How to Create a BattleMech

Created by Connor Monahan, last modified by Ryan Burrell on Aug 28, 2019

Process

Creating a new 'Mech for the game is a complicated affair with a lot of hand-offs and potential points of failure (or at least inefficiency and bug generation). Effective implementation of a new 'Mech involves close communication and status tracking across multiple disciplines – likely making use of a combination of JIRA ticketing and custom spreadsheet tracking.

A Note on Variants

Mech Variants generally require a drastically shorter number of steps to create as they are derived from a baseline 'Mech that has had most of its heavy lifting already accomplished. Creating a new variant 'Mech can be done largely without any of the product-level and art-related tasks that a newly added 'Mech would need.

> **NOTE:** New 'Mech variants still need to respect the directory structure of the Content Pack that their prime version is located within.

A Note on Vehicles

The process for creating Vehicles is largely the same as that for creating a 'Mech, though with less game mode-dependent testing and balance considerations.

Concept

Gameplay

All 'Mechs and their variants go through an initial concept phase, followed by paper design if necessary. The concept for a 'Mech should be vetted against several criteria:

- What does this 'Mech bring that's new and interesting to gameplay?
  - Should it have a special feature or "hero" capability? (Example: the Raven's EW suite, the Hatchetman's hatchet, the Cyclops' battle computer, etc.)
- Is the 'Mech and/or its variant appropriate for the current game era? (i.e. if we're set roughly in 3025, did this 'Mech exist then? Was this variant in service yet?)
  - If this is a new creation, what's the pitch for it and how would it fit into the universe? (What's Yang going to tell the player?)
- Do we already have access to art source for this 'Mech - namely, from Piranha Games/MechWarrior Online?
  - What hardpoint configurations can we support, and are they differentiated enough to be interesting?
  - If not, do we have budget to create a new 'Mech?

If this 'Mech is going to need some sort of new, special functionality, now is the time to start the process rolling to prototype and test whatever feature that is. Write up a spec and/or ticket that defines your first pass goal and get it sorted into the current workflow, working with the team as needed.

Important Tasks

Outsourcing Planning
JIRA Ticketing
Added to Production Tracking Spreadsheet(s)
Confluence Documentation - **INITIAL PASS**
Engineering Assessment for Special Functions (if applicable)
UX Assessment for Player Impact

Art Exploration (Optional)

In the case of a brand new 'Mech creation that exists outside the Piranha-fed pipeline, a concept art phase needs to be executed to determine the incarnation of this new 'Mech relative to our existing visual aesthetic. This is especially a

Creation Checklist

Product
- Content Pack Set Up (if applicable)
  - Directory Structure Created
  - AssetTracker(s) Added
- Outsourcing Planning
- JIRA Ticketing
- Added to Production Tracking Spreadsheet(s)

Art / Animation
- Concept Art (if applicable)
- Asset Import (FBX, Textures)
- Animation Import
- Prefab Setup (see Mech and Turret/Vehicle Pipeline/Setup)
  - Ghost Blips on Skeleton
  - Rep Data Complete
  - Colliders Customized
  - Animation Controller
  - Sim Game Prefab
  - Weapon Prefabs
  - Jump Jet Prefabs
- Heraldry
  - Crest Positioned
  - Textures Created & Applied
- Lighting Effects Created & Applied
- HardpointDataDef Created - **FIRST PASS**
- Mech Icon Creation
- Support Weapon Prefabs Position (if applicable)
  - New Support Prefabs Created
- Assets Associated w/ Proper Content Pack / Asset Bundle
  - Files Placed in Proper Content Pack Directory

Engineering Assessment for Special Functions (if applicable)

Design
- Entries Created in Data / Valuation Spreadsheets
- Confluence Documentation - **INITIAL PASS**
- Actor
  - MoveDef Created
  - ChassisDef Authored
    - HardpointDataDef Reviewed & Hardpoints Assigned - **SECOND PASS**
    - LOS Positions Added
    - FixedEquipment Authored & Assigned (if applicable)
  - MechDef Authored
    - Tags Applied
    - Added to Item Collections (Loot) (Shops, Flashpoints, etc.)
    - Spawn Requirements (RequiredToSpawnCompanyTags) (if applicable)
    - MinAppearanceDate Set (if applicable)
  - Variants Created (if applicable)
  - Files Placed in Proper Content Pack Directory & Asset Bundle Assigned
- Playtesting & Iteration

UX / AI
- AI Impact Assessment
- UX Assessment for Player Impact
- AI Tuning (if applicable)
- Tooltips Created (Mech + FixedEquipment)

Audio Events Authored & Inserted (Wwise)

QA / Testing
- Confluence Documentation - **FINALIZED**
  - Test Case(s) Written
- Appears in Combat
  - Weapons & FX Positioned Correctly
  - Special Effects Working (if applicable)
- Editable in MechLab
- Available in Skirmish (if applicable)
- Save / Load Validation

requirement in any instances where we may be fabricating a completely new design that doesn't exist in canon (i.e. the Bull Shark).

