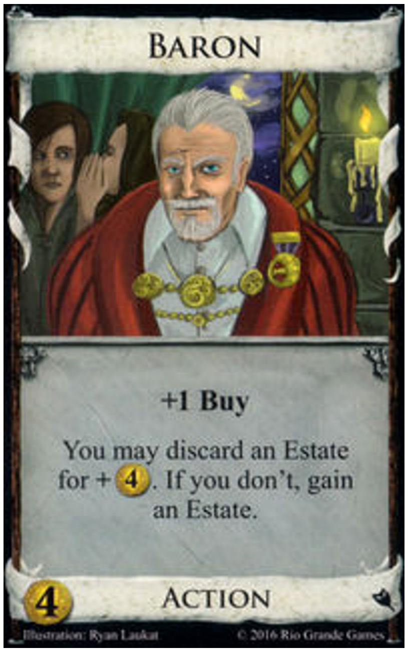
Assignment 1: Dominion description

Sunday, October 6, 2019 6:12 PM

Ryan Holland
hollryan@oregonstate.edu
10/6/19
CS 362
Assignment 1

I played the game online for a couple of hours, but I don't remember seeing any of these cards. They seem to be from expansion sets?

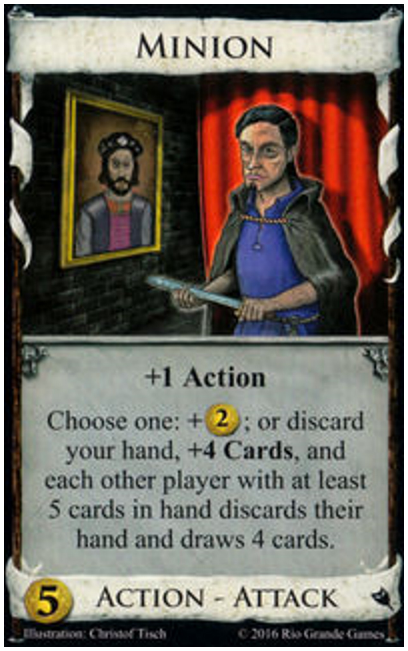
1) Baron


BARON
+1 Buy
You may discard an Estate for + 4 . If you don't, gain an Estate.
4
ACTION
Illustration: Ryan Laukat
© 2016 Rio Grande Games

This guy is an action card which lets you discard an Estate for 4 bucks. I think Estate is the cheapest +1 victory condition card. If you don't choose that option, you get a free Estate to clog up your deck and ruin your game. It also gives you +1 buys so you can spend money on an extra card this turn.

The source code increases the numBuys attribute on the state object by 1. I think "choice1" is a parameter chosen when you play the card. So it splits into a decision tree based on what you chose. It checks your hand for an estate card by iterating through it. Again, all players' hands are an attribute of the state object. Looks like it tracks the players' hands separately, but not the actions, buys, money, etc. I guess that's because different players can use the same counters. The other option (gaining an estate) checks a global estate counter by calling supplyCount(cardType, state). If one is available, it calls gainCard() to put it in the player's hand and decrements the estate counter.

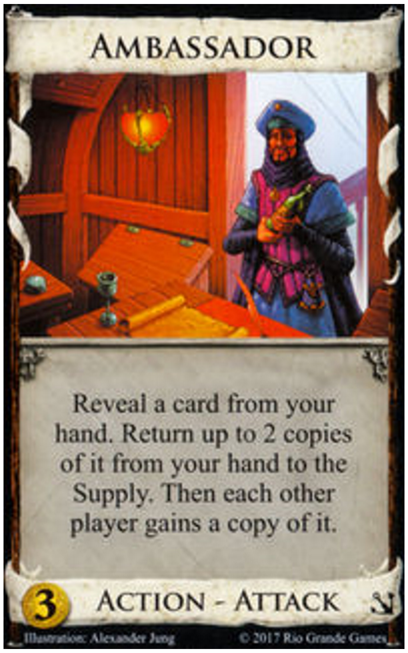
2) Minion


MINION
+1 Action
Choose one: +2; or discard your hand, +4 Cards, and each other player with at least 5 cards in hand discards their hand and draws 4 cards.
5
ACTION - ATTACK
Illustration: Christof Tisch
© 2016 Rio Grande Games

Looks like a decent card. +1 action so you can use another action afterwards. +2 money is ok, but the discard thing is the main ability. I guess it levels the playing field after they use Militia.

It adds +1 to the actions counter. Then it looks at what decision you made. If you chose money, it adds to global coins counter in state. If not, it discards your hand, draws 4 cards (somewhat strangely, it does this with a loop rather than a parameter in drawCard), and deletes the enemy's hand too, then gives them new cards.

