

Process Model and Data Dictionary Revision

Lindsey A. Monyelle

DAT 445 Predictive Analytics and Machine Learning (WI)

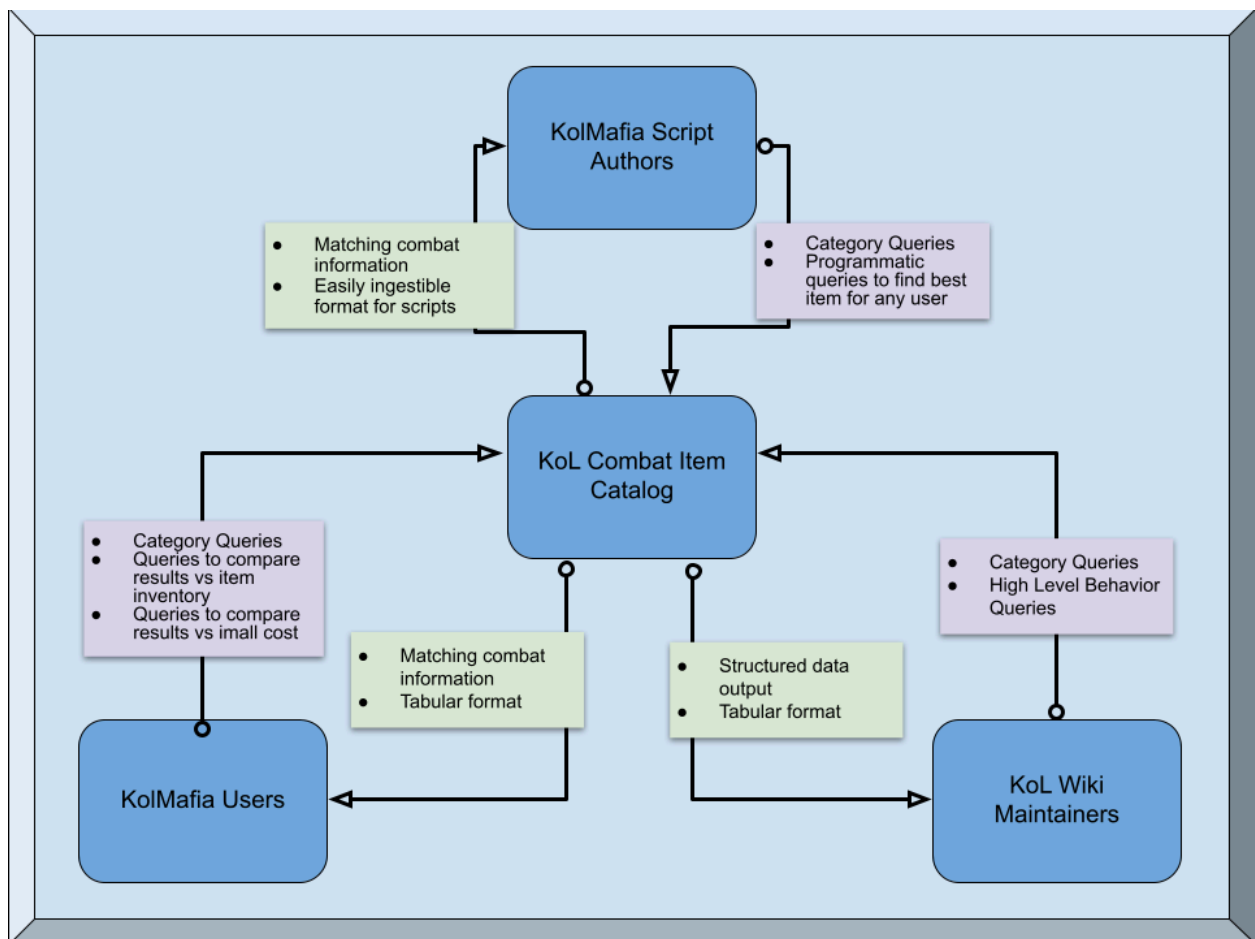
Dr. Jess Schwartz

September 29, 2026

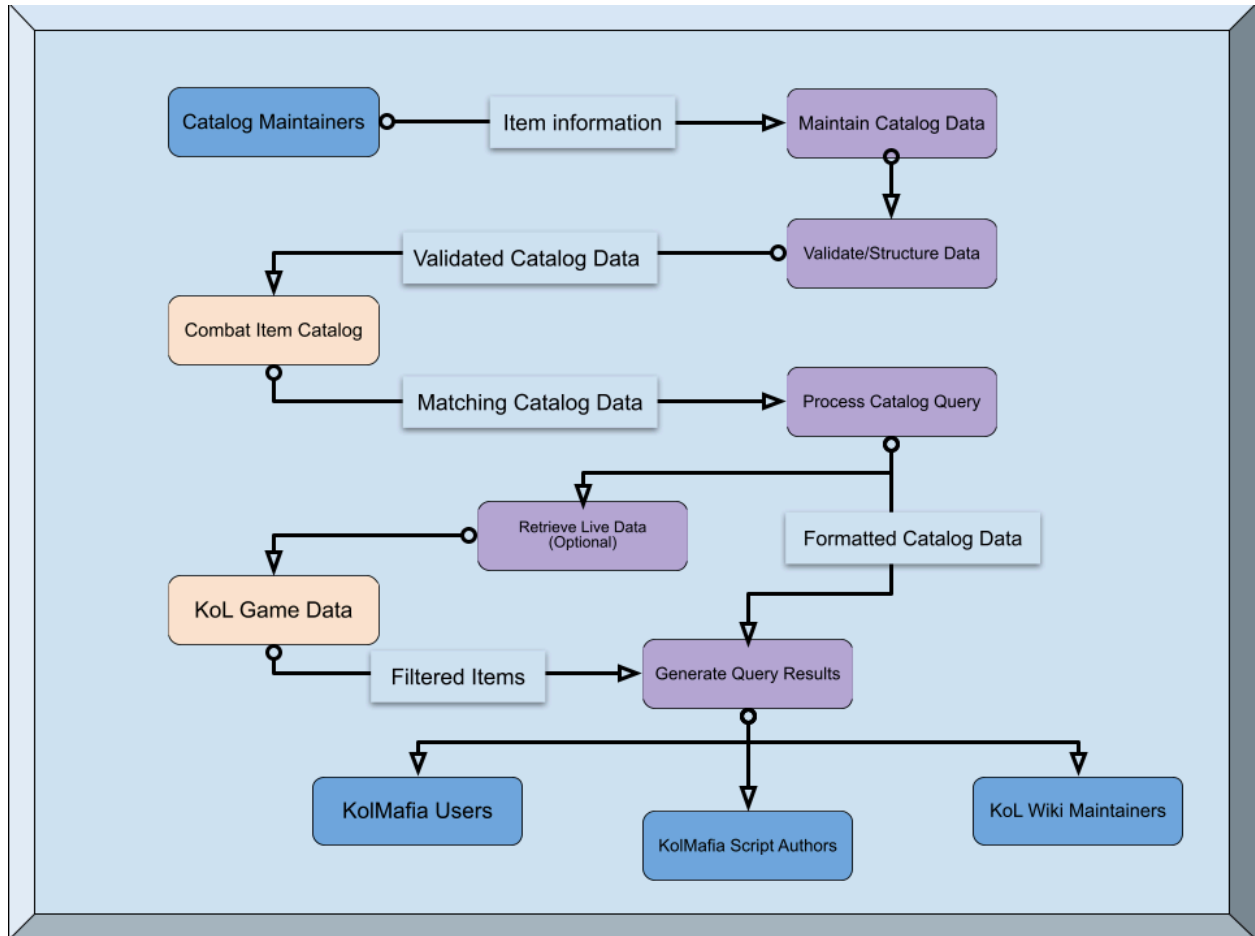
Project Overview

The proposed system is an independent structured catalog for combat items in Kingdom of Loathing (KoL). Currently, information about combat items is spread across wiki pages, item descriptions, player knowledge, and other sources, which makes it difficult for players and script authors to quickly compare items based on their actual combat effects. The proposed system would organize combat items into consistent categories and searchable properties, including elemental damage, healing effects, combat effects, unusual behaviors, inventory quantity, and Mall price information. The primary users would be KoL players who want to make informed decisions about which combat items to use, as well as script authors and community maintainers who need structured and consistent item information.

