# [Стандарты разработки Datalake](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971)

* 1[Введение](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%92%D0%B2%D0%B5%D0%B4%D0%B5%D0%BD%D0%B8%D0%B5)
  + 1.1[Назначение документа](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%9D%D0%B0%D0%B7%D0%BD%D0%B0%D1%87%D0%B5%D0%BD%D0%B8%D0%B5%D0%B4%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0)
  + 1.2[Глосарий документа](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%93%D0%BB%D0%BE%D1%81%D0%B0%D1%80%D0%B8%D0%B9%D0%B4%D0%BE%D0%BA%D1%83%D0%BC%D0%B5%D0%BD%D1%82%D0%B0)
* 2[Стандарты разработки](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8)
  + 2.1[Стандарты разработки Oozie](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Oozie)
    - 2.1.1[Правила наименования oozie workflow](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%9F%D1%80%D0%B0%D0%B2%D0%B8%D0%BB%D0%B0%D0%BD%D0%B0%D0%B8%D0%BC%D0%B5%D0%BD%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8Foozieworkflow)
    - 2.1.2[Атомарный поток (atom)](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%90%D1%82%D0%BE%D0%BC%D0%B0%D1%80%D0%BD%D1%8B%D0%B9%D0%BF%D0%BE%D1%82%D0%BE%D0%BA(atom))
  + 2.2[Стандарты разработки Sqoop](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Sqoop)
  + 2.3[Стандарты разработки Hive](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Hive)
    - 2.3.1[Общие требования](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%9E%D0%B1%D1%89%D0%B8%D0%B5%D1%82%D1%80%D0%B5%D0%B1%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)
    - 2.3.2[Среды развертывания](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%80%D0%B5%D0%B4%D1%8B%D1%80%D0%B0%D0%B7%D0%B2%D0%B5%D1%80%D1%82%D1%8B%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F)
    - 2.3.3[Стандарты именования объектов Hive, загружаемых 1-в-1](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D0%B8%D0%BC%D0%B5%D0%BD%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F%D0%BE%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BE%D0%B2Hive,%D0%B7%D0%B0%D0%B3%D1%80%D1%83%D0%B6%D0%B0%D0%B5%D0%BC%D1%8B%D1%851-%D0%B2-1)
    - 2.3.4[Стандарты именования объектов Hive , формируемых на основе таблиц Hadoop](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D0%B8%D0%BC%D0%B5%D0%BD%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D1%8F%D0%BE%D0%B1%D1%8A%D0%B5%D0%BA%D1%82%D0%BE%D0%B2Hive,%D1%84%D0%BE%D1%80%D0%BC%D0%B8%D1%80%D1%83%D0%B5%D0%BC%D1%8B%D1%85%D0%BD%D0%B0%D0%BE%D1%81%D0%BD%D0%BE%D0%B2%D0%B5%D1%82%D0%B0%D0%B1%D0%BB%D0%B8%D1%86Hadoop)
    - 2.3.5[Структура репозитория для Hive](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D1%80%D1%83%D0%BA%D1%82%D1%83%D1%80%D0%B0%D1%80%D0%B5%D0%BF%D0%BE%D0%B7%D0%B8%D1%82%D0%BE%D1%80%D0%B8%D1%8F%D0%B4%D0%BB%D1%8FHive)
    - 2.3.6[Подготовка скриптов установки](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%9F%D0%BE%D0%B4%D0%B3%D0%BE%D1%82%D0%BE%D0%B2%D0%BA%D0%B0%D1%81%D0%BA%D1%80%D0%B8%D0%BF%D1%82%D0%BE%D0%B2%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B8)
  + 2.4[Стандарты разработки pyspark](http://confluence.moscow.alfaintra.net/pages/viewpage.action?pageId=291414971#id-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8Datalake-%D0%A1%D1%82%D0%B0%D0%BD%D0%B4%D0%B0%D1%80%D1%82%D1%8B%D1%80%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D0%BA%D0%B8pyspark)

