



Hadoop at SciLifeLab

Gautier Berthou
SeRC Data Management Community



4 Vs of Big Data

- Volume
- Velocity
- Variety
- Value

Next Generation Sequencing at SciLifeLab

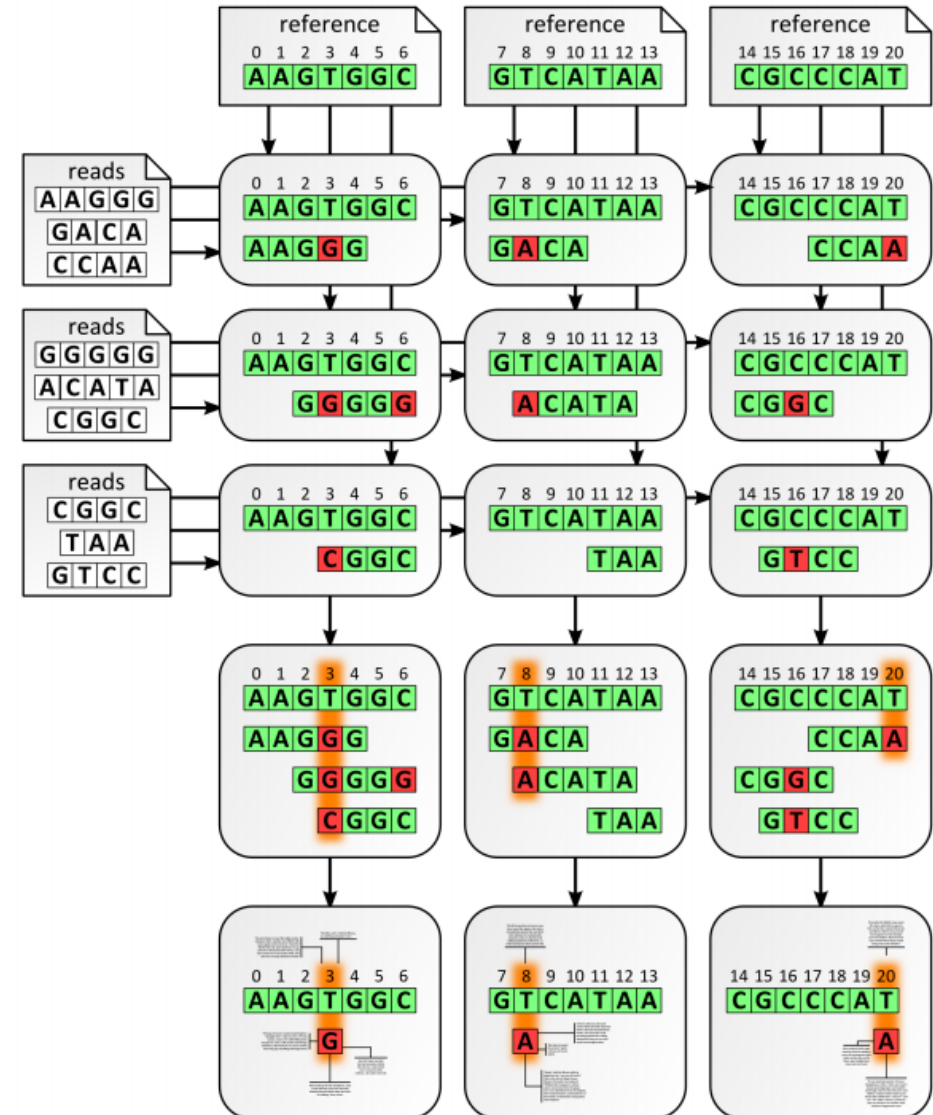
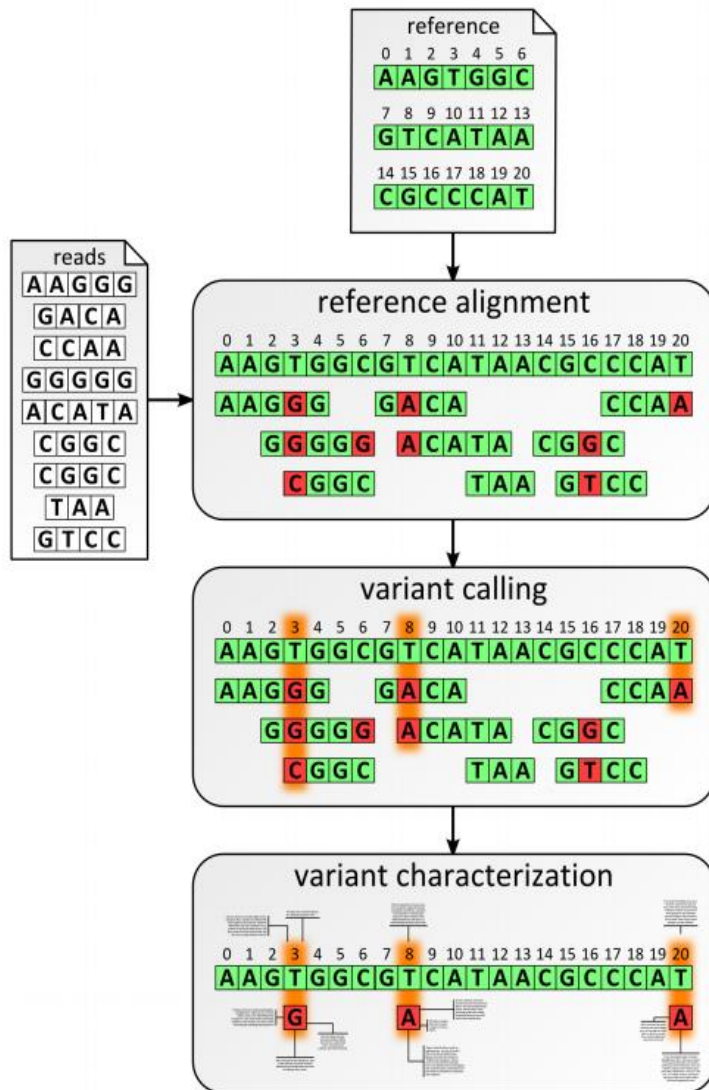


