**Overview**

* Simplified way of writing mappers and reducers
* Sits on top of MapReduce/TEZ
* Pig Latin
  + Scripting language that has a SQL-like syntax that lets you develop Map and Reduce steps
* Performance isn’t so much of an issue anymore
  + PIG can run on top of TEZ
    - TEZ is a much more efficient way of organizing MapReduce tasks
      * Uses a DAG for task management
    - TEZ runs x10 faster than MapReduce (? As claimed by Frank)
* Highly extensible UDF’s (user-defined-functions)

**Running Pig**

* Grunt
  + For experimentation
  + Run line by line
* Script
  + Script file
* Ambari/Hue (Gui)
  + GUI view

**Example**

* Relation
  + Dataset
* PigStorage
  + Use when you have a different delimiter in the input file
* Commands
  + LOAD
    - Load data from disk
  + AS
    - Speicifies schema
  + USING
    - … Using PigSotrage(‘|’) AS …
      * Specifying delimiter
  + DUMP
    - Dump out the contents of the relation (dataset)
  + GENERATE
    - Create new rows
  + DESCRIBE
    - Shows the schema
    - This seems similar to MySQL
  + FILTER
    - Similar to a WHERE clause
  + JOIN
    - Similar to JOIN clause
    - BY -> Join on
* Bag
  + Tuples related