

## DevelopmentCard - JSON format

this document shows how cards are represented into the game. All cards are written in JSON and must follow a well-defined syntax.

let's take as first example the card named "Trading Town". Following the JSON format explained in detail later, it will be abstract in this way:

```
{
  "name": "Trading Town",
  "period": 3,
  "permanentPayload": {
    "COINS": 3
  },
  "instantEffect": "ADD",
  "instantPayload": {
    "SERVANTS": 1,
    "COINS": 1
  },
  "permanentEffect": "ADD",
  "minimumActionValue": 1,
  "cardType": "TERRITORYCARD",
  "exclusivePermanentEffect": "TRUE"
}
```

the semantic of the various field is the following:

- **name**: the name of the card
- **cost**: the cost of the card (in this case null because territory cards has no cost)
- **requirements**: the necessary requirements, understood as resources, that the player must own for pay the card's cost (and so take it).
- **instantEffect**: the typology of instant effect which characterizes the card
- **instantPayload**: this field specifics the behavior of the specific effect of the card. the arguments which fill this field will be used by the **EffectBuilder** to generate the correct effect of the card. The parser (i.e. **JsonImporter**) will associate the effect to the card.
- **permanentEffect**: the typology of instant effect which characterizes the card
- **permanentPayload**: this field specifics the behavior of the specific permanent effect of the card. the arguments which fill this field will be used by the **EffectBuilder** to generate the correct effect of the card. The parser (i.e. **JsonImporter**) will associate the effect to the card.
- **minimumActionValue**: the minimum action value required by the card to activate the permanent effect of the same.

- **period**: the card period.
- **cardType**: the typology which characterizes this card.
- **exclusivePermanentEffect**: if setted true this flag indicate that only one of the permanent effect characterizing the card can be apply.

this format can be extended to cards which have more than one instant effect. As example consider the following card:

```
{
  "name": "Cardinal",
  "period": 3,
  "cost": [
    {
      "COINS": 4
    }
  ],
  "instantEffect": [
    "ADD",
    "ACTION"
  ],
  "instantPayload": [
    {
      "FAITH_POINTS": 2
    },
    {
      "TYPE": "HARVEST",
      "REGIONID": 1,
      "BONUSACTIONVALUE": 4,
      "EXCLUSIVEBONUS": null,
      "BONUSRESOURCE": null,
      "FLAGREGION": null
    }
  ],
  "cardType": "CHARACTERCARD",
  "exclusivePermanentEffect": "TRUE"
}
```

Cardinal has two instant effect, included into one JSON array, one of type ADD and the other of type ACTION (payload format will be explained later). Payload are included under the voice “instantPayload” as an array made by objects.

**WARNING: the order with which payloads appear into the “instantPayload” array must be the same order with which the respectively references (i.e. ADD or ACTION in this example) appears into “instantEffect” string array. In the example is not admitted the AC-**

**TION** payload occupies the first position into the array because in this case it will be wrongly associated with an **ADD** effect

## Effect Payload

6 different payload typology can be identified:

- **ADD** : as the name indicates, an **ADD** effect add the indicated resources to the player's resources. Payload format is:

```
{ "WOOD" : 2,  
  "STONE": 4,  
  "MILITARY_POINTS" : 2,  
  ...  
}
```

- **CHANGE**: allow to exchange the resources under **RESOURCEOUT** field in those appearing under **RESOURCEIN** field. Payload format:

```
[  
  {  
    "RESOURCEIN": {  
      "STONE": 1  
    },  
    "RESOURCEOUT": {  
      "VICTORY_POINTS": 3  
    }  
  },  
  {  
    "RESOURCEIN": {  
      "STONE": 3  
    },  
    "RESOURCEOUT": {  
      "VICTORY_POINTS": 7  
    }  
  }  
]
```

for example this effect allows to change 1 stone for 3 victory points or 3 stone for 7 victory points. If the change doesn't offer any choice possibility array notation can be omitted. The effect will open a context on client-side screen to allow the client to choose which change apply.

