

Protocol description of Capid

Capid uses as protocol the protocol of Monopd 0.9.3 as basis. With some changes and extensions. Capid also does not support deprecated Monopd 0.9.3 messages.

Notes:

Data types:

Capid uses the following data types:

Boolean: Is 1 for true, or 0 for false

Integer: An Integer value. It is always -1 or higher. The value -1 has a special meaning.

String: A String.

Messages to server:

Every Client can send orders to Capid. (The orders are listed later) Every order must end with a newline and each order must be in a single line. Every incoming line will be threaded as single command.

Messages to client:

Capid uses XML-formatet messages to send Clients information. Each message is in one line line followed by a newline

Every message starts with `<monopd>` and ends with `</monopd>`.

So its a good idea to put all Messages from Capid in a DOM-Tree and parse this one.

Connection to server:

When a TCP connection to Capid is build up, Capid sends to the client the introduction-message:

```
<monopd><server version="%1" /></monopd>
```

%1 will be replaced with a string that contains the current version. The version-string is in the form "[version] Capid", where [version] is replaced by the current version of Capid.

After sending Capid waits for an answer from the client, that is:

```
.n%1
```

%1 will be replaced by any name the client wishes to have.

As answer for that you get the message:

```
<monopd><client playerid="%1" cookie="%2" /></monopd>
```

%1 will be replaced with an integer that contains your playerid

%2 will be replaced with a cookie string. It can be uses to reconnected, after your client have crashed.

After that the connection is finished. The client should be ready to receive messages from server.

List of commands to server:

Now a list of all commands that are supported by Capid.

Note: All Variables will be replaced by "%1", "%2", "%3", and so on. It will be explained with which data they have to be filled

- .n%1

Sets the client players name to %1

%1 – String – The wished name for you

Note: The actual name is given by Capid. So don't set the name manually. Wait for a

message by Capid that sets the name.

- **.pi%1**

Sets the client players image to %1

%1 will be converted to an image. The name for that image will be set to “av[id]”, where [id] is replaced by the players-id that sended the image.

Note: You client will probably not know the image “av[id]”. In this case the client can ask for images with “.im[imgName]”.

- **.gn%1**

Starts a new game with of type %1.

%1 – String – The type of the new game.

Note: %1 must be known by Capid for success. Capid sends a list of templates for games, for all types that are known. See the <gameupdate /> message how to get the known Gametypes.

- **.gj%1**

Trys to join the game with the id %1

%1 – Integer – The Id of the game to be joined.

Note: Every game has its own unique id. The client will be always informed about the games. See <gameupdate /> message how to get the valid ids.

- **.gd%1**

Sets the description of the current game.

%1 – String – The new game description

Note: Only the master of a game can change the description

- **.d**

Disconnects from Server.

Note: Use this command always to quit your client, if it is connected to Capid.

Note: You will be disconnected from the game you are in. Everything that you currently own will be owned by the bank.

- **.gx**

Leave a game. Everything that you own gets back to bank. But you stay connected to Capid.

- **.gc%1:%2**

Sets the gameconfig with id %1 to the value %2

%1 – Integer – The id of the gameconfig to change

%2 – Boolean|String|Integer – The new value.

Note: Only the master of the game can change them.

See <configupdate /> for more information.

- **.gs**

Starts the current game. Nobody can now join this game and the game starts.

Note: Only the master of the game can start a game

- **.r**

Roll the dices for movement

Note: This only works, if you may roll. See <playerupdate /> to see how to determ, if a player may roll.

- **.t%1**

Confirms the latest player movement to position %1. If you want to animate the player movements, send this command, if your movement if finished. Otherwise send it immediatly. If you animate your player movement you should confirm all position that the player passes.

%1 – Integer – The position for that the latest movement is confirmend

- **.E**

End the current Turn.

