

Предложение по организации топиков и обмена сообщениями

Топики и ключи

Предлагается использовать четыре топика. (Именование топиком нужно согласовать)

Для инетификации запросов и ответом предлагается использовать

- request_id - для запросов выписки. Использовать как параметр в payload как так же как ключи сообщений(kafka key)
- response_id - для ответных сообщений (или частей) с результатом

Топики:

- way4.cust.statements.dkkfm.request — входящие запросы
- way4.cust.statements.dkkfm.result.chunk — части (чанки) ответов выписки
- way4.cust.statements.dkkfm.result.manifest — манифест завершённой выписки (сводка по частям)
- way4.cust.statements.dkkfm.result.error — ошибки обработки/доставки
-

Ключи сообщений (Kafka key):

- **Запросы:** way4.cust.statements.dkkfm.request — kafka-key: request_id
- **Ответ-часть:** way4.cust.statement.sdkkfm.result.chunks— kafka-key: response_id (уникально для конкретного ответа; все части одного ответа имеют одинаковые ключи)
- **Манифест:** way4.cust.statements.dkkfm.result.manifest — kafka-key: request_id
- **Ошибки:** way4.cust.statements.dkkfm.error — kafka-keyP request_id

Идентификаторы и заголовки

Идентификаторы и заголовки запроса

Идентификаторы:

- request_id — генерирует инициатор запроса, для корреляция «запрос#ответ». request_id используется как kafka key.

Заголовки (Kafka headers)

- x-request-id равен request_id
- x-sent-at время отправки выписки в формате UTC "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"

- x-schema: compact (опционально, на будущее, если перейти на компактное именование полей выписки)

Идентификаторы и заголовки ответа выписки

Идентификаторы:

- request_id — инициатор запроса, для для которого подготовлен ответ
- response_id — генерирует сервис выписки при старте формирования ответа, идентификатор ответа

Заголовки (Kafka headers)

Заголовки дублируем в payload, для надёжности.

- x-request-id равен request_id
- x-response-id равен response_id
- x-chunk-idx (1..N) идентифицирует часть сообщения в рамках всей выписки
- x-chunk-total (N) общее кол-во частей выписки
- x-range-rows = start-end/total (например, 1-800/2415) (опционально) строки попавшие в данную часть выписки
- x-sent-at время отправки выписки в формате UTC "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"
- x-schema: compact (опционально, на будущее, если перейти на компактное именование полей выписки)

Идентификаторы и заголовки манифеста

Идентификаторы:

- request_id — инициатор запроса, для для которого подготовлен ответ
- response_id — генерирует сервис выписки при старте формирования ответа, идентификатор ответа

Заголовки (Kafka headers)

- x-request-id равен request_id
- x-response-id равен response_id
- x-sent-at время отправки выписки в формате UTC "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"
- x-schema: compact (опционально, на будущее, если перейти на компактное именование полей выписки)

Идентификаторы и заголовки сообщения с ошибкой

Идентификаторы:

- request_id — инициатор запроса, для для которого подготовлен ответ
- response_id — генерирует сервис выписки при старте формирования ответа, идентификатор ответа

Заголовки (Kafka headers)

Заголовки дублируем в payload, для надёжности.

- x-request-id равен request_id

- x-sent-at (RFC3339) время отправки сообщения
- x-schema: compact (опционально, на будущее, если перейти на компактное именование полей выписки)

Общее описание служебных полей сообщений

- msg - сообщение "request|result-chunk|result-manifest|result-manifest" (опционально, тк все сообщения в разных топиках)
- request_id - инициатор запроса
- response_id - идентификатор ответа
- timestamp - дата/время отправки в формате UTC "yyyy-MM-dd'T'HH:mm:ss.SSS'Z'"
- query - описывает запрос с параметрами
- chunk - описывает часть ответов выписки
 - idx - идентифицирует часть сообщения в рамках всей выписки
 - total - общее кол-во частей выписки
 - rows - диапазон строк попавших в часть выписки
 - start - с какой строки
 - end - по какую строку
 - bytes - объем в байтах(опционально)
- total_rows - общее кол-во строк в выписке
- header - заголовки выписки
- transactions - строки выписки
-

