel.GoTo*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "usenpcallies". "useNPCAllies", "UseNPCAllies"
       (Consider NPC Allies too. By default the waypoint only checks for a player-controlled character.)
       (Node will succeed if player/allies are already in the volume)
    2. *Name:* "usesphere", "UseSphere", "useSphere"
       Value - The radius in meters
    3. *Name:* "length", "Length"
       Value - The length of the box in meters.
    4. *Name:* "breadth", "Breadth"
       Value - The breadth of the box in meters.
    5. *Name:* "height", "Height"
       Value - The height of the box in meters.
    6. *Name*: "TriggerOnExit"
       (Node succeeds when player/allies **exit** the volume rather than entering it)
       (If the player/allies are not in the volume when this node triggers, **they will have to enter it first**)

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Evac - Go To Safe Zone

Go to an extraction point and wait for extraction. *If you have an audio playing Success Audio, the mission component will play the audio before ending the game.*

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Evac.GoToSafeZone*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "usenpcallies". "useNPCAllies", "UseNPCAllies"
       (Consider NPC Allies too. By default the waypoint only checks for a player-controlled character.)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Evac - Go To Extraction Point

Go to an extraction point and wait for extraction. *If you have an audio playing Success Audio, the mission component will play the audio before ending the game.*

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Evacuation will occur at Location. If the location is not a hotdrop locator, the results are undefined.)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Evac.GoToExtractionPoint*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "usenpcallies". "useNPCAllies", "UseNPCAllies"
       (Consider NPC Allies too. By default the waypoint only checks for a player-controlled character.)
    2. *Name:* "spawnnearlance", "spawnNearLance", "SpawnNearLance" (Will spawn the wave near player lance. The exact reinforcement locator is chosen at random from a list, avoiding the last used one, following the rules listed below.)
       Value: The max pathing distance (in meters) it will check around the lance for reinforcement encounters. If it can't find any, it'll search the whole map for any reinforcement encounters to use.
    3. With this parameter set #3 Location still must be set.
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Go To Garrison

Go to a marked garrison. This component uses the size of the garrison's inner detection zone as its trigger.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the garrison)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Travel.Garrison*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "usenpcallies". "useNPCAllies", "UseNPCAllies"
       (Consider NPC Allies too. By default the waypoint only checks for a player-controlled character.)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Go To Units

Go to an unit (Activated by any player-controlled lancemate, not AI lancemates).

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Travel.GoToUnit*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "range", "Range"
       Value: The range from the unit in meters when this trigger activates.
    2. *Name:* "units", "Units"
       Guids: The guids of the target units (this will trigger successful if player enters range of ANY of these units).
    3. *Name:* "nomarker", "noMarker", "NoMarker" (No objective marker is displayed on target)
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

One-Offs

Repair

Use a repair bay.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** Y (Locator of the encounter the repair bay is in)
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Repair*
7. **Timer Value:** Y
8. **Additional Mission Data:** N
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:** N

Deprecated Nodes

[DEPRECIATED] Dispatch: Waves - Destroy Garrison

[DEPRECIATED] Dispatch: Waves - Destroy Garrison

Sends waves of units to destroy a garrison.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional) "waitfordeath", "waitforpreviouswavedeath" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died.
       Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* The encounter (locator) that will run this dispatch behavior. If this is left empty, a blank encounter will be created. (Unless you need a very specific encounter to run a dispatch behavior, generally leave this empty.)
       Dispatch Targets: List of target encounters (locators) to send units in waves to destroy.
       Support Sources: List of encounters (locators) where this dispatch behavior will get its units from. You can also spawn a wave without an encounter by referring to a "WaveLocator" - it will use a WaveLocator's aux spawn points to spawn.
       Dispatch Type: Waves - Destroy Garrison

NOTE: To post dialogue at the start of each wave.