- ACTION: allows to perform one action of type TYPE whit action value ACTIONVALUE applying the discount BONUSRESOURCE when taking a new card (the EXCLUSIVEBONUS flag is needed to indicate if the discount is applicable only to one specific resource, if the diskonut field allows to choose more then one resource to discount). The field REGIONID is needed to individuate a specific region on the board (for example the action can be performed only on a yellow tower). the FLAGREGION attribute if setted to the specific value “ALL” indicate that there is no limit on the region on which the action can be performed (for example the action can be performed on all the towers, without limit on the tower color). Payload format:

```
{
  "TYPE" : "HARVESTACTION" ,
  "REGIONID" : 5,
  "BONUSACTIONVALUE" : 3,
  "BONUSRESOURCE" : {"WOOD" : -1,
                     "COINS" -1},
  "EXCLUSIVEBONUS" : true,
  "FLAGREGION": "ALL"
}
```

- BONUS: for each QUANTITY of CARD or RESOURCE (values assumed by the attribute TYPE) owned by the player, BONUS effect allows to increment the resource under the field INCREASE of INCREASINGQUANTITY unity. Payload format:

```
{ "TYPE" : "CARD",
  "FOREACH": "CHARACTERCARD",
  "QUANTITY" : 1,
  "INCREASE" : "MILITARY",
  "INCREASINGQUANTITY": 2}
```

for example this effect allows to increase the player’s military points of 2 for each carachter card owned by the player.

- PRIVILEGE: allows to consume a council privilege . A field NUMBER indicates the number of council privilege which can be spent. A COST field is used for particular CHANGE effects which allows to change resources or cards with council privilege. According to this, those effects are mapped as PRIVILEGE but decreasing the player resources. Payload format:

```
{ "NUMBER": 1,
  "COST": { "COINS":1, "WOOD":3 }
}
```

Like CHANGE effect, PRIVILEGE effects need the client cooperation. So when a PRIVILEGE is applied a context will be show on client-side screen to allow the client to choose which resource he want to convert his council privileges.

- PERMANENT: those effects are used to map permanent effect characterizing character cards. The payload has the same format of an ACTION effect, the difference between this effect type and ACTION type is that ACTION is always an instant effect, while PERMANENT is often used like permanent effect. Payload format:

```
{ "TYPE" : "HARVESTACTION" ,
  "REGIONID" : 5,
  "BONUSACTIONVALUE" : 3,
  "BONUSRESOURCE" : [{"WOOD" : -1,
                      "COINS" -1},{ "STONE" : -1}],
  "EXCLUSIVEBONUS" : true,
  "FLAGREGION": "ALL" }
```

According to this format, more complex cards are in this way abstracted:

```
{
  "name": "Fortress",
  "period": 3,
  "cost": [
    {
      "COINS": 2,
      "WOOD": 2,
      "STONE": 4
    }
  ],
  "permanentPayload": [
    {
      "VICTORY_POINTS": 2
    },
    {
      "NUMBER": 1
    }
  ],
  "instantEffect": "ADD",
  "instantPayload": {
    "VICTORY_POINTS": 9
  },
  "permanentEffect": [
    "ADD",
    "PRIVILEGE"
  ]
}
```

```

    ],
    "minimumActionValue": 5,
    "cardType": "BUILDINGCARD",
    "exclusivePermanentEffect": "FALSE"
  }
}

{
  "name": "Treasury",
  "period": 2,
  "cost": [
    {
      "WOOD": 3
    }
  ],
  "permanentPayload": [
    {
      "RESOURCEOUT": {
        "VICTORY_POINTS": 3
      },
      "RESOURCEIN": {
        "COINS": 1
      }
    },
    {
      "RESOURCEOUT": {
        "VICTORY_POINTS": 5
      },
      "RESOURCEIN": {
        "COINS": 2
      }
    }
  ],
  "instantEffect": "ADD",
  "instantPayload": {
    "VICTORY_POINTS": 4
  },
  "permanentEffect": [
    "CHANGE",
    "CHANGE"
  ],
  "minimumActionValue": 3,
  "cardType": "BUILDINGCARD",
  "exclusivePermanentEffect": "FALSE"
}

