Modules and Other Tools

Contains the docs for official modules or tools that extend BattleArena functionality.

- Built-in Modules
  - Arena Restoration
  - Auto Arena
  - Boundary Enforcer
  - Classes
  - Duels
  - One in the Chamber
  - Scoreboards
  - Team Colors
  - Team Heads
  - Tournaments
  - Vault Integration
- Third Party Modules
- Creating Modules

Built-in Modules

Contains documentation for modules that are built-in the BattleArena plugin.

Arena Restoration

BattleArena comes pre-installed with an arena restoration module. This module lets you restore arenas once a competition has concluded. **This module requires WorldEdit in order to function.**

Configuring the Module

This module can be enabled by adding **arena-restoration** to the **modules** section of your Arena Configuration. You will also want to add the **restore-arena** action when you wish to restore the arena. Here is an example of an arena that restores once the on-complete event has been called:

```
victory:
  allow-join: false
  allow-spectate: false
  next-phase: waiting
  duration: 5s
  events:
    on-complete:
      - leave
      - restore-arena
```

Once added to your config, run **/battlearena reload** to reload the plugin, then for your arena, you can run **/arena schematic <map>** to create a schematic of your map. This will be based on the arena boundaries configured in the Map Creation process. Once the event you have placed the **restore-arena** action is called, the arena will be restored the schematic set.

Auto Arena

BattleArena comes pre-installed with an auto arena module. This module will make it to where when you enter the bounds of an arena, you will automatically be "joined" into the arena. Once you leave the arena boundaries, you will be kicked out of the arena.

Configuring the Module

This module can be enabled by adding **auto-arena** to the **modules** section of your Arena Configuration. No other configuration is required from this point, and once BattleArena has been reloaded with **/battlearena reload**, players will be placed into a competition upon entering arena boundaries configured in the Map Creation process.

Boundary Enforcer

BattleArena comes pre-installed with a boundary enforcer module which ensures that players do not leave the bounds of an arena.

Configuring the Module

This module can be enabled by adding **boundary-enforcer** to the **modules** section of your Arena Configuration. No other configuration is required from this point, and once BattleArena has been reloaded with **/battlearena reload**, players will be unable to leave the arena boundaries configured in the Map Creation process.

Classes

BattleArena comes pre-installed with a classes module that lets you configure classes for an arena.

Configuring the Module

This module can be enabled by adding **classes** to the **modules** section of your Arena Configuration.

Classes Configuration

This module adds a **classes.yml** to BattleArena, which can be found in **plugins/BattleArena/classes.yml**. Here is an example of a class included inside of the classes YML file:

```
classes:
  warrior:
    name: Warrior # The name of the class
    # The items included in the class
    items:
      - iron_sword
      - cooked_beef{amount=16}
      - golden_apple{amount=2}
    # What the player will be given in their armor/offhand slots
    equipment:
      helmet: iron_helmet
      chestplate: iron_chestplate
      leggings: iron_leggings
      boots: iron_boots
      offhand: shield
```

Additionally, if you wish to restrict certain classes by arena, the classes.yml file includes a **require-permission** option (default: true), which means the player must have the "battlearena.classes.equip.<class>" permission.

Arena Configuration

By default, this feature is used in the built-in arena.yml, and is configured to let you "select" a class while in a waiting or countdown phase, with the class being equipped once the game enters the ingame phase. The options used are documented below.

Options

Option	Description	Syntax
class-equipping	Whether class equipping is enabled using the /<arena> equip <class> command.	class-equipping={enabled=<true|false>}
class-equip-only-selects	Whether class equipping only "selects" a class, and must be applied in the arena by using the **equip-class** option.	class-equip-only-selects={enabled=<true|false>}

Actions

- **<, >** denotes a *required* option
- **[,]** denotes an optional option
- Options are separated using the semicolon (**;**)

Equip Class

- Description: Equips a class for a player
- Options
  - **<class>** The class to equip. By default, if a player has selected a class, this will only run if they have not selected one
  - **[clear-inventory]** Whether the player's inventory should be cleared when they select a class (default: true)
  - **[ignore-player-selection]** Whether a player's selection should be overridden (default: false)
- Syntax
  - equip-class{class=**<class>**;clear-inventory=**[clear-inventory]**;ignore-player-selection=**[ignore-player-selection]**}

Duels

BattleArena comes pre-installed with a duels module that lets you duel other players in an arena.