1. Форматы сообщений без разбиения на части(чанки)

1.1.1 Пример запроса по номеру счета

Topic: way4.cust.statements.dkkfm.request **Key:** 11111111-1111-1111-1111-111111111111 (request_id)

Headers:

- content-type: application/json
- x-request-id: 11111111-1111-1111-1111-111111111111
- x-sent-at: 2025-10-28T13:00:00Z

```
{
  "msg": "request",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "timestamp": "2025-10-28T13:00:00Z",
  "query": {
    "account_number": "40702*****5678",
    "date_from": "2025-08-01",
    "date_to": "2025-08-31"
  }
}
```

1.1.2 Пример запроса по номеру карты

Topic: way4.cust.statements.dkkfm.request **Key:** 11111111-1111-1111-1111-111111111111 (request_id)

Headers:

- content-type: application/json
- x-request-id: 11111111-1111-1111-1111-111111111111
- x-sent-at: 2025-10-28T13:00:00Z

```
{
  "msg": "request",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "timestamp": "2025-10-28T13:00:00Z",
  "query": {
    "pan": "220024*****3847",
    "date_from": "2025-08-01",
    "date_to": "2025-08-31"
  }
}
```

1.1.1 Пример запроса по клиенту(mdm-id)

Topic: way4.cust.statements.dkkfm.request **Key:** 11111111-1111-1111-1111-111111111111 (request_id)

Headers:

- content-type: application/json
- x-request-id: 11111111-1111-1111-1111-111111111111
- x-sent-at: 2025-10-28T13:00:00Z

```
{
  "msg": "request",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "timestamp": "2025-10-28T13:00:00Z",
  "query": {
    "client_number": "53000168_30",
    "date_from": "2025-08-01",
    "date_to": "2025-08-31"
  }
}
```

1.2 Пример ответа выписки в одной части (idx=1,total=1)

Topic: way4.cust.statements.dkkfm.result.chunk **Key:** 22222222-2222-2222-2222-222222222222 (response_id)

Headers:

- content-type: application/json
- x-request-id: 11111111-1111-1111-1111-111111111111
- x-response-id: 22222222-2222-2222-2222-222222222222
- x-chunk-idx: 1
- x-chunk-total: 1
- x-range-rows: 1-2/2
- x-sent-at: 2025-10-28T13:00:02Z

Payload (JSON):

```
{
  "msg": "result-chunk",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "response_id": "22222222-2222-2222-2222-222222222222",
  "chunk": {
    "idx": 1,
    "total": 1,
    "rows": {
      "start": 1,
      "end": 2
    },
    "bytes": 856,
  },
  "total_rows": 2,
  "header": {
    "bank_name": "Банк ВТБ (ПАО), филиал № 7701 в г. Москве",
    "account_number": "4081*****812",
    "date_from": "2025-05-14",
    "date_to": "2025-05-14",
    "client_name": "Копусов Владимир Михайлович",
  }
}
```

```

"inn": "",
"birth_date": "1900-01-05",
"client_number": "Test_client_20250508204335",
"total_amount": "21500.00"
},
"transactions": [
{
  "account_number": "4081*****812",
  "contract_number": "2200240220503847",
  "posting_date": "2025-05-14",
  "transaction_id": "203254974625",
  "debit_amount": "20500.00",
  "credit_amount": "0.00",
  "currency": "20500.00",
  "local_amount": "20500.00",
  "transaction_description": "Retail Advice",
  "counter_party": {
    "name": "",
    "inn": "",
    "account_number": "3023*****000",
    "bic": "",
    "bank_name": ""
  },
  "arn": "P1281100QAO",
  "auth_amount": "20500.00",
  "auth_currency": "RUR",
  "transaction_date": "08.05.2025 20:43:56",
  "rrn": "512880043563",
  "member_id": "888887",
  "merchant": {
    "sic_name": "5631 Women's Accessory And Speci",
    "id": "240125298",
    "merchant_name": "Conte",
  },
  "terminal": {
    "id": "Y0070542",
    "location": "",
    "city": "",
    "country": ""
  },
  "authorization_code": "241013",
  "transaction_condition": "CAT PBT Chip Contactless",
  "cpid": "",
  "card_expire": "3205"
},
{
  "account_number": "4081*****812",
  "contract_number": "2200240220503847",
  "posting_date": "2025-05-14",
  "transaction_id": "203254974625",
  "debit_amount": "20500.00",
  "credit_amount": "0.00",
  "currency": "20500.00",
  "local_amount": "20500.00",