Volume	=>	~5.2 PB/year
Velocity	=>	~45 MB/sec
Variety	=>	Low
Value	=>	??

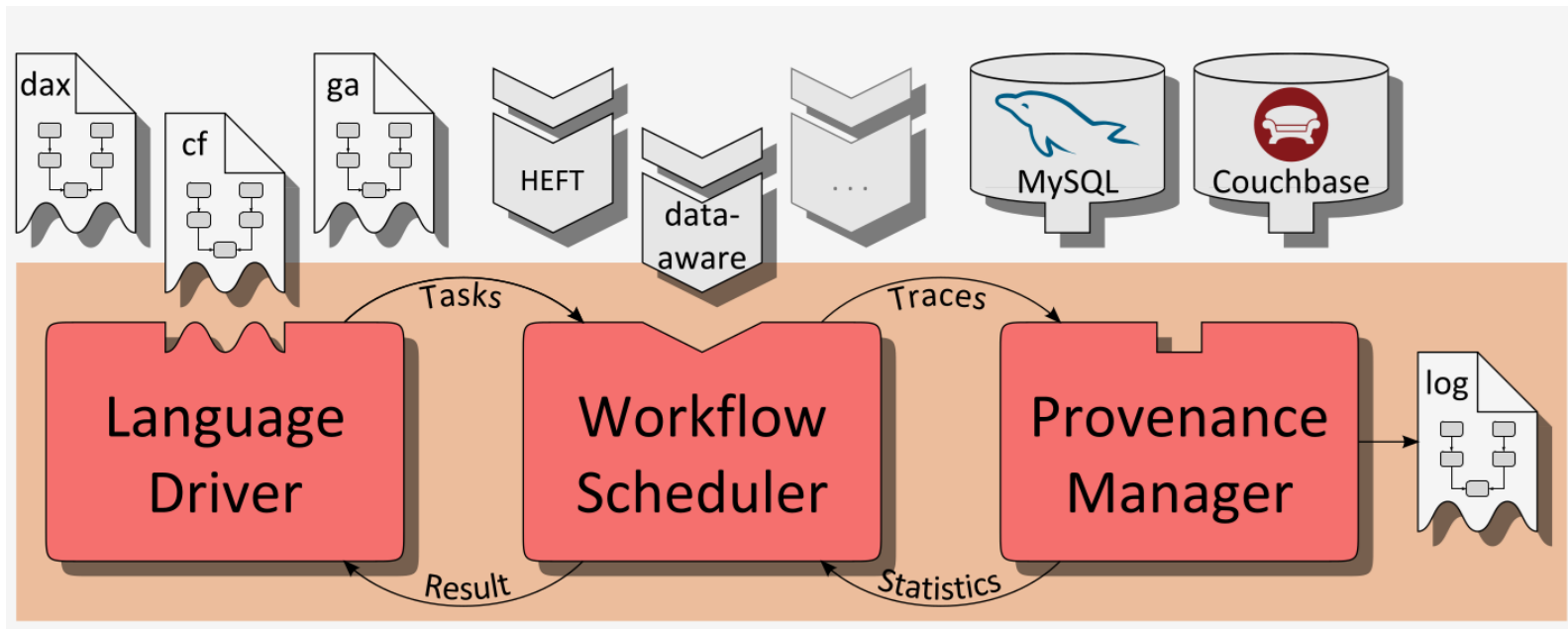
What do you do with NGS Data?

