

# Dominion Protocol v2

February 12, 2012

Figure 1 shows a revised grammar for `action` and `play`. The extension adds seven action cards and their associated plays.

Some specific constraints on the new `plays` (in addition to the corresponding card being in the player's `hand` and the game state having non-zero `actions`):

- `(act cellar card ...)` — Each `card` must correspond to a distinct card in the player's `hand` (and also distinct from the `cellar` card itself).
- `(act market)` — No extra constraints.
- `(act remodel card card)` — The first `card` must be in the player's `hand`, the second `card` must be in the `supply`, and the cost of the second `card` must be no more than the cost of the first `card` plus two.
- `(act smithy)` — No extra constraints. Note that the presence of `smithy` makes it possible to drive both the player's `deck` and `discards` to empty. In that case, a further attempt to draw a card (via any action) produces no new cards for the player's `hand`.
- `(act village)` — No extra constraints.
- `(act woodcutter)` — No extra constraints.
- `(act workshop card)` — The `card` must be in the `supply`, and it must cost no more than four.

```
action = mine ; from version 1
| cellar
| market
| remodel
| smithy
| village
| woodcutter
| workshop

play = .... ; all from version 1
| (act cellar card ...)
| (act market)
| (act remodel card card)
| (act smithy)
| (act village)
| (act woodcutter)
| (act workshop card)
```

---

Figure 1: Dominion protocol, changes for version 2