```

```

    "transaction_description": "Retail Advice",
    "counter_party": {
      "name": "",
      "inn": "",
      "account_number": "3023*****000",
      "bic": "",
      "bank_name": ""
    },
    "arn": "P1281100QAO",
    "auth_amount": "20500.00",
    "auth_currency": "RUR",
    "transaction_date": "08.05.2025 20:43:56",
    "rrn": "512880043563",
    "merchant": {
      "sic_name": "5631 Women's Accessory And Speci",
      "id": "240125298",
      "merchant_name": "Conte",
    },
    "terminal": {
      "id": "Y0070542",
      "location": "",
      "city": "",
      "country": ""
    },
    "authorization_code": "241013",
    "transaction_condition": "CAT PBT Chip Contactless",
    "cpid": "",
    "card_expire": "3205"
  }
],
"timestamp": "2025-10-28T13:00:02Z",
}

```

1.3 Пример манифеста

Topic: way4.cust.statements.dkkfm.result.manifest **Key:** 11111111-1111-1111-1111-111111111111 (request_id)

Headers:

- content-type: application/json
- x-request-id: 11111111-1111-1111-1111-111111111111
- x-response-id: 22222222-2222-2222-2222-222222222222
- x-sent-at: 2025-10-28T13:00:02Z

Payload (JSON):

```
{
  "msg": "result-manifest",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "response_id": "22222222-2222-2222-2222-222222222222",
  "total_rows": 2,
  "chunks": [
    { "idx": 1, "rows": { "start": 1, "end": 2 }, "bytes": 856 }
  ],
  "timestamp": "2025-10-31T13:00:02Z"
}
```

2. Запрос и ответ с разбиением на части(пример: 2 части)

2.1 Пример запроса по номеру счета

Topic: way4.cust.statements.dkkfm.request **Key:** 33333333-3333-3333-3333-333333333333 (request_id)

Headers:

- content-type: application/json
- x-request-id: 33333333-3333-3333-3333-333333333333
- x-sent-at: 2025-10-28T13:00:00Z

Payload (JSON):

```
{
  "msg": "request",
  "request_id": "11111111-1111-1111-1111-111111111111",
  "timestamp": "2025-10-28T13:00:00Z",
  "query": {
    "account_number": "40702*****5678",
    "date_from": "2025-08-01",
    "date_to": "2025-08-31"
  }
}
```

2.2 Ответ-часть 1/2

Topic: way4.cust.statements.dkkfm.result.chunk **Key:** 44444444-4444-4444-4444-444444444444 (response_id)

Headers:

- content-type: application/json
- x-request-id: 33333333-3333-3333-3333-333333333333
- x-response-id: 44444444-4444-4444-4444-444444444444

- x-chunk-idx: 1
- x-chunk-total: 2
- x-range-rows: 1-1500/2415
- x-sent-at: 2025-10-28T13:05:04Z