1. Process the raw data (run some workflow)
2. Store
3. Share
4. Analyze/visualize

Variant calling



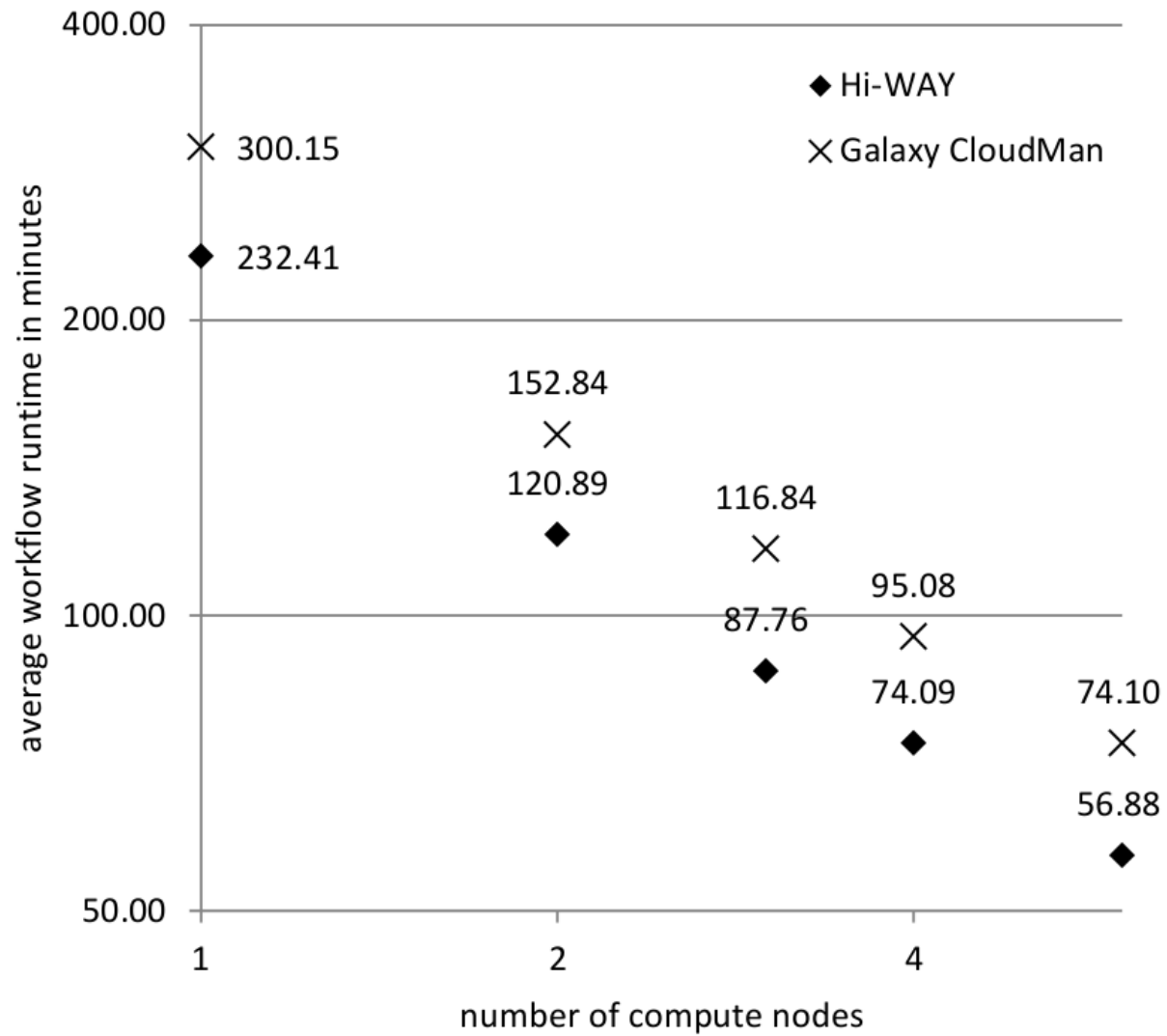
Hi-WAY



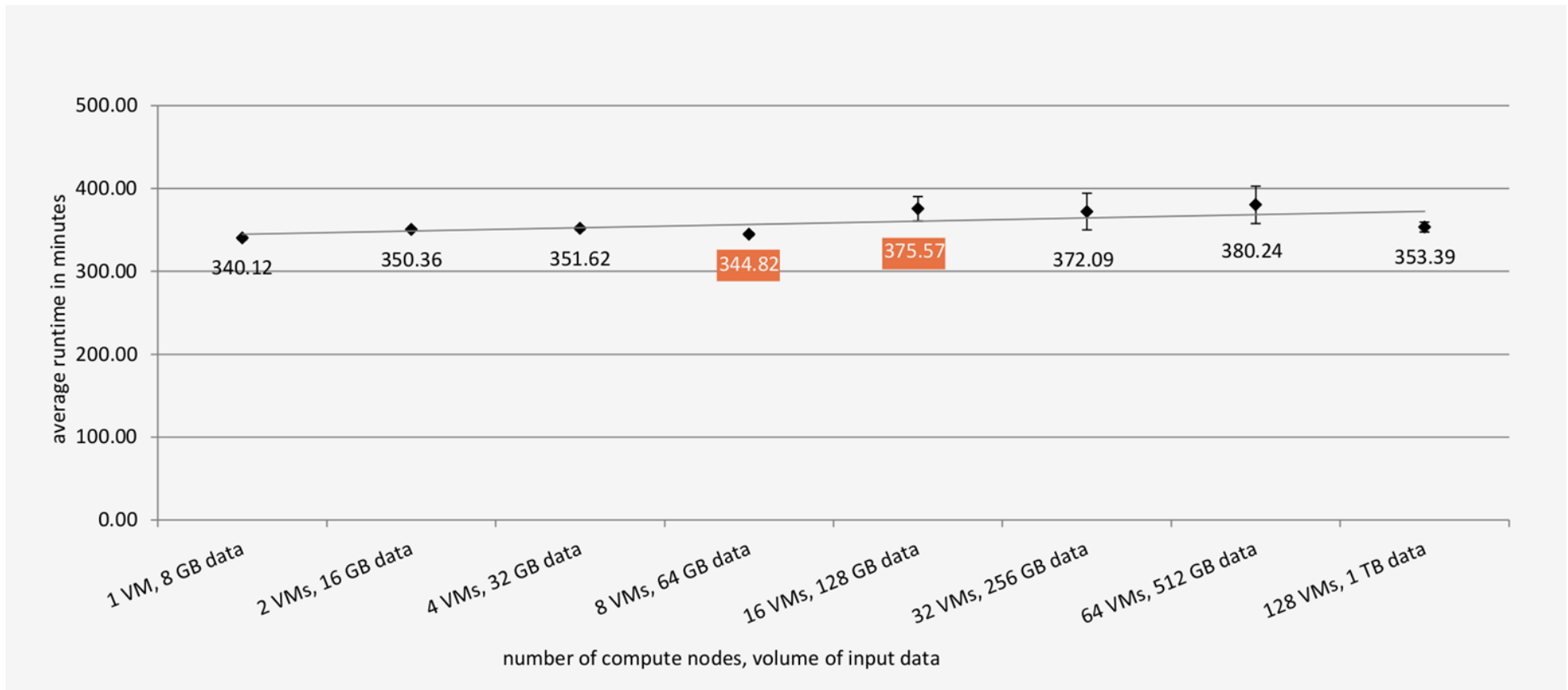
- Multilingual workflow system
- Scalable and efficient
- Adaptative scheduling policies
- Reproducibility with Karamel

