Command List

Commands
- `wb wpdebug` – Toggles waypoint viewing on and off
- `wb wpadd` – Adds a waypoint
- `wb wpdel` – Removes a waypoint
- `wb wppathadd1` – Tags nearest waypoint as start point of a path
- `wb wppathadd2` – Tags nearest waypoint as end point of a path
- `wb wppathdel1` – Selects start point of a path for deletion
- `wb wppathdel2` – Selects end point of a path for deletion
- `wb wpdoor` – Tags nearest waypoint as a door
- `wb wpladder` – Tags nearest waypoint as a ladder
- `wb wpsave` - Saves the waypoint file (do this early, do this often!)

Order of Operations
- Creating a Path
    - Turn on waypoint debugging (at beginning)
    - Create a waypoint (`wb wpadd`) (creates Waypoint 0)
    - Tag waypoint as starting point for path (`wb wppathadd1`)
    - Move to location for next waypoint (must be in Line of Sight of waypoint 0)
    - Create a waypoint (`wb wpadd`) (creates Waypoint 1)
    - Tag waypoint as ending point for path (`wb wppathadd2`)
    - You now have a path from waypoint 0 to waypoint 1
    - Tag waypoint 1 as starting point for path (`wb wppathadd1`)
    - Move to location for next waypoint (stay in LOS of waypoint 1)
    - Create a waypoint (`wb wpadd`) (creates Waypoint 2)
    - Tag waypoint 2 as ending point for path (`wb wppathadd2`)
    - Lather, rinse, repeat

- Removing a Path (from waypoint 0 to waypoint 1)
    - Move a waypoint 0
    - Tag waypoint 0 as starting point of path to be deleted (`wb wppathdel1`)
    - Move to waypoint 1
    - Tag waypoint 1 as ending point of path to be deleted (`wb wppathdel2`)
    - The path is now removed

- Removing a waypoint
    - Move to waypoint to be deleted
    - Delete the waypoint (`wb wpdel`)

Recommended Binds and Aliases

 (These may or may not cause problems, your game to crash, your waypoints to delete, or your head to explode. Use at your own risk)

- Aliases
    - ```
      alias wpstart "wb wpadd;wb wppathadd1;say Path Started"
      ```
    - ```
      alias wpnext "wb wpadd;wb wppathadd2;wb wppathadd1;say Path Continued"
      ```
    - ```
      alias wpdoor "wpnext;wb wpdoor;say Door WP Marked"
      ```
    - ```
      alias wpladder "wpnext;wb wpladder;say Ladder WP Marked"
      ```
    - ```
      alias wpdelete "wb wpdel;say Waypoint Deleted"
      ```
    - ```
      alias wppathdel1 "wb wppathdel1;say Waypoint marked for Path Delete"
      ```
    - ```
      alias wppathdel2 "wb wppathdel2;say Path Removed"
      ```
    - ```
      alias wpsave "wb wpsave;say Waypoints Saved"
      ```

- Binds (change x to whatever you want the key to be)
    - ```
      bind "x" "wpstart"
      ```
    - ```
      bind "x" "wpnext"
      ```
    - ```
      bind "x" "wpdoor"
      ```
    - ```
      bind "x" "wpladder"
      ```
    - ```
      bind "x" "wpdelete"
      ```
    - ```
      bind "x" "wbpathdel1"
      ```
    - ```
      bind "x" "wbpathdel2"
      ```
    - ```
      bind "x" "wbsave"
      ```

Doors

- Sensor Doors
  - Create a waypoint within sensor range of the door (`wb wpadd`) and mark it as a beginning of path (`wb wppathadd1`)
  - Create a waypoint on the other side of the door (`wb wpadd`) in the doors sensor range, and mark it as end of path (`wb wppathadd2`)
- Button-less, Useable Doors (Note that in version .5 this does not work correctly, and should be addressed by version .51)
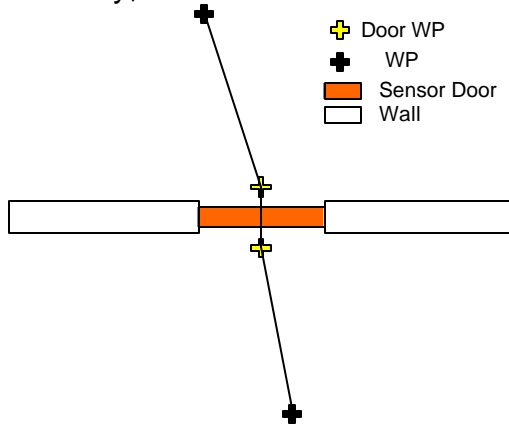
Door WP
WP
Sensor Door
Wall