Введение

Назначение документа

Данный документ предназначен для определения стандартов разработки Hadoop Datalake и описывает:

* Правила наименования объектов Hadoop Datalake
* Структуру репозитория git
* Подготовку скриптов установки патчей
* Слои данных
* Стратегии загрузки данных

Глосарий документа

| **Термин** | **Определение** |
| --- | --- |
|  |  |
|  |  |
|  |  |

Стандарты разработки

Стандарты разработки Oozie

Для регулярного запуска любых ETL процедур в Datalake, используется Apache Oozie. Из абстракций Oozie используются workflow (wf) и coordinator (crd).

ETL процедуры организованы как отдельные компоненты (oozie workflow) и организованы в иерархию из трёх уровней вложенности:

1. Атомарные потоки (atom)
2. Контролирующие потоки (ctl)
3. Регулярные потоки (reg)

wf\_ctl\_dwh\_  
daily\_facts

|  |
| --- |
| wf\_ctl\_ dmacrm\_ monthly\_ day4\_facts |
| wf\_ctl\_ vertica\_daily | |

|  |
| --- |
| wf\_ctl\_dwh\_ daily\_dictionaries |
| wf\_dmacrm\_dealloanlimit\_m\_sagg | |

|  |
| --- |
| wf\_dwh\_dealrate\_vhist |
| wf\_dwh\_deal2acct\_vhist | |

|  |
| --- |
| wf\_dwh\_dealloanaction\_htran |
| wf\_dwh\_sparkfinreport\_hstat |

|  |  |
| --- | --- |
| wf\_dwh\_sparkgovernance\_hstat | |
| wf\_dwh\_sparkhead\_hstat |

|  |
| --- |
| wf\_dwh\_balance\_hstat |
| wf\_dwh\_client2division\_vhist | |

|  |  |
| --- | --- |
| wf\_dmacrm\_dealloanutility\_m\_sagg | |
| wf\_vertica\_intra\_event\_log |

|  |
| --- |
| wf\_dwh\_product\_hdim |
| wf\_dwh\_salesplace\_hdim | |

|  |
| --- |
| wf\_dwh\_staffing\_hdim |
| wf\_dwh\_clientsegment\_hdim | |

|  |
| --- |
| wf\_dwh\_voc\_sdim |
| wf\_dwh\_claim\_hdim | |

**wf\_reg\_dwh\_daily**

|  |
| --- |
| wf\_reg\_ssvd\_ monthly |
| wf\_reg\_ vertica\_daily | |

**Правила наименования oozie workflow**

Все oozie workflow должны наименоваться в нижнем регистре и соответствовать шаблонам наименования.

| **Тип объекта** | **Шаблон наименования** |
| --- | --- |
| REG | wf\_reg\_<SOURCE>\_<PERIODICITY> |
| CTL | wf\_ctl\_<SOURCE>\_<PERIODICITY>\_<OBJ\_TYPE> |
| ATOM | wf\_<OBJECT\_NAME> |

, где <SOURCE> - это мнемонический код системы-источника  
, <PERIODICITY> - тип регламента, например: daily, weekly, monthly  
, <OBJ\_TYPE> - тип объекта, например: dictionaries или facts  
, <OBJECT\_NAME> - наименование объекта системы источника

**Атомарный поток (atom)**

atom - минимальная единица работы. В одном atom может загружаться только одна таблица.  
atom состоит из нескольких шагов:

* подготовка (header)
* извлечение данных из системы-источника во временную область HDFS (sqoop)
* загрузка данных в регулярную таблицу hive (hive)
* завершение (tail)

Header

Sqoop

Hive

Tail

**Atom**

Java-код для чтения метаданных из бд и параметризации workflow

Выгрузка данных из реляционной системы-источника во временную область hdfs

Скрипты переливки данных из временной области hdfs в постоянную область hive

Java-код для записи метаданных в бд

Подготовка (header) - это технический шаг, реализованный в виде java-action, предназначенный для интеграции в atom функций управляющего механизма, например считывание метаданных УМа для динамической параметризации.