Marc Bux, Jörgen Brandt, Carsten Lipka, Kamal Hakimzadeh, Jim Dowling, and Ulf Leser. Saasfee: Scalable scientific workflow execution engine. In VLDB Demonstrations Track, forthcoming, Hawaii, USA, September 2015.


Hi-WAY vs Galaxy



Hi-WAY



Hops.site

HopsWorks Beta 

SIGN IN TO CONTINUE.

Email

Password

Mobile PIN or Yubikey PIN

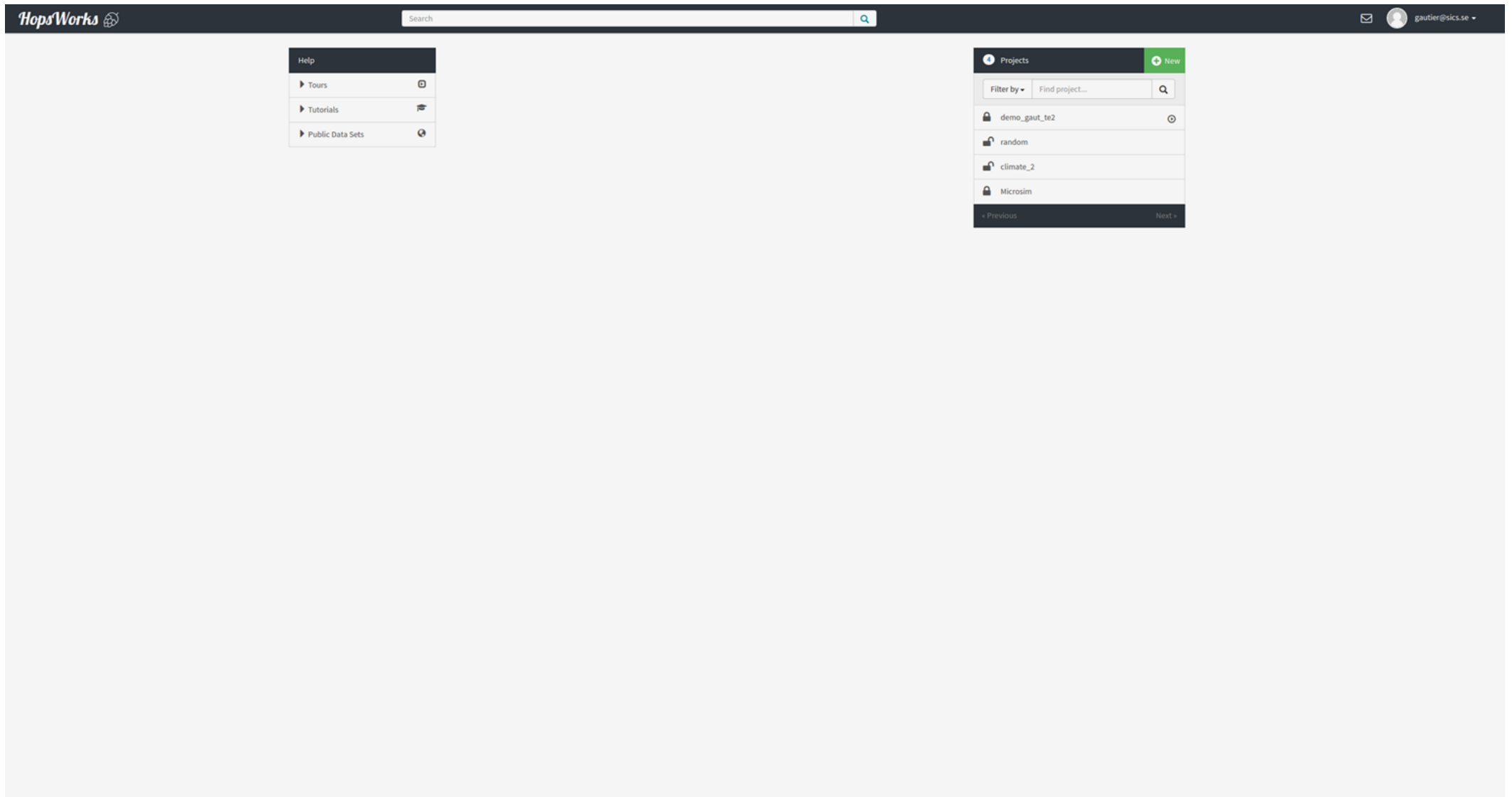
Need support?