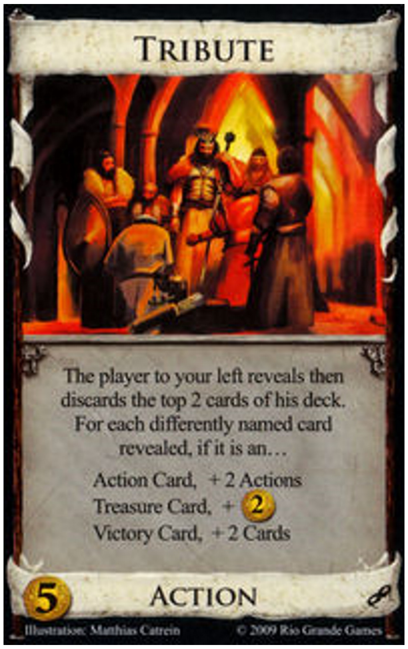
3) Ambassador


AMBASSADOR
Reveal a card from your hand. Return up to 2 copies of it from your hand to the Supply. Then each other player gains a copy of it.
3
ACTION - ATTACK
Illustration: Alexander Jung
© 2017 Rio Grande Games

I guess you should use it on a curse or a card you don't want, like copper, to clean your deck and clog the enemy's.

The code lets you choose to return 1 or 2 copies of a card. It returns an error if choice1 is equal to hand position (which I guess would mean Ambassador itself.) Then choice2 is probably the number of cards to discard -- 1 or 2. It also counts the number of the card you actually have (calling it "j"), and if you don't have enough copies, it doesn't let you continue. I really like the idea of putting "if (debug) {} " print statements, and I will steal it. So then the card increases the global counter for the chosen card by j, and puts the card in others' hands, then discards it from your hand.

4) Tribute



The player to your left reveals then discards the top 2 cards of his deck. For each differently named card revealed, if it is an…

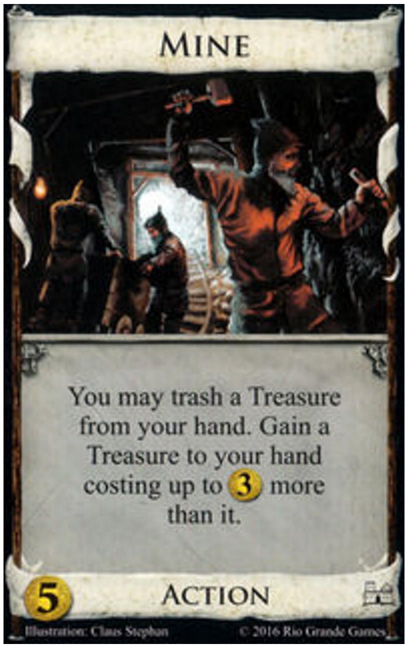
Action Card, +2 Actions
Treasure Card, + 2
Victory Card, +2 Cards

This card gives you a random bonus: 2 actions, 2 cards, 2 gold. I guess it could mess with an enemy who somehow chose what cards are on their deck. I would consider it too unpredictable and expensive.

This card has its own variable local to the cardEffect() function. It adds the revealed cards to that variable (as long as the enemy even has enough cards to reveal.) The variable holds 2 cards because that's what Tribute looks at. Then the code looks at each of those stored cards' type with an if statement, and changes state to give you the correct benefit.

Did you know the chicks of the blue-footed booby bird practice siblicide in times of nutrient deficiency? This type of bird spaces out its eggs so that the nest is shared between an older and a younger chick, unlike most birds. If they don't get enough food, the older chick will kill the younger chick. Actually, at first it tries to share, but if it seems dire enough, he just offs the younger one. It's sad, but that's the way of the jungle. Yep, I spend a lot of time on Wikipedia.

5) Mine



You may trash a Treasure from your hand. Gain a Treasure to your hand costing up to 3 more than it.

This card lets you throw out a money card and replace it with a better one, which you can then play. The wiki says it's a weak card.

The code makes a variable j to store the discarded card. It gets that card from the hand method on the global state variable, with the index of currentPlayer, then the nested index of choice1, the index of the discarded card in the hand. It calls getCost on choice1, then adds 3 to see if it's more than choice2. If you made it past the validations, it calls gainCard and passes it your choice, the state, 2 (for "goes into hand", according to earlier code), and currentPlayer. After giving you the new card, it first discards the MINE card (not the old Treasure card), and then searches your hand for the old Treasure card and trashes it. Actually, it looks like it just discards it… The last param you pass to discardCard is a 0 or 1, and if you want to trash it, it’s a 0. We pass a 0…