Context Diagram



Process Model



Revised Data Dictionary

Entity 1: Item

Description: A single combat item. It is the hub of the catalog and holds the item's identity, usage flags, situational notes, etc.

Attribute	Data Type	Key Type	Required ?	Definition / Business Meaning	Example
item_id	INT	Primary Key	Yes	Unique identifier for each item effect	1
kol_item_id	INT	Unique	Yes	KoL's unique item number, used by the provided script to identify and join catalog records with live KoL data	8489
name	Text	None (unique)	Yes	Item name as it appears in the game	New Age hurting crystal

transforms_into_item_id	INT	Foreign Key (→ ITEM)	No	The item you receive after using this one, if any	1
is_tradeable	BOOL	None	Yes	Whether the item can be traded or sold in the Mall. Defaults to yes	1
is_reusable	BOOL	None	Yes	Whether the item survives use instead of being consumed. Defaults to no	1
monster_condition	TXT	None	No	Monster types the item affects specially, such as instant kills	Instantly kills Undead
restrictions	TXT	None	No	Where or when the item can be used	Underwater only
unusual_behavior	TXT	None	No	Odd behaviors such as fumbles, self-damage or random results	Fumbles about 10% of the time

Entity 2: ITEM_EFFECT

Description: One row for each thing an item does in combat: damage, healing, MP or Meat changes, stagger, stun or delevel. An item with several results, such as a heal plus a stun, has several rows.

Attribute	Data Type	Key Type	Required?	Definition / Business Meaning	Example
effect_id	INT	Primary Key	Yes	Unique identifier for each effect row	12
item_id	INT	Foreign Key (→ ITEM)	Yes	The item that produces this effect	3
effect_type	TXT	None	Yes	Identifies a specific combat behavior produced by the item. Each distinct behavior is recorded separately.	stun
target	TXT (list)	None	Yes	Who is affected: monster or self. Defaults to monster	monster
element	TXT (list)	None	No	Damage element (Physical, Hot, Cold, Spooky, Stench, Sleaze, Prismatic, Poison). Only used when effect_type is damage	Hot

amount_type	TXT (list)	None	Yes	How the amount is measured: flat, full, percent_max or percent_missing. Defaults to flat	full
min_value	INT	None	No	Low end of the amount. Empty when unknown or not applicable	80
max_value	INT	None	No	High end of the amount. Equal to the minimum for fixed amounts	120
duration_rounds	INT	None	No	Combat rounds the effect lasts. If it lasts the entire combat, the value will be 99	3
timing	TXT (list)	None	Yes	When the effect acts: immediate, delayed or passive. Defaults to immediate	immediate
notes	TXT	None	No	Extra detail or verification flags	Wiki lists no element; verify

Entity 3: STATUS_EFFECT

Description: A lookup of named effects that an item can grant, such as Sugar Rush or Berry Experiential. These are separate from the immediate combat results in ITEM_EFFECT.

Attribute	Data Type	Key Type	Required?	Definition / Business Meaning	Example
status_effect_id	INT	Primary Key	Yes	Unique identifier for each named effect	1
name	TXT	None (unique)	Yes	The effect's name in the game	Sugar Rush
description	TXT	None	No	What the effect does or where it comes from	Granted by chocolate frosted sugar bomb
effect_kind	TXT (list)	None	Yes	Whether it is a buff, debuff or other. Defaults to other	buff

Entity 4: ITEM_STATUS_EFFECT

Description: A junction table connecting items to status effects. One item can grant several effects, and one effect can come from several items.