Login help?

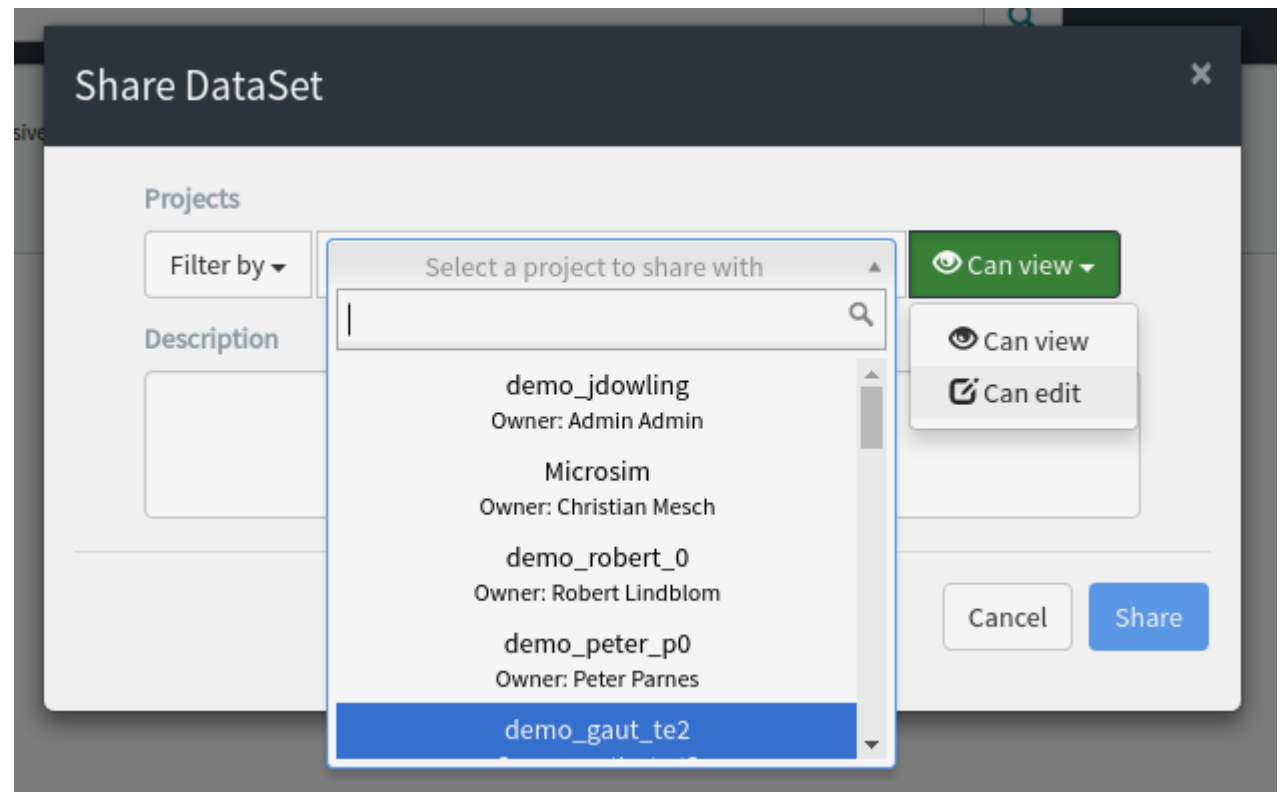
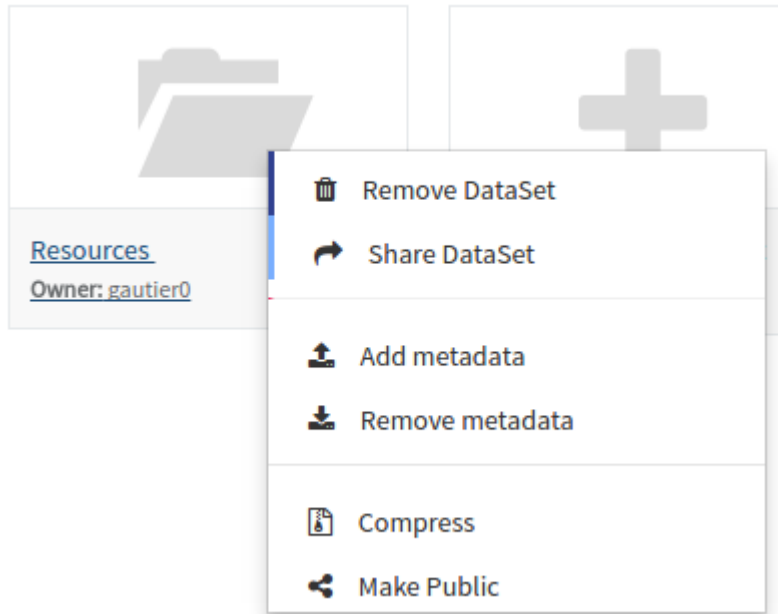
Login

