Расчёт метрик ликвидности облигаций Тут описано, как оно работает на домашней машине и/или как будет работать на работе на сервере (без кластеризации) в виде прототипа. Как оно будет работать в итоге и в идеале - неведомо пока что. MySql cluster Чёрным отмечено то, что уже сделано. Синим - что надо бы сделать скорее. Оранжевым - что сделать тоже надо, и тоже скорее Kafka Connect Kafka topic Boo eurobonds-quotes-topic 2.2.1. **Transporter module** SPARK 2.2.3. 1.2.2. (AKKA) 2.1.4 В РСУБД WS. В РСУБД cbonds.info 1.2.1. https json Интернет (можно 2.1.5. allow 2.1. SPARK Batch 1.2. Transporter 2.2. SPARK Streaming Job: оставить, будет tcp-петля, DataFrame Job: signal + Job: Enrich bonds но можно и доработать) DataFrames stream intraday Оркестратор 1.1.3. liquidity metrics Архивы котировок delayed quotes (cron; сейчас 1.1.1. https 1.1. Transporter allow with intraday delayed трансформируем в руками) signal bonds quotes Job: метрики ликвидности по database. (trading volume) stage csv files to fs zip списку бумаг cbonds.info TODO: разделить на самостоятельные TODO: job-ы; разделить на модуляризировать самостоятельные 2.1.1. csv files код 1.1.2. extract 2.1.2. transformations 2.2.2. transformations - список бондов job-ы zip to csv files (local mode local[2]) - архив биржевых (local mode local[2]) котировок за месяц - архив внебиржевых за месяц FS (сейчас общий 2.1.3. parquet (для архивного контроля) volume докеров для архивного контроля и csv (для текущей наглядности) "shara") остаются последние за день (TODO: архив - data vault fили что-то в этом роде) человеки