Note: Only the player that is on turn can send this command. It works only if you could by

an estate and auctions are disabled. (If for the player on turn can_buyestate is set to 1 and canauction is set to 0)

- **.eb**
Buys the current estate.
Note: This works only, if you are on turn, if can_buyestate is set to 1 for you and if you have enough money to buy it.
- **.em%1**
Toggles the mortgage of estate with id %1
%1 – Integer – The id of the estate to toggle mortgage for.
Note: This only works, if you can toggle the mortgage of the estate, and if you can pay it (for unmortgaging).
- **.es%1**
Sells estate %1
%1 – Integer- The id of the estate to be sold
Note: This only works, if selling is enabled in the game
- **.hb%1**
Buys a a house on estate with id %1
%1 – Integer – The id of the estate to buy a house for
Note: This only works, if you can buy a house for that estate, and if you can pay the house.
- **.hs%1**
Sells a house on estate with id %1
%1 – Integer – The id of the estate to sell a house for
Note: This only works, if you can sell a house for that estate.
- **.jr**
Trys to roll in jail. On doubles you get free.
- **.jp**
Pays for jail.
Note: You will need the money to be able to pay.
- **.jc**
Uses a “Out of Jail” card to get free.
- **.Tn%1**
Starts a new trade with the player with the id %1
%1 – Integer – The id of the player to begin a trade with
- **.Tr%1**
Rejects the trade with the id %1
%1 – Integer – The id of the trade to reject
Note: You cannot reject trades, that you are not involved
- **.Ta%1:%2**
Accepts the current Revision (rev %2) trade with id %1
%1 – Integer – The id of the trade to accept
%2 – Integer – The revision that is accepted
- **.Te%1:%2:%3**
Add the estate with id %2 with target playerid %3 to the trade with id %1
%1 – Integer – The trade to add the estate
%2 – Integer – The id of the estate to add.
%3 – Integer – The id of the player that should get the estate
Note: It is possible that further players are added to the trade.
- **.Tc%1:%2:%3**
Add the card with id %2 with target playerid %3 to the trade with id %1
%1 – Integer – The trade to add the card
%2 – Integer – The id of the card to add.

%3 – Integer – The id of the player that should get the card
Note: It is possible that further players are added to the trade.

- `.Tm%1:%2:%3:%4`

Adds a Money to a trade

%1 – Integer – The trade to add the money

%2 – Integer – The id of the player that gives the money

%3 – Integer – The amount of the money

%4 – Integer – The id of the player that gets the money

Note: It is possible that further players are added to the trade.

- `.ab%1:%2`

Bid for an auction

%1 – Integer – The id of the auction

%2 – Integer – Your bid for that auction

- `.R%1`

Reconnects a player (Useful, if your client crashed)

%1 – String – The cookie (You get it after handshaking with the server)

- `.T$`

Chooses to pay the fixed taxamount (Only if you have to choose)

- `.T%`

Chooses to pay the percentage based tax. (Only if you have to choose)

- `.l%1`

Sets the language that your client receives to %1

%1 – String – The languagecode for the wished language

The used format is “[lang]_[country]”, where [lang] is replaced with the ISO 639-1 language-code (see http://en.wikipedia.org/wiki/List_of_ISO_639-1_codes).

[country] is replaced with the ISO 3166-1 alpha-2 country-code (see http://en.wikipedia.org/wiki/ISO_3166-1_alpha-2).

- `.im%1`

Requests the image %1 from the server.

%1 – String – The name of the image to get gotten. If there is no image called %1, nothing will be returned.

Server messages:

For bedder readability `<monopd>` and `</monopd>` will be left out.

Messages:

```
<server version="%1" />
```

Introduction by Capid.

%1 – String – Version of Server

```
<client playerid="%1" cookie="%2" />
```

Information about your client.

%1 – Integer – Your playerid (So you know with player you are controlling)

%2 – String – A Cookie. Used for reconnecting.

```
<msg type="%1" value="%2" />
```

Messages from Server. Including chat.