Register

Hops.site



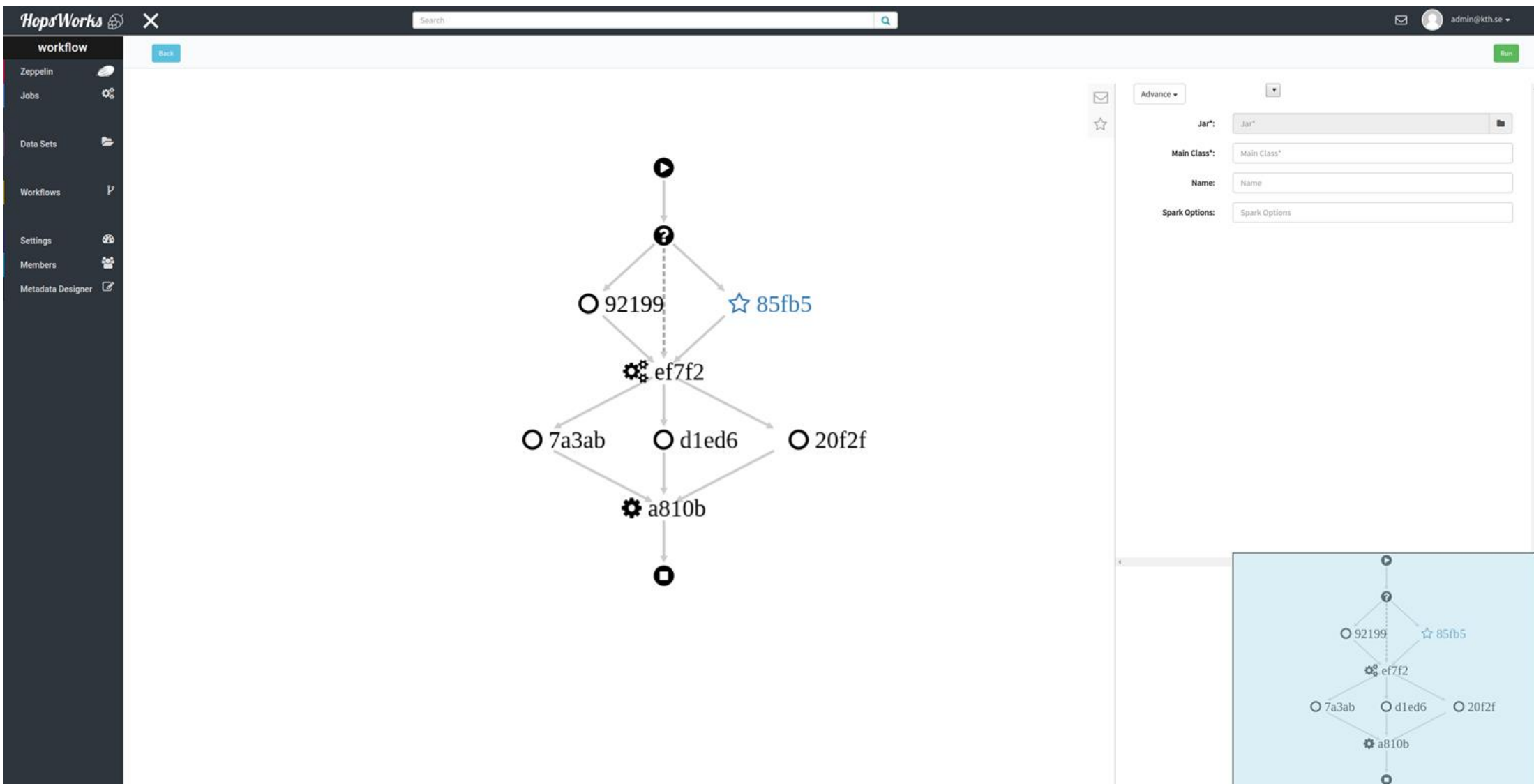
Hops.site



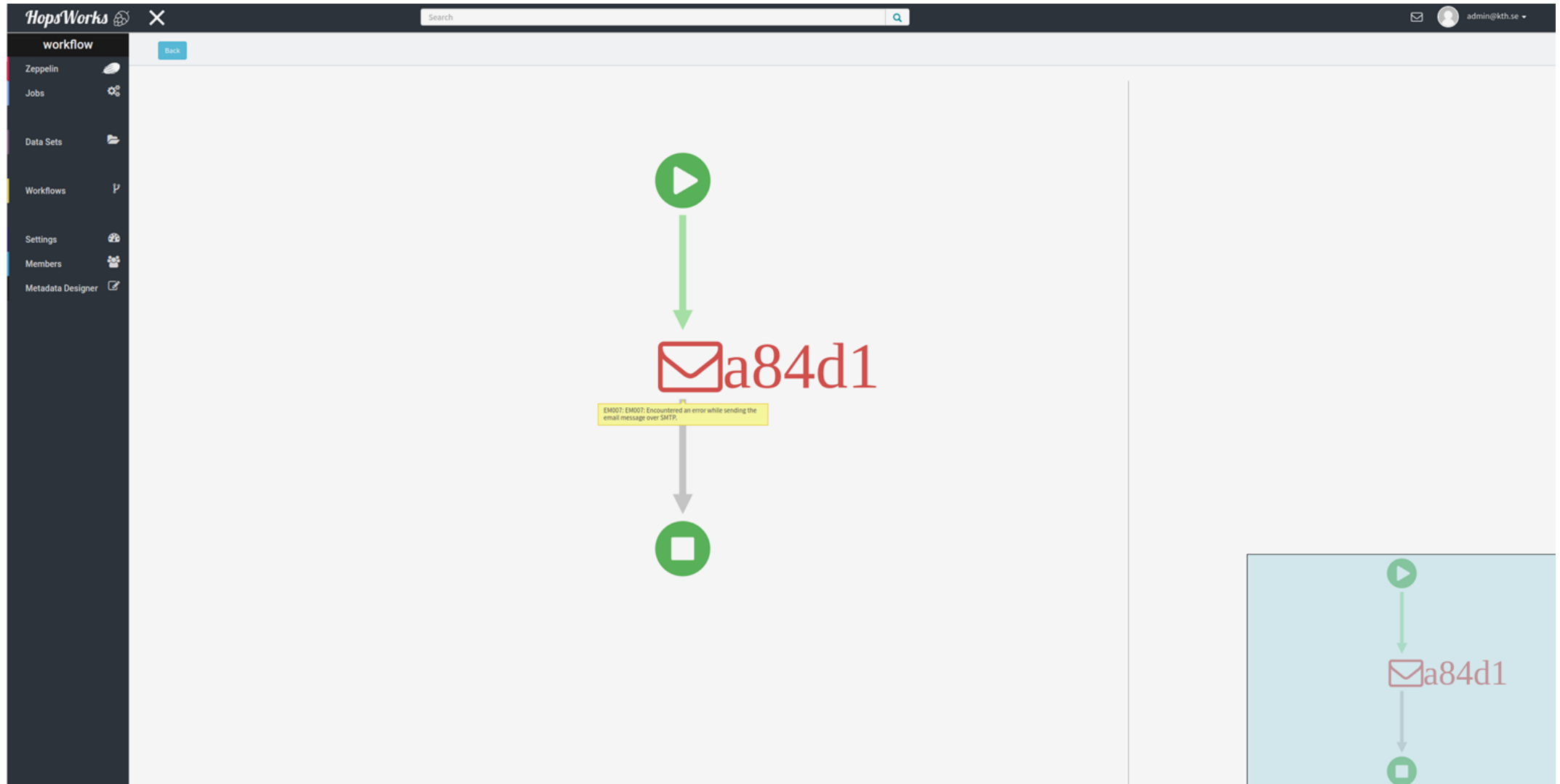
Oozie

- Oozie is a workflow scheduler system to manage Apache Hadoop jobs.
- Oozie Coordinator: jobs triggered by time and data availability.
- Oozie is integrated with the rest of the Hadoop.