  - Create a waypoint within 'use' range of the door (`wb wpadd`)
  - Mark the waypoint as a door waypoint (`wb wpdoor`)
  - Mark the waypoint as a beginning of path (`wb wppathadd1`)
  - Open the door, and move to the opposite side of the door
  - Create a waypoint within 'use' range of the door (`wb wpadd`)
  - Mark the waypoint as a door waypoint (`wb wpdoor`)
  - Mark the waypoint as an end of path (wb wppathadd2)
- Button Doors

Door WP
WP
Door Button
Door
Wall

Door WP
WP
Door Button
Door
Wall
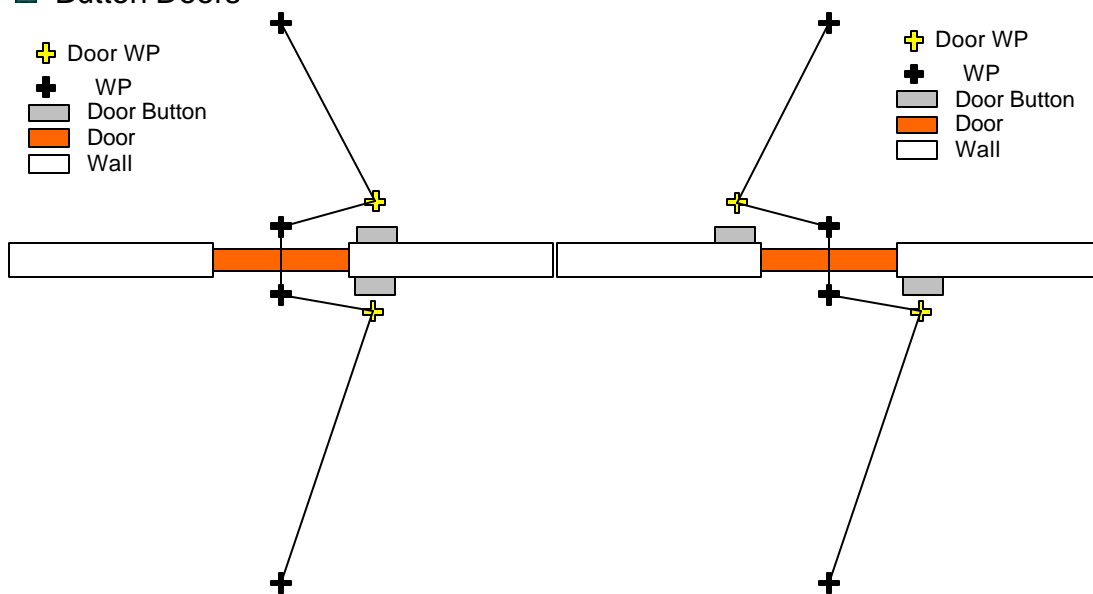
  - Create a waypoint within 'use' range of the door button

- ◼ use `wb wpdoor` to mark the waypoint as a door waypoint
- ◼ Tag the waypoint as the start of a path (`wb wppathadd1`)
- ◼ Move to the near side of the door, and create a waypoint and mark it as the end of the path, then begin a new path (`wb wppathadd2, wb wppathadd1`)
- ◼ Move through the doorway, and create a waypoint directly opposite of the last waypoint and end/begin the path.
- ◼ Move within 'use' range of the other door button and create a waypoint, mark it as a door waypoint, end the path, then continue on.

Cornering

L-Turns:

To make your alien minions move quickly around corners, try to avoid right-angle turns, which require the alien to reach a waypoint, stop, turn, then continue on. Try 'softening' the corners into two 45-degree turns, rather than one 90-degree turn. This will cause them to not slow down as much in the course of the turn. (see diagram below)
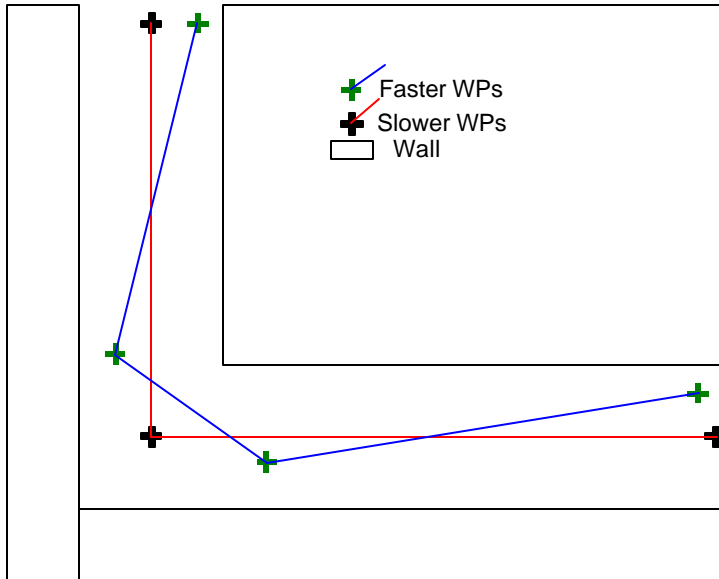
T- Intersections:

To give your alien minions locations to turn around to go back the way they came from (in the event that a hive is under attack, or whatever 'goal' they were headed for has been destroyed) place 3 inter-connected waypoints in T intersections (see diagram) .
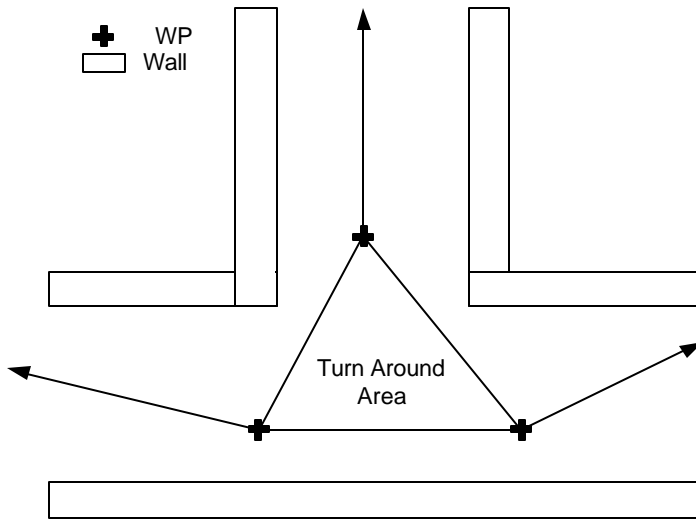

A note of warning: If you have multiple T intersections close to each other, avoid placing triangles in both locations, as the bots could conceivably get caught in a loop, and race in mad figure 8's until a marine appears for them to munch on.

Ramps and sloped surfaces

Non-wall climbing aliens do a check for Walkability on the path between waypoints. Essentially, the Waypoint Manager draws a straight line between two waypoints that are connected by a path. It then moves a certain distance down the line and measures the line's height over ground. If the height is less than the height that the alien can jump, it moves on further, lathering, rinsing, and repeating until it hits the second waypoint. If at some point the height of the line is greater than the height that the alien can jump, the Manager marks that path as non-walkable, and forces the alien to choose a different route. Note that skulks do NOT do walkability checks. Only Gorges, Onos, and Fades do so.

For example, consider the following waypoints:

Not Walkable

X
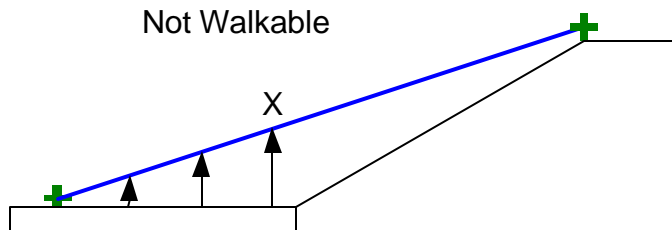
At check number X the Waypoint manager determines that the height of the path is too tall to jump, and nixes the path, even though the alien could continue walking forward, and up the ramp. To make the path walkable, you will need a waypoint at the base of the ramp, like the following waypoints:
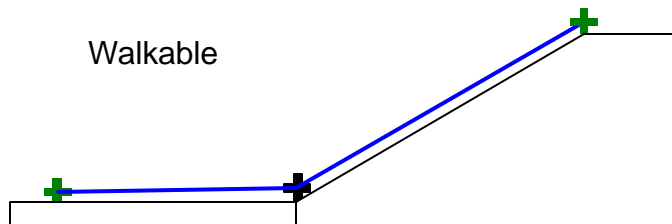
Walkable

If the path between the waypoints is blocked by anything other than a door (no direct line of sight) the path is not walkable for any alien.

To check for walkability, evolve into a gorge with waypoint debugging turned on, This will display only walkable paths.

Ladders

To cause the WhichBots to use ladders, you will need to create a waypoint at the base of the ladder then mark it as a ladder waypoint, mark it as the start of the path, then move to the top of the ladder (in line of sight of the waypoint at the base) and create another waypoint, mark it as a ladder waypoint, and finish the path. Using ladder waypoints causes the Walkability check to succeed, regardless of the height along the waypoint path.