Payload (JSON):

```
{
  "msg": "result-chunk",
  "request_id": "33333333-3333-3333-3333-333333333333",
  "response_id": "44444444-4444-4444-4444-444444444444",
  "chunk": { "idx": 1, "total": 2, "rows": { "start": 1, "end": 1500 }, "bytes": 920384 },
  "total_rows": 2415,
  "header": {
    "bank_name": "Банк ВТБ (ПАО), филиал № 7701 в г. Москве",
    "account_number": "4081*****812",
    "date_from": "2025-05-14",
    "date_to": "2025-05-14",
    "client_name": "Копусов Владимир Михайлович",
    "inn": "",
    "birth_date": "1900-01-05",
    "client_number": "Test_client_20250508204335",
    "total_amount": "21500.00"
  },
  "transactions": [ /* 1500 compact-строк */ ],
  "timestamp": "2025-10-28T13:05:04Z"
}
```

2.3 Ответ-часть 2/2

Topic: way4.cust.statements.dkkfm.result.chunk **Key:** 44444444-4444-4444-4444-444444444444 (response_id)

Headers:

- content-type: application/json
- x-request-id: 33333333-3333-3333-3333-333333333333
- x-response-id: 44444444-4444-4444-4444-444444444444
- x-chunk-idx: 2
- x-chunk-total: 2
- x-range-rows: 1501-2415/2415
- x-sent-at: 2025-10-28T13:05:07Z
-

Payload (JSON):

```
{
  "msg": "result-chunk",
  "request_id": "33333333-3333-3333-3333-333333333333",
  "response_id": "44444444-4444-4444-4444-444444444444",
  "chunk": { "idx": 2, "total": 2, "rows": { "start": 1501, "end": 2415 }, "bytes": 612345 },
  "total_rows": 2415,
  "header": {
    "bank_name": "Банк ВТБ (ПАО), филиал № 7701 в г. Москве",
    "account_number": "4081*****812",
    "date_from": "2025-05-14",
    "date_to": "2025-05-14",
    "client_name": "Копусов Владимир Михайлович",
    "inn": "",
    "birth_date": "1900-01-05",
    "client_number": "Test_client_20250508204335",
    "total_amount": "21500.00"
  },
  "transactions": [ /* 915 compact-строк */ ],
  "timestamp": "2025-10-28T13:05:07Z"
}
```

2.4 Манифест

Topic: way4.cust.statements.dkkfm.result.manifest **Key:** 33333333-3333-3333-3333-333333333333 (request_id)

Headers:

- content-type: application/json
- x-request-id: 33333333-3333-3333-3333-333333333333
- x-response-id: 44444444-4444-4444-4444-444444444444
- x-sent-at: 2025-10-31T13:05:07Z
-

Payload (JSON):

```
{
  "msg": "result-manifest",
  "request_id": "33333333-3333-3333-3333-333333333333",
  "response_id": "44444444-4444-4444-4444-444444444444",
  "total_rows": 2415,
  "chunks": [
    { "idx": 1, "rows": { "start": 1, "end": 1500}, "bytes": 920384 },
    { "idx": 2, "rows": { "start": 1501, "end": 2415}, "bytes": 612345 }
  ],
  "timestamp": "2025-10-31T13:05:07Z"
}
```

3. Запрос → ошибка(например, не найден клиент или неверный формат даты)

3.1 Запрос

Topic: way4.cust.statements.dkkfm.request **Key:** 55555555-5555-5555-5555-555555555555 (req_id)

Payload (JSON):

```
{
  "msg": "request",
  "request_id": "55555555-5555-5555-5555-555555555555",
  "timestamp": "2025-10-28T13:10:00Z",
  "query": {
    "account_number": "40702*****5678",
    "date_from": "2032-08-01",
    "date_to": "2025-08-31"
  }
}
```

3.2 Сообщение об ошибке

Topic: way4.cust.statements.dkkfm.result.error

Key: 55555555-5555-5555-5555-555555555555 (request_id)

Headers:

- content-type: application/json
- x-request-id: 55555555-5555-5555-5555-555555555555
- x-sent-at: 2025-10-28T13:10:00Z

Payload (JSON):

```
{  
  "msg": "result-error",  
  "status": 400,  
  "request_id": "55555555-5555-5555-5555-555555555555",  
  "type": "account-not-found",  
  "detail": "Указанный в запросе счет не найден"  
  "timestamp": "2025-10-28T13:10:00Z"  
}
```