%1 – String – Can be `chat`, `error` or `info`

%2 – String – The actual message

Optional attributes:

Name	Type	Content
playerid	Integer	The player that sented a chat message. When type="chat" this attribute is sendet.

```
<display estateid="%1" text="%2" cleartext="%3"
clearbuttons="%4">...</display>
```

Used to send the clients events in the game. They should be displayed to see what happens.

%1 – Integer – The id of the estate to show. If set to -1 no estate should be shown

%2 – String – The Test to display. Should be appended to previous-display-texts.

%3 – Boolean – If 1 all previous-text should be cleared

%4 – Boolean – If 1 all shown buttons should be cleared

Otional subobjects:

```
<button command="%1" caption="%2" enabled="1" />
```

This subobject is only sendet to the player that is on turn. It is used to show buttons. By clicking on such a button the given command should be send to Capid.

%1 – String – A command to send, if clicked

%2 – String – A Caption for the button

```
<playerupdate playerid="%1" />
```

Used to inform all clients about changed properties of a player.

If you don't have a player with that id. Create it.

%1 – Integer – The Id of the player with changed properties.

Optional attributes:

Name	Type	Content
game	Integer	The id of the game the the player is in. -1 if the player is not in a game
name	String	The name of the player
image	String	The image name. If your client does not the given name, it can ask for the image with ".im[imageName]".
money	Integer	The current money
location	Integer	The current location
jailed	Boolean	1 if player is in jail
directmove	Boolean	1 if movement should not be animated.
bankrupt	Boolean	1 if player is bankrupt
hasturn	Boolean	1 if player is on turn
can_roll	Boolean	1 if player may roll
can_buyestate	Boolean	1 if player can buy current estate
spectator	Boolean	1 if player is a spectator
canauction	Boolean	1 if player can start an auction for the current estate
canusecard	Boolean	1 if player can use a card to leave jail

```
<deleteplayer playerid="%1" />
```

The Player with the id %1 should be deleted.

%1 – Integer – The id of the player that should be deleted.

```
<estateupdate estateid="%1" />
```

Used to inform all clients about changed properties of an estate.

If you don't have an estate with that id. Create it.

%1 – Integer – The id of the estate with the changed properties.

Optional attributes:

Name	Type	Content
name	String	The name
owner	Integer	The id of the owning player (-1 if unowned)
houses	Integer	The number of houses on it (You can see 5 as a hotel)
houseprice	Integer	The price to buy a house
sellhouseprice	Integer	The price for selling a house (You get that money)
mortgaged	Boolean	1 if estate is mortgaged
group	Integer	The id of the group that estate belongs to (-1, if it doesn't belong to any group)
can_be_owned	Boolean	1 if this estate can be bought during game
can_toggle_mortgage	Boolean	1 if the estate can be (un)mortgaged
can_buy_houses	Boolean	1 if houses can be bought on the estate
can_sell_houses	Boolean	1 if houses can be sold on the estate
money	Integer	The money that this estate contains. That player that lands on this estate gets the money
price	Integer	The price for that estate
mortgageprice	Integer	The price for mortgaging the estate (You get that money)
unmortgageprice	Integer	The price to unmortgage the estate
color	String	HTML colorcode for the estate
bgcolor	String	HTML colorcode for the background
ren0	Integer	Rent with no houses
rent1	Integer	Rent with 1 house
rent2	Integer	Rent with 2 houses
rent3	Integer	Rent with 3 houses
rent4	Integer	Rent with 4 houses
rent5	Integer	Rent with 5 houses

```
<cardupdate cardid="%1" title="%2" owner="%3" />
```

This is used to inform all clients about changed property of a card. (Only owner can change)

If you have no card with that id, create it.

%1 – Integer – The id of the card

%2 – String – The text of the card

%3 – Integer – The id of the player that owns the card. (-1, if the card is unowned)

```
<estategroupupdate groupid="%1" name="%2" />
```

Sended, when the game starts for all groups. You should create such groups, when receiving this message.