**atom-header**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <action name="atom-header-claim\_hdim">      <java>          <job-tracker>${jobTracker}</job-tracker>          <name-node>${nameNode}</name-node>          <main-class>ru.alfa.bi.hdp.orchestration.AtomHeader</main-class>          <java-opts>-DATOM\_NAME=wf\_dwh\_claim\_hdim -DCTL=${wf\_ctl\_name} -DREG=${wf\_reg\_name} -DDB\_CONNECTION=${oracle\_jdbc\_url} -DDB\_USER=${oracle\_jdbc\_user} -DDB\_PASSWORD=${oracle\_jdbc\_password} -DCURRENT\_DAY=${current\_day}</java-opts>          <file>/user/${user\_name}/libs/generator-1.0-SNAPSHOT-jar-with-dependencies.jar#generator-1.0-SNAPSHOT-jar-with-dependencies.jar</file>          <file>/user/${user\_name}/libs/ojdbc6.jar#ojdbc6.jar</file>      </java>      <ok to="sqoop-claim\_hdim"/>      <error to="Kill"/>  </action> |

Извлечение данных как правило реализуется средствами Apache Sqoop и описывается в workflow.xml как sqoop-action. Данные выгружаются в HDFS в директорию для соответствующей временной таблицы (external table) в формате parquet.

**sqoop-action**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31 | <action name="sqoop-claim\_hdim">      <sqoop xmlns="uri:oozie:sqoop-action:0.2">          <job-tracker>${jobTracker}</job-tracker>          <name-node>${nameNode}</name-node>          <arg>import</arg>          <arg>-D</arg>          <arg>mapreduce.map.memory.mb=4096</arg>          <arg>-D</arg>          <arg>mapreduce.map.java.opts=-Xmx2048m</arg>          <arg>--connect</arg>          <arg>${connection\_string}</arg>          <arg>--username</arg>          <arg>${ORA\_USER}</arg>          <arg>--password</arg>          <arg>${ORA\_PASSWORD}</arg>          <arg>--compression-codec</arg>          <arg>org.apache.hadoop.io.compress.SnappyCodec</arg>          <arg>--as-parquetfile</arg>          <arg>--delete-target-dir</arg>          <arg>--target-dir</arg>          <arg>${PATH\_TO\_TMP}/claim\_hdim\_tmp</arg>          <arg>--query</arg>          <arg>select UK, NCODE, CLAIMREASON\_GK, CLIENTGROUP\_UK, CURRENCY\_UK, SALESPLACE\_UK, CLIENT\_UK, CLIENTCONTACT\_ADDPHONE\_UK, CLIENTCONTACT\_PHONE\_UK, CLAIMTYPE\_UK, AUTHOR\_NAME, to\_char(REGISTRATION\_DATE, 'yyyy-MM-dd HH24:MI:SS') as REGISTRATION\_DATE, CLIENT\_FLAG, CLAIM\_CUR\_AMT, DESCRIPTION, CONCLUSION\_DESCRIPTION, REPCLAIM\_FLAG, PAPERCLAIM\_FLAG, to\_char(START\_DATE, 'yyyy-MM-dd HH24:MI:SS') as START\_DATE, to\_char(END\_DATE, 'yyyy-MM-dd HH24:MI:SS') as END\_DATE, to\_char(AS\_OF\_DAY, 'yyyy-MM-dd HH24:MI:SS') as AS\_OF\_DAY, DEFAULT\_FLAG, DELETED\_FLAG, DWSCMIX, JOB\_INSERT, JOB\_UPDATE, to\_char(VALIDFROM, 'yyyy-MM-dd') as VALIDFROM, to\_char(VALIDTO, 'yyyy-MM-dd') as VALIDTO, MODULE\_UK, CLAIMTHEME\_UK, PUBLIC\_CLAIM\_REF, to\_char(SLA\_END\_DATE, 'yyyy-MM-dd') as SLA\_END\_DATE, CLAIMREASON\_UK, to\_char(RESOLUTION\_DATE, 'yyyy-MM-dd') as RESOLUTION\_DATE, CHANNEL\_DETAIL\_UK, CLAIM\_REF, USER\_UK, CHANNEL\_UK, ANALYSISGRP\_UK, CLAIMPRODUCT\_UK, PARENT\_UK, MAINBANK\_FLAG, CLIENTSEGMENT\_UK, CLAIMCONSIDERRULE\_UK, USER\_BLAME\_UK, SALESPLACE\_BLAME\_UK from DWH.CLAIM\_HDIM where $CONDITIONS</arg>          <arg>-m</arg>          <arg>1</arg>          <arg>--map-column-java</arg>          <arg>UK=Double,NCODE=Double,CLAIMREASON\_GK=Double,CLIENTGROUP\_UK=Double,CURRENCY\_UK=Double,SALESPLACE\_UK=Double,CLIENT\_UK=Double,CLIENTCONTACT\_ADDPHONE\_UK=Double,CLIENTCONTACT\_PHONE\_UK=Double,CLAIMTYPE\_UK=Double,CLAIM\_CUR\_AMT=Double,DWSCMIX=Double,JOB\_INSERT=Double,JOB\_UPDATE=Double,MODULE\_UK=Double,CLAIMTHEME\_UK=Double,CLAIMREASON\_UK=Double,CHANNEL\_DETAIL\_UK=Double,USER\_UK=Double,CHANNEL\_UK=Double,ANALYSISGRP\_UK=Double,CLAIMPRODUCT\_UK=Double,PARENT\_UK=Double,CLIENTSEGMENT\_UK=Double,CLAIMCONSIDERRULE\_UK=Double,USER\_BLAME\_UK=Double,SALESPLACE\_BLAME\_UK=Double</arg>      </sqoop>      <ok to="hive2-claim\_hdim"/>      <error to="Kill"/>  </action> |