1. To the definition tags add:
    1. Objectives.Dispatch.UseDialogueContextBetweenWaves
    2. The desired dialogue context tag, e.g. DialogueContext.Mission.Wave

[DEPRECIATED] Dispatch: Waves - Attack Player

[DEPRECIATED] Dispatch: Waves - Attack Player

Sends waves of units to attack player lance. ***Try to avoid using this except for special cases because it makes enemy units aware of the entire lance even if they are halfway across the map.***

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional) "waitfordeath", "waitforpreviouswavedeath" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died.
       Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* The encounter (locator) that will run this dispatch behavior. If this is left empty, a blank encounter will be created. (Unless you need a very specific encounter to run a dispatch behavior, generally leave this empty.)
       Dispatch Targets: List of target encounters (locators) to send units in waves to destroy.
       Support Sources: List of encounters (locators) where this dispatch behavior will get its units from. You can also spawn a wave without an encounter by referring to a "WaveLocator" - it will use a WaveLocator's aux spawn points to spawn.
       Dispatch Type: Waves - Attack Player

[DEPRECIATED] Dispatch: Waves - Go To New base (Home location)

[DEPRECIATED] Dispatch: Waves - Go To New base (Home location)

Sends waves of units to a location - ***this is the optimal wave dispatch to use*** in most cases.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch*
7. **Timer Value:** Y
8. **Additional Mission Data:**

    1. *Name:* "wavedataX", "waveDataX", "WaveDataX", "wave data X", "Wave Data X" *where X = Wave number* (eg. wavedata1 for wave 1, wavedata2 for wave 2 etc.)
       Value: If nothing is specified under "Options", the time from *previous wave* after which this will happen
       Options: (Optional) "waitfordeath", "waitforpreviouswavedeath" specifies whether this wave should spawn after *(value)* time after every unit in the previous wave has died.
       Guids - The units this wave will use (these units MUST be present in the encounters listed under support sources or they will be ignored).

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* The encounter (locator) that will run this dispatch behavior. If this is left empty, a blank encounter will be created. (Unless you need a very specific encounter to run a dispatch behavior, generally leave this empty.)
       Dispatch Targets: The target locator. (If more than one locator is listed, only the first one will be considered.)
       Support Sources: List of encounters (locators) where this dispatch behavior will get its units from. You can also spawn a wave without an encounter by referring to a "WaveLocator" - it will use a WaveLocator's aux spawn points to spawn.
       Dispatch Type: Waves - Go To New base (Home Location)

[DEPRECIATED] Dispatch: Reinforce 1st Dispatch Target

[DEPRECIATED] Dispatch: Reinforce 1st Dispatch Target

Sends units to reinforce a garrison.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch*
7. **Timer Value:** Y
8. **Additional Mission Data:** N
9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* The encounter (locator) that will run this dispatch behavior. If this is left empty, a blank encounter will be created. (Unless you need a very specific encounter to run a dispatch behavior, generally leave this empty.)
       Dispatch Targets: The target encounter (locator) to reinforce. (If more than one encounter is listed, only the first one will be considered.)
       Support Sources: List of encounters (locators) where this dispatch behavior will get its units from. (Eg. If this refers to a hotdrop locator - it will call reinforcements from that hot drop. If a support source is a normal garrison, it will ask that garrison for detachments it can send as reinforcements.)
       Dispatch Type: Reinforce 1st Dispatch Target

Click here to expand...

[DEPRECIATED] Dispatch: Alert 1st Dispatch Target

Sends units to alert a garrison.

1. **Name:** Y
2. **Start On Setup?:** Y
3. **Location:** N
4. **Node Type:** Y
5. **Parent Mission Flow Node:** Y
6. **Definition Tags:** *Objectives.Type.Dispatch*
7. **Timer Value:** Y
8. **Additional Mission Data:**
    1. *Name:* "units", "Units"
       Guids - The units that will run to alert the dispatch target. It does not matter where these units are. If they are not spawned or dead, this dispatch will do nothing.
    2. *Name: "redcon increase", "REDCON increase", "REDCON Increase"*
       Value - How much REDCON value to increase on the garrison when unit has reached (default 1)
    3. *Name:* "base redcon", "Base REDCON", "Base redcon", "base REDCON"
       Value: REDCON will only increase if it's below this level (default 2)

9. **Reward/Briefing:** Y
10. **Description MainText:** Y
11. **Description SubText:** Y
12. **Dispatch Data:**
    1. *Dispatcher:* The encounter (locator) that will run this dispatch behavior. If this is left empty, a blank encounter will be created. (Unless you need a very specific encounter to run a dispatch behavior, generally leave this empty.)

       Dispatch Targets: The target encounter (locator) to alert by sending an unit. (If more than one encounter is listed, only the first one will be considered.)

       Support Sources: (Irrelevant.)

       Dispatch Type: Alert 1st Dispatch Target