%1 – Integer – The id of the group
%2 – String – The name of the group

```
<tradeupdate tradeid="%1" revision="%2" actor="%3" type="%4"
>...</tradeupdate>
```

Sended if there is an update for a trade that you are invovled with.

%1 – Integer – The id of the trade

%2 – Integer – The current revision of the trade

%3 – Integer – The id of the player that started the trade

%4 – String – The status of the current trade. Can be new if the trade is new, edit if the trade is free for editing, or completed if all players accepted the current revision, or rejected if a player rejected the trade.

Note: You can also edit a trade with status new.

Optional subobjects:

```
<tradeestate estateid="%1" targetplayer="%2" />
```

This subobject is sendet for each estate that is involved with the trade.

%1 – Integer – The id of the estate

%2 – Integer – The if of the target player (if set to -1 this part of trade was deleted)

```
<tradecard cardid="%1" targetplayer="%2" />
```

This subobject is sendet for each card that is involved with the trade.

%1 – Integer – The id of the card

%2 – Integer – The if of the target player (if set to -1 this part of trade was deleted)

```
<trademoney playerfrom="%1" playerto="%2" amount="%3" />
```

This subobject is sendet for each money transfer that is invovled with the trade

%1 – Integer – The id of the player that gives the money

%2 – Integer – The if of the player that gets the money

%3 – Integer – The amount of the transfer (if set to 0 this part of trade was deleted)

```
<tradeplayer playerid="%1" accept="%2" />
```

This subobject is sendet for each player that is involved with the trade.

%1 – Integer – The id of the player

%2 – Boolean – 1 if the player accepted the current terms of the trade, 0 otherwise.

```
<auctionupdate auctionid="%1" actor="%2" estateid="%3"
highbid="%4" highbidder="%5" status="%6" />
```

Sended if there is a auction in progress.

%1 – Integer – The id of the auction

%2 – Integer – The if of the player that started the auction

%3 – Integer – The if of the estate to auction for

%4 – Integer – The current highest bid

%5 – Integer – The id of the player with the highest bid

```
<gameupdate gameid="%1" />
```

Used to inform all clients about changed properties of a game. It is also uses to inform the clients about all known gametype.

%1 – Integer – The id of the game. It it is -1 this message describes a template of the game.

Optional attributes:

Name	Type	Content
gametype	String	The type of the game. If gameid is -1 this attribute is sended. You should create a template with that gametype.
name	String	The name
description	String	A short description

status	String	The current status. Can be config, or init, or run, or end
players	Integer	The number of players that are currently in the game
canbejoined	Boolean	1 if you can join this game
minplayers	Integer	The minimum number of players that are needed to start a game
maxplayers	Integer	The maximum number of players that are allowed in this game
master	Integer	The id of the current Master of the game.
allowestatesales	Boolean	Indicates if estates can be sold. 1 if it is allowed, 0 otherwise

```
<deletegame gameid="%1" />
```

This game with the id %1 should be deleted.

%1 – Integer – The id of the game to be deleted.

```
<configupdate configid="%1" name="%2" title="%3" type="%4"
value="%5" edit="1" />
```

Used to inform all clients about the current configuration of the game. If you are the master of the game you can edit them. See the .gc%1:%2 command how to do this.

%1 – Integer – The id of the configoption

%2 – String – The name of the configoption

%3 – String – The shown text of the config option

%4 – String – The type of the configoption. Can be string or int or bool

%5 – Integer|String|Boolean – The current value of the configoption

```
<imageupdate name="%1" data="%2" />
```

Used for sending images to the client. Will be send to the client that asks for image %1.

Will also be send, if a client changes its image.

See .im%1 command how to get an image.

%1 – String – The name of the image

%2 – String – Hex coded raw data of an image. Pass it directly to an image library.