Oozie: Enterprise Quality Workflows



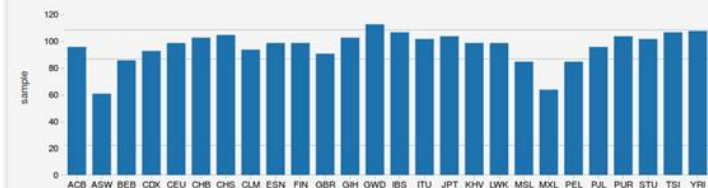
Oozie: Enterprise Quality Workflows



Visualisation

Let's look at how many individuals are in each of these populations in the panel

> display(panel)



To get a sense of this geographically, let's add an extra column to the panel dataframe for the ISO Alpha-3 Country Codes for a map visualization.

```
val countryMap = Map("FIN" -> "FIN", "CHS" -> "CHN", "GBR" -> "GBR", "PUR" -> "PRI", "CLM" -> "COL", "MXL" -> "MEX", "TSI" -> "ITA", "LWK" -> "KEN", "JPT" -> "JPN", "IBS" -> "ESP", "PEL" -> "PER", "CDX" -> "CHN", "YRI" -> "NGA", "KHV" -> "VNM", "ASW" -> "USA", "ACB" -> "BRB", "CHB" -> "CHN", "GIH" -> "IND", "GWD" -> "GMB", "PJI" -> "PAK", "MSL" -> "SLE", "BEB" -> "BGD", "ESN" -> "NGA", "STU" -> "LKA", "ITU" -> "IND")
def udftoCC = udf((pop: String) => {
  countryMap.get(pop)})
val panelWithCountryDF = panel.withColumn("CCode", udftoCC(panel("pop")))

countryMap: scala.collection.immutable.Map[String,String] = Map(PJI -> PAK, IBS -> ESP, FIN -> FIN, GWD -> GMB, PUR -> PRI, ITU -> IND, CHB -> CHN, JPT -> JPN, STU -> LKA, ACB -> BRB, MXL -> MEX, TSI -> ITA, GIH -> IND, ASW -> USA, PEL -> PER, CDX -> CHN, CLM -> COL, CHS -> CHN, BEB -> BGD, GBR -> GBR, YRI -> NGA, KHV -> VNM, MSL -> SLE, LWK -> KEN, ESN -> NGA)
udftoCC: org.apache.spark.sql.UserDefinedFunction
panelWithCountryDF: org.apache.spark.sql.DataFrame = [sample: string, pop: string, super_pop: string, gender: string, CCode: string]
```

> display(panelWithCountryDF)



<https://databricks.com/blog/2016/05/24/predicting-geographic-population-using-genome-variants-and-k-means.html>

Visualisation

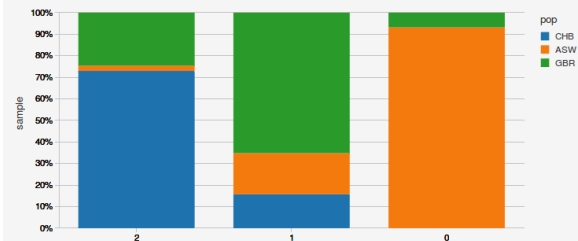
11) Predict populations, compute the confusion matrix.

```
> // Create predictionRDD that utilizes clusters.predict method to output the model's predictions
val predictionRDD: RDD[(String, Int)] = dataPerSampleId.map(sd => {
  (sd._1, clusters.predict(sd._2))
})

// Convert to DataFrame to more easily query the data
val predictDF = predictionRDD.toDF("sample", "prediction")

predictionRDD: org.apache.spark.rdd.RDD[(String, Int)] = MapPartitionsRDD[2546] at map at <console>:185
predictDF: org.apache.spark.sql.DataFrame = [sample: string, prediction: int]
```

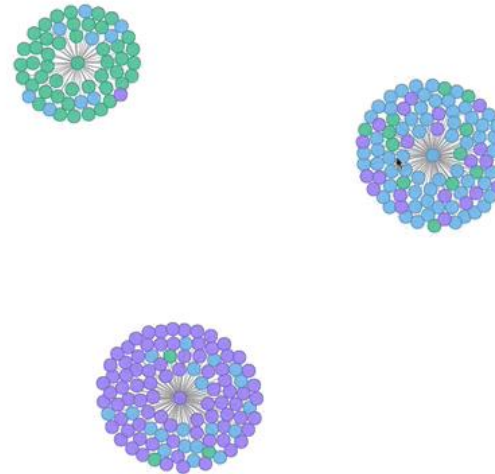
```
> // Join back to the filterPanel to get the original label
val resultsDF = filterPanel.join(predictDF, "sample")
display(resultsDF)
```



```
> resultsDF.registerTempTable("results_table")
```

```
> %r
resultsRDF <- sql(sqlContext, "SELECT pop, prediction FROM results_table")
confusion_matrix <- crosstab(resultsRDF, "prediction", "pop")
head(confusion_matrix)
```

```
prediction_pop CHB GBR ASW
1              2  89  30   3
2              1  14  58  17
3              0   0   3  41
```



<https://databricks.com/blog/2016/05/24/predicting-geographic-population-using-genome-variants-and-k-means.html>

- Workshop at SciLifeLab, May 22nd. Organized by Mikael Huss.
- Big Data task force, including Mikael Huss.
- SSF proposal submitted in april, Arne Elofsson

Conclusions

- Hadoop provide scalability and efficiency to the live sciences workflows.
- Hops.site offer you a place to use Hadoop to store, share and analyze your datasets.
- We are working on making it easier and more efficient to use Hadoop for live science.