Configuring the Module

This module can be enabled by adding **duels** to the **modules** section of your Arena Configuration. No other configuration is required from this point, and once BattleArena has been reloaded with **/battlearena reload** command, players will be able to use the /<arena> duel <player> command to duel another player. Players must also be granted the battlearena.command.<arena>.duel permission.

One in the Chamber

BattleArena comes pre-installed with a one in the chamber module which adds an ability for players to one-shot others when a bow is used. Additionally, any time a player scores a kill on another player, they are granted an arrow.

An example arena can be found here.

Configuring the Module

This module can be enabled by adding **one-in-the-chamber** to the **modules** section of your Arena Configuration. No other configuration is required from this point, and once BattleArena has been reloaded with **/battlearena reload**, the arena this module is configured on will support one in the chamber.

Scoreboards

BattleArena comes pre-installed with a scoreboards module that lets you configure scoreboards for an arena.

Configuring the Module

This module can be enabled by adding **scoreboards** to the **modules** section of your Arena Configuration.

Scoreboards Configuration

This module adds a **scoreboards.yml** to BattleArena, which can be found in **plugins/BattleArena/scoreboards.yml**. In this file, you can configure scoreboard templates which can be used across any arena. Here is an example of a template used in the waiting phase and in-game phase.

```
templates:
  waiting:
    title: "<gold>Waiting for players..."
    refresh-time: 1s
    lines:
      - simple:
          lines:
            - " "
            - "<yellow>Players: <white>%online_players%/%max_players%"
            - "<yellow>Map: <white>%map%"
            - " "
            - "<white>battleplugins.org"
  ingame-list:
    title: "<gold>%arena%"
    refresh-time: 1s
    lines:
      - simple:
          lines:
            - " "
            - "<yellow>Map: <white>%map%"
```

```
        - "<yellow>Time remaining: <white>%time_remaining_short%"
        - " "
        - "<yellow>Players:"
    - player-list:
        max-entries: 8
    - simple:
        lines:
        - " "
        - "<white>battleplugins.org"
```

Template Options

- **title:** The title of the scoreboard
- **refresh-time:** How quickly the scoreboard contents will be updated
- **lines:** A list of the lines to include.

Line Types

This contains all the line types you can include in the scoreboard.

Simple

- Description: Shows a static list of lines, which will always be visible on the scoreboard.
- Options
  - **lines** (string list): A list of the lines to include

Player List

- Description: A line type which dynamically shows a list of players that are in the game. The line count will update based on how many players are in the game
- Options
  - **max-entries** The maximum number of lines this section will occupy
  - **show-team-color** Whether the player's name should include the team color
  - **require-alive** Whether only alive players should be shown in the scoreboard

Top Stat

- Description: A line type which shows a list of players ordered by which player has the highest of a certain stat (i.e. kills)
- Options
  - **max-entries** The maximum number of lines this section will occupy
  - **stat** The name of the stat to display on the scoreboard
  - **stat-color** The color of the stat
  - **ascending** Whether to display the stat in ascending order
  - **show-team-color** Whether the player's name should include the team color

Top Team Stat

- Description: A line type which shows a list of teams ordered by which team has the highest of a certain stat (i.e. kills)
- Options
  - **max-entries** The maximum number of lines this section will occupy
  - **stat** The name of the stat to display on the scoreboard
  - **stat-color** The color of the stat
  - **ascending** Whether to display the stat in ascending order

Arena Configuration

Scoreboards can be applied using the **apply-scoreboard** action, with the **scoreboard** option set to the scoreboard name. As an example: **apply-scoreboard{scoreboard=waiting}**.

