# Overview

Jigsaw are looking to develop a nascent capability within Data Science. It is the intent to continue to grow this discipline, enabling data analytics to become a core feature within Jigsaw products and the services which we provide to clients.

The first step in this journey is seen to be the delivery of an initial data analytics problem; design and development of analytics functions and the configuration of the target execution environment to support deployment / execution of these functions.

Aspirational evolution of a Jigsaw Data Science capability would involve increasing our own maturity to the point whereby we could lesson dependencies upon freelance / consulting domain expertise to repeat the mechanics of solving data exercises within our own workforce.

# Initial Data Analytics Problem

The problem trying to be solved is summarised as follows:

`Given transactional history for a SME and ingest of current Accounting data (Invoices / Liabilities / Expenses / Overheads), to what accuracy can a CashFlow Forecast be generated’.

Transaction history will provide a categorised feed for all credits / debits against an SME Business Current Account [BCA]. Additionally, information will be available around any Scheduled Payments (Direct Debits / Standing Orders) that exist against the BCA which will provide an indicator around future events.

Accounting Invoices / Liabilities journal indicators will provide an indicator around future events.

## Approach to Date

Given that the problem [as described] relates to a sharded dataset, hypotheses have been explored through MicroService development. The current evolution of the model is as follows:

* As Accounting journal data is ingested, value date aggregates are maintained for postings against journal sub-categories
* Linear trend[[1]](#footnote-1) [3 month timeseries] for aggregate sub-category transactions value is re-calculated for revised aggregate value. Assumption is that linear plots will demonstrate seasonal variation and therefore, a quarterly focus is taken
* Forward forecast [3 month timeseries] for sub-category transactions plotted based upon revised linear curve
* Dedupe assessed to eliminate potential double counting around Scheduled Payment / Liability & Invoice.

Areas under current further investigation:

* Identification and modelling of low frequency transactions (e.g. yearly)
* Seasonal pattern overlay
* Non-linear interpolation (cubic spline) within the forward forecast period
* Closed loop feedback through measurement of Actual curve upon period close in order to assess accuracy of Forecast curve at key intra-period milestones (e.g. monthly)
* Correlation of actual invoice payment and invoice data to more accurately reflect micro-cashflow.

## Impact on forecast accuracy due in the event of low data

The current analytics approach is limited to the ideal scenario where there is sufficient data. In the event of historical transaction data depth being non-existent / shallow (aka “thin book”), then there is the aspiration to be able to derive suitable correlation factors against other [deeper / more mature] data profiles in order that a CashFlow forecast can still be generated.

It is assumed that correlation around SME sector / location/ revenue / profitability will enable approximations to be made around Direct Costs (although perhaps not around Income).

Uncertainties:

1. Optimal approach is to continue re-evaluate the entire dataset in order to establish / re-evaluate correlation ? Should a more structure demographic model be employed where metrics are maintained for a demographic category. SME association with the demographic category would be subject to ongoing re-evaluation ? I lean more towards the former in my thinking
2. Execution event. Assume that full data surface analysis is triggered on a periodical basis and would not necessarily have a specific [high frequency] event that would initiate execution ?

# Target Execution Environment

Based upon review of widely adopted patterns, an Apache ecosystem is the analytics execution environment that Jigsaw are focussing upon. The Azure Public cloud is being used for the purpose of this initiative. The Azure HDInsight PAAS Catalogue item is being subscribed in order to provision the Apache ecosystem.

The current Azure execution stack is provisioned for each Jigsaw developer:



Jigsaw’s primary development capability is around MicroServices (running within the Azure Kubernetes Service). Development of a nascent Data Science capability will start to leverage the Azure HDInsights component of the Developer stack.

HDInsights provides a number of managed Apache services. The current approach to ingest / execution and resultant processing leverages the components shown below:



The processing event is considered to be the ingest of data across a Kafka ETL Stream. The Kafka Stream writes to a HIVE External table and triggers the Spark Engine across the Livy interface.

HIVE Internal Tables are being used [low cost] persistence for historical data (and therefore, should be considered to be an Analytics Warehouse).

Spark execution will be dependent upon historical data from the HIVE Analytics Warehouse, as well as the ingest data persisted to the HIVE External table.

Spark output would be written to both HIVE Internal table and [some] of the output would be placed on a Kafka stream, for persistence into the developer’s ODS instance. **Note:** This element of the proposed pipeline has not been Spiked, therefore, implementation guidance would be sought as part of this exercise.

# Objective Outcomes

To be further defined as part of our exploratory conversations:

* Suggestions around a workable approach to sourcing / generating synthetic data in order to support development
* Development of sharded analytics (to be discussed how many hypotheses to be explored)
* Development of full dataset analytics
* Understanding of how to scope / build / run tests within a CI/CD pipeline around both sharded & full dataset analytics (specific concern is time taken to run tests i.e. *should* I be considering breaking the several minute CI/CD pipeline targets that we currently engineer to ?
* Understanding of how to schedule / execute Spark jobs that are dependent upon the full data set.

1. We haven’t tried interpolating non-linear curves around the historic data sets to date [↑](#footnote-ref-1)