Table of Contents

Вывод в json 3

Вывод в json

Все CLI-команды могут выводить данные в формате JSON. Для этого в вызове `fdpi_cli` указываются опции:

```
fdpi_cli --json --strict <команда> [аргументы]*
```

Формат JSON позволяет обрабатывать данные CLI внешними скриптами: фильтровать, делать выборки и пр.

Мы осознанно не даем полного описания полей и их назначения для каждой CLI-команды, так как предполагаем, что

- текстовый выхлоп достаточно информативен;
- сопоставление текстового выхлопа команды с JSON-выхлопом дает достаточно полную картину, какое поле какую информацию несет.

Но мы также стараемся не изменять имена полей и не удалять поля: какое бы уродливое имя поля ни было выбрано по незнанию или недоразумению, раз оно вошло в релиз СКАТа, изменять его мы не будем. Ровно та же ситуация с удалением полей: если при развитии СКАТа то или иное поле становится не нужным, мы будем его выводить с default-значением в JSON (но не будем в текстовый выхлоп). Возможность и право добавлять новые поля в существующие структуры мы оставляем за собой и активно этим пользуемся, так как СКАТ активно развивается и появляются новые свойства у ранее заданных сущностей.

Также мы оставляем за собой право изменять текстовый формат вывода команд CLI, поэтому не надо пытаться грепать в скриптах текстовый выхлоп команд CLI, - пользуйтесь JSON, мы стараемся поддерживать его в более стабильном состоянии.

Многие CLI-команды, выводящие дамп внутренних баз данных СКАТа, выводят именно дамп, со всеми внутренними полями и их значениями, поэтому следует предупредить о некоторых особенностях вывода в JSON.

СКАТ в ответ на команду может "выплюнуть" десятки и сотни мегабайт данных, если это команда дампа типа `show all`. Сформировать такой ответ в одном блоке СКАТ не может, - для этого пришлось бы аллоцировать и реаллоцировать многие мегабайты памяти, что может катастрофически сказаться на основной работе СКАТа. Поэтому СКАТ выдает результат команды пачками сравнительно малого размера (несколько килобайт бинарных данных максимум). Внешне этого не видно, утилита fdpi_cli скрывает это при выводе в текстовом виде. Но при выгрузке ответа в JSON-формате вы увидите эти пачки.

Выхлоп CLI в JSON-формате выглядит так (общий скелет):

```
[
  {
  # пачка 1
  },
  {
  # пачка 2
  },
```

```
  ...
]
```

то есть это массив пачек (объектов). Каждая пачка начинается с `result`:

```
[
  {
    "result": {
      "result_code": 0
    },
  # данные пачки - зависят от команды
  },
  {
    "result": {
      "result_code": 0
    },
  # данные пачки - зависят от команды
  },
  ...
]
```

`result.result_code=0` - это успешное выполнение. Для неуспешного значение `result.result_code` будет отличен от нуля и может присутствовать поле `result.message` - текстовое сообщение об ошибке:

```
[
  {
    "result": {
      "result_code": 1,
      "message": "что-то пошло не так"
    }
  # при неуспехе данных может и не быть, а может быть какой-то огрызок данных
]
```

Надо учитывать, что `result.message` может присутствовать и при успешном выполнении, так что единственным критерием, что пачка сформирована без ошибок, является `result.result_code=0`.

При формировании *каждой* пачки может произойти ошибка. При возникновении ошибки СКАТ не будет пытаться выполнить команду далее, - всякая ошибка является фатальной. Поэтому если в выхлопе JSON n-ая (последняя) пачка содержит `result.result_code` отличным от нуля, - это значит, что вся команда выполнилась с ошибкой и такой JSON нужно выбросить.

Иначе это можно сформулировать так: команда выполнена успешно, если каждый объект JSON-массива ответа имеет `result.result_code=0`. Только при этом условии можно работать с JSON-данными.