

Key-value DB data provider functional object messages protocol

Request protocol

Json fomate of message:

```
{
  "type": "<action_type>",
  "data": "<message_body>"
}
```

action_type - numeric integer {0, 1, 2, 3, 4}; 0 – put, 1 – get, 2 – delete, 3 – check, 4 – admin manage commands

message_body - json data different for each action type.

Action types:

0 – put:

```
{
  "documents": { "<document_id>": "<document_body>", ... }
}
```

document_id - numeric integer unique document Id, key of DB data action.

document_body - document body, value of DB data action.

1 – get:

```
{
  "documents": [ "<document_id>", ... ]
}
```

document_id - unique document Id, key of DB data action. **If document_id is "*" value – get all documents from db. (Not implemented yet and possibility of implementation under research.)**

2 – delete:

```
{
  "documents": [ "<document_id>", ... ]
}
```

document_id - unique document Id, key of DB data action. **If document_id is "*" value – delete all documents from db. (Not implemented yet and possibility of implementation under research.)**

3 – check:

```
{
  "documents": [ "<document_id>", ... ]
}
```

document_id – unique document Id, key of DB data action. If **document_id** is “*” value – check all documents in db. (Not implemented yet and possibility of implementation under research.)

4 – manage:

```
{  
  “documents”: [“<command_id>”, ...]  
}
```

command_id – Id of management command. 0 – Get DB engine name. 1 – clean (complete remove all data and initialize the db engine)

5 – list: (NOT IMPLEMENTED)

```
{  
  “documents”: []  
}
```

Response protocol

```
{
  "error_code": "<error_code>",
  "error_message": "<error_message>"
  "data": "<response_body>"
  "time": "<elapsed_time>"
}
```

error_code - error state code, 0 - no errors, >0 - some error

error_message - error description text

response_body - result json depends on requested action type.

elapsed_time - internal common execution time (including request parsing and response creation), micro sec

Response body field value depends on action type:

0 - put:

```
{
  "documents": { "<document_id>": "<document_state>", ... }
}
```

document_id - unique document Id, key of DB data action.

document_state - numeric document state {0, 1}. 1 - fault, document was not inserted in to the DB. 0 - success, document was inserted in to the DB.

1 - get:

```
{
  "documents": { "<document_id>": "<document_body>", ... }
}
```

document_id - numeric integer unique document Id, key of DB data action.

document_body - document body, value of DB data action.

2 - delete:

```
{
  "documents": { "<document_id>": "<document_state>", ... }
}
```

document_id - unique document Id, key of DB data action.

document_state - numeric document state {0, 1}. 1 - fault, document was not deleted from the DB. 0 - success, document was deleted from the DB.

3 - check:

```
{
  "documents": {"<document_id>": "<document_state>", ...}
}
```

document_id - unique document Id, key of DB data action.

document_state - numeric document state {0, 1}. 1 - document not present in the DB. 0 - document present in the DB.

4 - manage:

```
{
  "documents": {"<command_id>": "<result_context>", ...}
}
```

command_id - unique document Id, key of DB data action.

result_context - string content with printout of command execution results.

0 - name of DB engine.

1- DB engine cleanup result message (**NOT IMPLEMENTED**)

5 - list: (NOT IMPLEMENTED)

```
{
  "documents": {"<document_id>", ...}
}
```

document_id - numeric integer unique document Id, key of DB data action.

Returns array of all Id of all documents stored in the engine.

Possible errors

Code	Description
1	Parse error - wrong json format or another parse error.
2	Action type error (not supported, not recognized, etc...)
3	General internal error of functional object
1000	Specific internal errors
1001	...