Загрузка данных из временной таблицы в постоянную реализуется в виде hive скрипта и описывается в workflow.xml как hive-action.

**hive-action**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14 | <action name="hive2-claim\_hdim" cred="hive2">      <hive2 xmlns="uri:oozie:hive2-action:0.1">          <job-tracker>${jobTracker}</job-tracker>          <name-node>${nameNode}</name-node>          <jdbc-url>${hive2\_jdbc\_url}</jdbc-url>          <script>/user/${user\_name}/scripts/load\_data\_inpath\_no\_part.hql</script>              <param>DATABASE=${DATABASE}</param>              <param>TABLE\_NAME=claim\_hdim</param>              <param>PATH\_TO\_TMP=${PATH\_TO\_TMP}</param>              <param>TMP\_TABLE\_TYPE=${TMP\_TABLE\_TYPE}</param>      </hive2>      <ok to="atom-tail-claim\_hdim"/>      <error to="Kill"/>  </action> |

В hive-action описываются параметры, сам скрипт находится в hdfs в виде тестового hql-файла.

**hql-script**

|  |  |
| --- | --- |
| 1  2  3 | LOAD DATA INPATH '${PATH\_TO\_TMP}/${TABLE\_NAME}\_${TMP\_TABLE\_TYPE}' OVERWRITE INTO TABLE ${DATABASE}.${TABLE\_NAME};    analyze table ${DATABASE}.${TABLE\_NAME} compute statistics; |

Завершение (tail) - это технический шаг, реализованный в виде java-action, предназначенный для интеграции в atom функций управляющего механизма, например логирования и обновления метаданных УМ.