```

```

    "name": "Carpenter's Shop",
    "period": 1,
    "cost": [
      {
        "COINS": 1,
        "WOOD": 2
      }
    ],
    "permanentPayload": [
      {
        "RESOURCEOUT": {
          "COINS": 3
        },
        "RESOURCEIN": {
          "WOOD": 1
        }
      },
      {
        "RESOURCEOUT": {
          "COINS": 5
        },
        "RESOURCEIN": {
          "WOOD": 2
        }
      }
    ],
    "instantEffect": "ADD",
    "instantPayload": {
      "VICTORY_POINTS": 3
    },
    "permanentEffect": [
      "CHANGE",
      "CHANGE"
    ],
    "minimumActionValue": 4,
    "cardType": "BUILDINGCARD",
    "exclusivePermanentEffect": "FALSE"
  }
}

{
  "period": 1,
  "cost": [
    {
      "MILITARY_POINTS": 3
    }
  ],

```

```

    "permanentPayload": {
      "VICTORY_POINTS": 5
    },
    "instantEffect": "ADD",
    "instantPayload": {
      "FAITH_POINTS": 2
    },
    "permanentEffect": "ADD",
    "minimumActionValue": 0,
    "requirements": {
      "MILITARY_POINTS": 5
    },
    "cardType": "VENTURECARD",
    "name": "Fighting Heresies",
    "exclusivePermanentEffect": "TRUE"
  }
}

```

## ExcommunicationCard - JSON format

ExcommunicationCard follow a similar format as DevelopmentCard. They are characterized by the name, the period and an instant or permanent effect. Instant effect are apply only one time when the excommunication card is activated. those effects add a particular flag to a list of flag interanal to the Player object. Inside the code, when the player is flagged, particular function are performed in order to apply the excommunication effects.

Permanent effects Instead are applied only one time when the effect has been activated like instant effects. But the Effect is loaded into the effect list of the Player object. In this way they are activated every time the player effects are triggered (for example by an action).

Example of ExcommunicationCard are the following:

```

{
  "instantEffect": "LESSRESOURCE",
  "instantPayload": [
    "STONE",
    "WOOD"
  ],
  "name": "Malus Stone or Wood",
  "period": 1
}

```

LESSRESOURCE effect will substract -1 to the listed resources every time one of them is added to the player resourceSet.



```

{
  "permanentPayload": {
    "FLAGREGION": null,
    "BONUSRESOURCE": null,
    "BONUSACTIONVALUE": -3,
    "TYPE": "PRODUCTION",
    "EXCLUSIVEBONUS": null,
    "REGIONID": 0
  },
  "name": "Malus Production Action",
  "permanentEffect": "PERMANENT",
  "period": 1
}

```

PERMANENT effect works in the same mode of PERMANENT Development-Card effect. Note that the BONUSACTIONVALUE has a negative value, so this represent a penalty on the action value of the player action.

```

{
  "instantEffect": "FLAG",
  "instantPayload": "NOENDPURPLE",
  "name": "No Victory Points Purple",
  "period": 3
}

```

FLAG effects add the specific flag indicated under the voice instantPayload to the flag list owned by the Player object. In this case the activation of this excommunication card will flag the player as “NOENDPURPLE”, which means that in the final score computation the VENTURECARD won’t be calculated.

other excommunication effects are: \* LESSDICE: decrease the action value of family member by 1 \* NO\_MARKET\_ACTION: (FLAG-type effect) prevents the player affected by this effect to make action in the MARKETREGION. \* DOUBLESERVANTS: (FLAG-type effect) increase the action value of the family member by 1 will cost 2 SERVANTS. \* LESSFORVISTORY: (FLAG-effect): map this card *At the end of the game, before the Final Scoring, you lose 1 Victory Point for every 5 Victory Points you have.* \* LESSFORMILITARY: map this card *At the end of the game, you lose 1 Victory Point for every Military Point you have.* \* LESSFORBUILDING: map this card *At the end of the game, you lose 1 Victory Point for every wood and stone on your Building Cards’ costs.* \* LESSFORRESOURCE: map this card *At the end of the game, you lose 1 Victory Point for every resource (wood, stone, coin, servant) in your supply on your Personal Board.*

## LeaderCard - JSON format

LeaderCard are very similar to DevelopmentCard and ExcommunicationCard in their JSON representation, here an example:

```
{
  "requirements": {
    "RESOURCE": {
      "MILITARY_POINTS": 12
    }
  },
  "instantPayload": {
    "WOOD": 1,
    "STONE": 1,
    "COINS": 1
  },
  "instantEffect": "ADD",
  "name": "Giovanni delle Bande Nere"
}
```