This is also used in the default arena.yml file.

Team Colors

BattleArena comes pre-installed with a team colors module which colors a player's username based on their team color, and gives them a prefix with their team name.

Configuring the Module

This module can be enabled by adding **team-colors** to the **modules** section of your Arena Configuration. No other configuration is required from this point, and once BattleArena has been reloaded with **/battlearena reload**, the module will be activated in the arena.

However, if prefixes are not desired and only coloring the player name based on the team color is desired, the prefixes feature can be disabled by adding **team-prefixes{enabled=false}** to your arena options.

Team Heads

BattleArena comes pre-installed with a team heads module which lets you add the team item configured in **teams.yml** to the player's head.

Configuring the Module

This module can be enabled by adding **team-heads** to the **modules** section of your Arena Configuration. You can then use the **team-heads** action in any event to equip the team item onto the player's head. This option is used by default in ArenaPaintball.

Tournaments

BattleArena comes pre-installed with a tournaments module which lets you run bracket tournaments in BattleArena. This is a slightly different module than the rest in that tournaments are not configured per-arena, and instead all tournament options are in a global **tournament-config.yml**

Configuring the Module

```
# Whether to broadcast when tournaments start to all players on the server.
broadcast-tournament: true

# How long until each round starts in the tournament after the previous
# round has ended. It is recommended to keep this value at 10s or higher
# to allow for players to be removed from the arena and for the arena to
# be reset fully.
advance-time: 10s

# The commands to run when a player wins a tournament. The argument
# %player_name% will be replaced with the player's name. The commands are run
# in the console. These are executed individually for each player, so be careful
# with commands that affect multiple players (like message broadcasts.
commands-on-win:
  - "give %player_name% diamond 16"
```

The default config is included above.

Running a Tournament

A tournament can be started using the **/tournament create <arena>** command. Keep in mind that some arenas may not be capable of running tournaments and the following is required of an arena type to support tournaments:

- Each team must be able to support only 1 player. A 5v5 for instance would not support a tournament.
- A competition must have exactly two teams. A 3v3v3 for instance would not be able to support tournaments.

- Multiple maps. A competition must have multiple maps created in order to support players in a tournament. However, if an arena is dynamic, a single map will suffice as it will scale up.

Once a tournament is created, players can join using the **/tournament join <arena>** command. Once players have joined, a server administrator can start the tournament using **/tournament start <arena>**. If a server administrator desires to end a tournament early, they can run **/tournament end <arena>**.

Vault Integration

BattleArena comes pre-installed with a Vault integration module which hooks into currency plugins and lets you give or take currency or permissions.

Configuring the Module

This module can be enabled by adding **vault-integration** to the **modules** section of your Arena Configuration.

Actions

- **<, >** denotes a *required* option
- **[,]** denotes an optional option
- Options are separated using the semicolon (**;**)

Add Permission

- Description: Adds a permission to a player.
- Options
  - **<permission>** The permission to add to the player.
  - **[transient]** Whether the permission is transient, meaning it will not be permanently added to the player
- Syntax
  - add-permission{permission=**<permission>**;transient=**[transient]**}

Remove Permission

- Description: Removes a permission from a player.
- Options
  - **<permission>** The permission to remove from the player.
  - **[transient]** Whether the permission is transient, meaning it will not be permanently removed from the player
- Syntax
  - remove-permission{permission=**<permission>**;transient=**[transient]**}

Edit Currency

- Description: Deposits (or withdraws) a value from a player's currency.
- Options

    - **<amount>** The amount to add or remove from a player's balance.
    - **[bank]** The bank to update the currency for. If unspecified, the default currency will be used
- Syntax
    - edit-currency{amount=**<amount>**;bank=**[bank]**}

Third Party Modules

A list of third party modules from the community not bundled in BattleArena.

Creating Modules