The results of this concepting phase will then be used to inform the 3D asset creation should we decide to move forward with developing the 'Mech.

Important Tasks

- Concept Art (if applicable)

Infrastructure

New 'Mechs need a home for their assets and data files in the directory structure. This is largely dependent on what Content Pack the 'Mech is intended for; new DLC releases will need their Content Pack directory structure created.

> ⚠ **NOTE:** It drastically reduces the points of failure if the Content Pack directory structure is created *before* any new 'Mech definitions for its associated release are added. Definitions *can* be moved between content locations after the fact, but this can present tracking and communications issues. Further, all the data files should be found in the same content pack.

Important Tasks

- Content Pack Set Up (if applicable)
  - Directory Structure Created
  - AssetTracker(s) Added

Construction

Modeling & Import

The majority of 'Mech visual designs and assets come via Piranha Games as part of our agreement to use the 'Mech version from their MechWarrior Online (MWO) title. If we're bringing in an existing MWO 'Mech, the modeling has already been done and we only need to perform additional steps in the animation and texturing departments. Portions of this are normally done via outsourcing though the final import of the visual 'Mech entity and all its components is done in-house. For more info, see the Mech and Turret/Vehicle Pipeline/Setup page for further documentation.

Important Tasks

- FBX Authored ('Mech Model + Weapons)
- Prefabs Generated (see Mech and Turret/Vehicle Pipeline/Setup)
- Heraldry & Lighting Effects Created & Applied
- HardpointDataDef Created - **FIRST PASS**
- Mech Icon Creation
- Assets Associated w/ Proper Content Pack
  - Files Placed in Proper Content Pack Directory

Entity Definitions

Like most content in BattleTech, a 'Mech is made manifest in-game via a set of JSON-based data files which serve to link up visual presentation with behavioral info and game logic. 'Mechs are comprised of a number of nested data files, all with a host of parameters and tuning settings. Much of the data within these files is based on the *weight class* of the 'Mech and so can be copy/pasted from existing sources, but just as much of it is unique to the individual 'Mech being added.

As an initial step, a new line item should be created in the 'Mech valuation sheets for the base chassis and then normally-loaded 'Mech. Stats are populated in this new entry (largely derived from tabletop BattleTech values) which will generate final values to be entered into subsequent data files.

If applicable, unique FixedEquipment should be authored at this stage and assigned to the 'Mech for subsequent playtesting.

Once all data defs have been created, the 'Mech can (hopefully) appear in-game for testing via test maps and encounters or using the Skirmish play mode – both artists and designers should perform their own verification.

Important Tasks

- Entries Created in Data / Valuation Spreadsheets
- MoveDef Created
- ChassisDef Authored
  - HardpointDataDef Reviewed & Hardpoints Assigned - **SECOND PASS**
  - LOS Positions Added
  - FixedEquipment Authored & Assigned (if applicable)
- MechDef Authored
  - Tags Applied
  - Files Placed in Proper Content Pack Directory

Playtesting & Polish

Once available in-game with proper presentation the 'Mech should be evaluated in actual gameplay to spot any issues with its construction or behavior, with any issues being ticketed in JIRA for tracking. This stage may see simultaneous work on multiple fronts depending on any issues that arise – special equipment may not be working correctly, art issues may become apparent, etc.

This stage should see the checking and addition of data to the 'Mech's definitions to support inclusion in Campaign / Career play and its economy.

This stage should also see the inclusion of audio event linkage and any related tuning, and functional test cases written.

Important Tasks

- Playtesting & Iteration
- Appears in Combat
  - Weapons & FX Positioned Correctly
  - Special Effects Working (if applicable)
- Support Weapon Prefabs Position (if applicable)
  - New Support Prefabs Created
- Tags Applied (more, if applicable)
- Added to Item Collections (Loot) (Shops, Flashpoints, etc.)
- Spawn Requirements (RequiredToSpawnCompanyTags) (if applicable)
- MinAppearanceDate Set (if applicable)
- Audio Events Authored & Inserted (Wwise)
- Test Case(s) Written

Final QA Pass & Sign-off

Before officially completing the 'Mech for ship, a final QA pass needs to be made to make certain the unit is behaving correctly and accessible in all the places that everyone up to this point *thinks* that it is. Likely at least a portion of this will be handed off to an external QA partner, so updated and finalized documentation and test cases are critical to make sure there aren't wasted cycles on miscommunications.

If we pass this final stage, the 'Mech can be said to be complete and included in the targeted release.

Important Tasks

- Confluence Documentation - **FINALIZED**
- Editable in MechLab
- Available in Skirmish (if applicable)
- Save / Load Validation