Attribute	Data Type	Key Type	Required?	Definition / Business Meaning	Example
item_id	INT	Primary Key, Foreign Key (→ ITEM)	Yes	The item that grants the effect	7
status_effect_id	INT	Primary Key, Foreign Key (→ STATUS_EFFECT)	Yes	The effect that is granted	1
target	TXT (list)	Primary Key	Yes	Who receives the effect: self or monster. Defaults to self	self
duration_turns	INT	None	No	Adventure turns the effect lasts	10
notes	TXT	None	No	Extra detail about how the effect is applied	Applied on use

Revision Memo

After reviewing the Week 5 data dictionary alongside the revised context diagram and process model, I made several changes to better reflect how the proposed KoL Combat Item Catalog will be used. The largest change was removing the CATEGORY entity and the category_id attribute from ITEM. The original design treated each combat item as having a primary category, but the revised design treats each individual combat behavior as equally important. An item may have multiple behaviors, such as dealing damage, healing, stunning an opponent, or applying a status effect, and none of these behaviors needs to be considered the item's primary function. The ITEM_EFFECT entity now represents each distinct behavior as its own record, making the information easier to search and query independently.

The process model also helped clarify the role of the catalog compared with information that changes during gameplay. The revised process includes maintaining catalog data, validating and structuring item information, processing combat behavior queries, retrieving live KoL information, and generating query results. This showed that inventory quantity and Mall price should not be stored in the catalog because they are dynamic

and can change based on the player or current market conditions. Instead, the provided KoLmafia script retrieves this information from KoL when needed and combines it with the stable information stored in the catalog. I therefore added this as a business rule rather than adding inventory and price attributes to ITEM.

The revised data dictionary still contains the ITEM, ITEM_EFFECT, STATUS_EFFECT, and ITEM_STATUS_EFFECT entities. The ITEM_EFFECT.effect_type attribute was clarified so that each distinct combat behavior can be represented and queried separately. The ITEM entity continues to store stable information such as the KoL item ID, tradeability, reusability, restrictions, transformations, and unusual behavior. The STATUS_EFFECT and ITEM_STATUS_EFFECT entities remain because named status effects can be reused across multiple items, while the relationship table identifies how each item applies a particular status effect. The kol_item_id attribute was also clarified as the KoL identifier used to connect catalog records with live information retrieved by the script.

One remaining area that may need revision is how much structure should be given to fields such as restrictions, monster_condition, and unusual_behavior. They are currently stored as text because the project is focused on the major combat-item behaviors rather than creating an exhaustive database of every possible condition. As the catalog develops, these fields may need to be separated into additional structured attributes or entities if users need to search them individually. The relationship between item effects and status effects may also need further review as more combat items are documented. Overall, the revisions make the data dictionary more consistent with the process model and better support the goal of querying individual combat behaviors rather than assigning each item a single primary category.

References

- A KoL Wiki. (2026, July 8). *Banishing*. A KoL Wiki. Retrieved August 30, 2026, from <https://wiki.kingdomfloathing.com/Banishing>
- A KoL Wiki. (2024, January 18). *Combat Items*. A KoL Wiki. Retrieved September 1, 2026, from https://wiki.kingdomfloathing.com/Combat_Items
- gausie. (2026, August 22). *kolmafia/src/data/modifiers.txt*. Github. Retrieved August 30, 2026, from <https://github.com/kolmafia/kolmafia/blob/4b19681c000308d08c4e55452c246883847a4d6c/src/data/modifiers.txt>
- KoLmafia Wiki. (2026). CLI reference. KoLmafia Wiki. Retrieved September 1, 2026, from https://wiki.kolmafia.us/index.php?title=CLI_Reference
- KoLmafia Wiki. (2021, March 15). *Custom Combat Script - Kolmafia*. KoLmafia Wiki. Retrieved September 1, from https://wiki.kolmafia.us/index.php?title=Custom_Combat_Script
- KoLmafia Wiki. (2023, March 10). Modifier Maximizer - Kolmafia. KoLmafia Wiki. Retrieved September 1, 2026, from https://wiki.kolmafia.us/index.php?title=Modifier_Maximizer