**atom-tail**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12 | <action name="atom-tail-claim\_hdim">      <java>          <job-tracker>${jobTracker}</job-tracker>          <name-node>${nameNode}</name-node>          <main-class>ru.alfa.bi.hdp.orchestration.AtomTail</main-class>          <java-opts>-DATOM\_NAME=wf\_dwh\_claim\_hdim -DDB\_CONNECTION=${oracle\_jdbc\_url} -DDB\_USER=${oracle\_jdbc\_user} -DDB\_PASSWORD=${oracle\_jdbc\_password} -DCURRENT\_DAY=${current\_day}</java-opts>          <file>/user/${user\_name}/libs/generator-1.0-SNAPSHOT-jar-with-dependencies.jar#generator-1.0-SNAPSHOT-jar-with-dependencies.jar</file>          <file>/user/${user\_name}/libs/ojdbc6.jar#ojdbc6.jar</file>      </java>      <ok to="End"/>      <error to="Kill"/>  </action> |

Стандарты разработки Sqoop

На прямую утилиты sqoop не используется, вместо этого она описывается в рамках sqoop-action в oozie workflow.

TBD: описать допустимые типы данных, как использовать параллельность.

Стандарты разработки Hive

**Общие требования**

Кодировка всех исходных файлов должна быть UTF8, длина строки не превышать 120 символов.

Код создания объектов должен храниться в отдельных файлах, имя файла должно быть в формате <имя\_объекта>.hql.

Код содержащий трансформации (копирование/слияние) должен храниться в отдельных файлах, одна логическая трансформация должна быть в рамках одного файла. Если для одной и той же целевой таблицы используется несколько трансформаций, то после имени должен быть добавлен суффикс, логически отражающий суть. Если одна трансформация заполняет несколько таблиц, то имя скрипта должно отражать бизнес процесс.

В качестве формата хранения таблиц обязательно использовать parquet с компрессией snappy.

**Среды развертывания**

В процессе разработки используются следюущие среды:

* dev
* test
* prod

Разделение сред реализуется на уровне hdfs.

Для среды dev используется директория /dev и префикс d\_ для различных областей данных.

**hdfs-dev**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | [u\_m0znj@bda31node04 ~]$ hadoop fs -ls /dev  Found 28 items  drwxrwx---+  - hdfs   d\_deriveddata\_w              0 2019-02-27 11:25 /dev/d\_deriveddata  drwxrwx---+  - hdfs   d\_deriveddata\_tmp\_w          0 2019-02-27 11:25 /dev/d\_deriveddata\_tmp  drwxrwx---+  - hdfs   d\_dmacrm\_w                   0 2019-02-27 11:26 /dev/d\_dmacrm  drwxrwx---+  - hdfs   d\_dmacrm\_tmp\_w               0 2019-02-27 11:46 /dev/d\_dmacrm\_tmp  ...  drwxrwx---+  - hdfs   d\_odh\_w                      0 2019-02-27 11:39 /dev/d\_odh  drwxrwx---+  - hdfs   d\_odh\_tmp\_w                  0 2019-02-27 11:39 /dev/d\_odh\_tmp  drwxrwx---+  - hdfs   d\_sourcedata\_w               0 2019-02-27 11:39 /dev/d\_sourcedata |

Для среды test используется директория /test и префикс t\_ для различных областей данных.

**hdfs-test**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | [u\_m0znj@bda31node04 ~]$ hadoop fs -ls /test  Found 31 items  drwxrwx---+  - hdfs  t\_deriveddata\_w              0 2019-02-26 14:36 /test/t\_deriveddata  drwxrwx---+  - hdfs  t\_deriveddata\_tmp\_w          0 2019-02-26 14:36 /test/t\_deriveddata\_tmp  drwxrwx---+  - hdfs  t\_dmacrm\_w                   0 2019-02-26 14:36 /test/t\_dmacrm  drwxrwx---+  - hdfs  t\_dmacrm\_tmp\_w               0 2019-02-26 14:36 /test/t\_dmacrm\_tmp  ...  drwxrwx---+  - hdfs  t\_odh\_w                      0 2019-04-01 09:35 /test/t\_odh  drwxrwx---+  - hdfs  t\_odh\_tmp\_w                  0 2019-02-26 14:36 /test/t\_odh\_tmp  drwxrwx---+  - hdfs  t\_sourcedata\_w               0 2019-02-26 14:36 /test/t\_sourcedata |

Для среды prod используется директория /storage и префикс s\_ для различных областей данных

**hdfs-prod**

|  |  |
| --- | --- |
| 1  2  3  4  5  6 | [u\_m0znj@bda31node04 ~]$ hadoop fs -ls /storage  Found 26 items  drwxrwx---+ - hdfs s\_dmacrm\_w 0 2018-07-10 16:59 /storage/s\_dmacrm  drwxrwx---+ - hdfs s\_dmacrm\_tmp\_w 0 2018-07-11 11:44 /storage/s\_dmacrm\_tmp  ...  drwxrwx---+ - s\_odh s\_odh\_w 0 2018-06-05 19:14 /storage/s\_odhdrwxrwxr-x+ - s\_sourcedata s\_sourcedata\_w 0 2018-04-27 17:17 /storage/s\_sourcedata |

**Стандарты именования объектов Hive, загружаемых 1-в-1**

Имя объекта должно соответствовать имени источника, в других случаях выражать основную цель его существования.

Таблицы из области репликации DWH и витрин должны именоваться аналогично источнику.

Таблицы из иных реляционных систем источников должны именоваться по правилу: <имя\_системы\_источника>\_<имя\_таблицы><\_тип\_таблицы> ,где  
<имя\_системы\_источника> - это код системы-источника, например: ssvd\_edftsr, rbprod\_azheleznyakov, ankstandby и т.п.  
<имя\_таблицы> - имя таблицы системы источника, например: application\_pilq, ftclient, all\_inflow и т.п.  
<тип\_таблицы> - постоянная или временная, например в случае временной это \_tmp, в случае постоянной суффикса нет.

**Стандарты именования объектов Hive , формируемых на основе таблиц Hadoop**

При оптимизации витрин при создании временной таблицы витрины предварительно могут создаваться промежуточные временные таблицы.

Правило наименования этих промежуточных временных таблиц витрин:

**<MART\_TMP\_DATABASE>.tmp\_<mart\_table\_name>\_<table\_name>**

, где <MART\_TMP\_DATABASE>= l\_deriveddata\_tmp

<mart\_table\_name> - название таблицы витрины

<table\_name> - название промежуточной временной таблицы

Пример:

create table **${MART\_TMP\_DATABASE}.tmp\_epay\_operation\_account\_debet**  
stored as parquet  
as  
select   
    accb.start\_date,  
[accb\_t.name](http://accb_t.name/),  
[accb.uk](http://accb.uk/),  
    accb.client\_pin,  
    accb.client\_uk   
from ${environment}\_DMRB.ACCOUNT\_SDIM accb  
LEFT JOIN ${environment}\_DMRB.ACCOUNTEQTYPE\_LDIM accb\_t  
    ON accb.ACCOUNTEQTYPE\_UK = [accb\_t.uk](http://accb_t.uk/);

Соответствие:

**<MART\_TMP\_DATABASE>.tmp\_<mart\_table\_name>\_<table\_name>**

**${MART\_TMP\_DATABASE}.tmp\_<epay\_operation>\_<account\_debet>**

**Структура репозитория для Hive**

|–**hive\_scripts**  
    |–**s\_dwh@bda11**        |–address\_hdim.hql  
        |–sparkcompany\_sdim.hql  
        |–...  
    |–**s\_dwh\_tmp@bda11**  
        |–address\_hdim\_tmp.hql  
        |–sparkcompany\_sdim\_tmp.hql  
        |–...  
|–run\_hive\_scripts.sh

Где **синим** обозначены директории, зеленым - исполняемые-файлы, остальные файлы - черным.

**Подготовка скриптов установки**