**Create a graph database with relationships.**

**First File (Movie Data) has 5 fields: Movie id, Title, Year, Director and Category**

**Command to Load:**

LOAD CSV WITH HEADERS FROM 'file:///c:/Users/nacampa/Documents/dataanalytics/IS620/dfmovies.txt' AS row FIELDTERMINATOR "\t" CREATE (m cinema { movieid: row.Movieid, Title: row.Title, Year: toInt(row.Year), director: row.Director, category: row.Category})

**Second file (Actor Data) has 6 fields: Cast id, Movie id, Actor, Role, Role Desc**

**Command to Load:**

LOAD CSV WITH HEADERS FROM "file:///c:/Users/nacampa/Documents/dataanalytics/IS620/dfcast.txt" AS row FIELDTERMINATOR "\t" CREATE (a: cast {actorid: row.Castid, movieid: row.Movieid , Name: row.Actor, Role: row.Roleid, RoleDesc: row.Altroledesc})

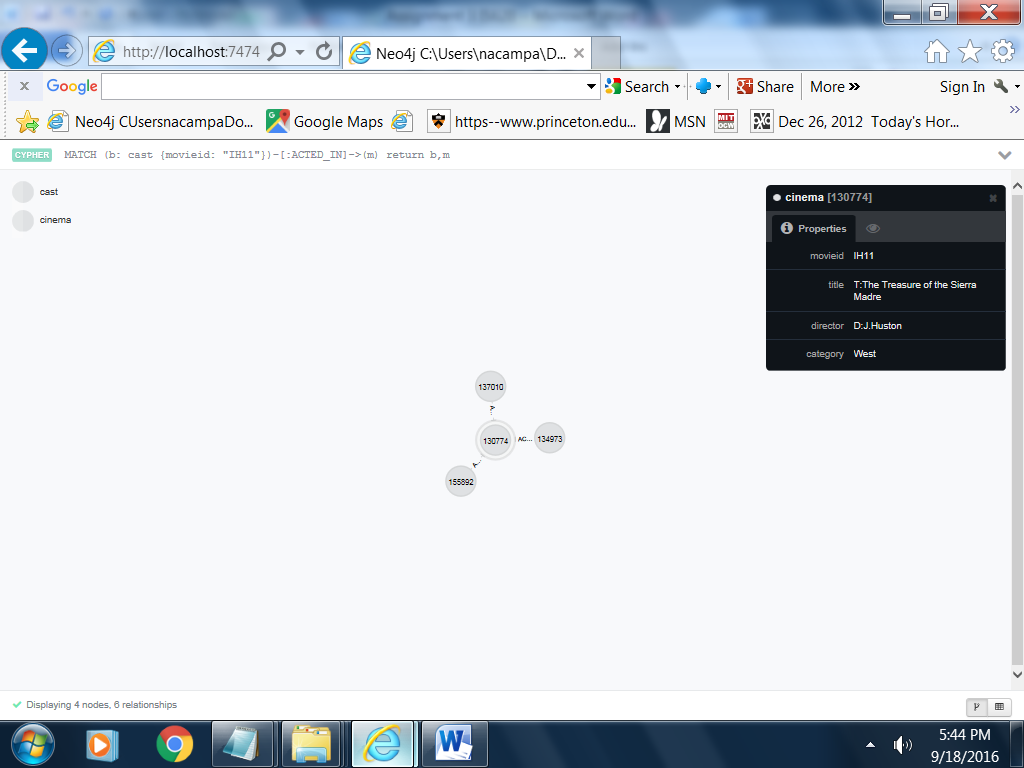
**Relationship created is [:ACTED\_IN]**

**With movie id as the common key (loaded a subset of actor data)**

LOAD CSV WITH HEADERS FROM "file:///c:/Users/nacampa/Documents/dataanalytics/IS620/dfcastrev2.txt" AS row FIELDTERMINATOR "\t" match(a: cast {movieId: row.Movieid}), (m: cinema {movieid: row.movieid})

**CREATE (a)-[r:ACTED\_IN {category: row.Category, Year: toInt(row.Year)}]->(m)**

**Node:** Movie**: The Treasure of Sierra Madre** Directed by: **J. Houston** Category**: Western**



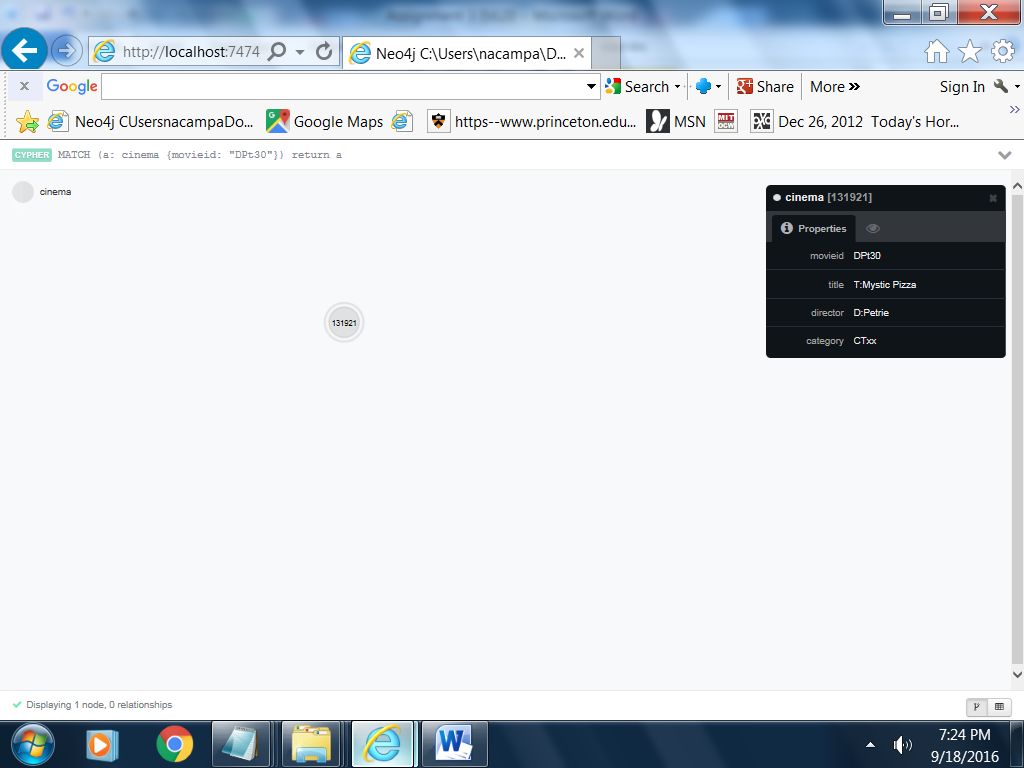
**Node:** ActorActor Name **: Alfonson Bedoya** Role Desc: **Primitive Bandit**

**Edge:** **[:ACTED\_IN] : The Treasure of Sierra Madre**

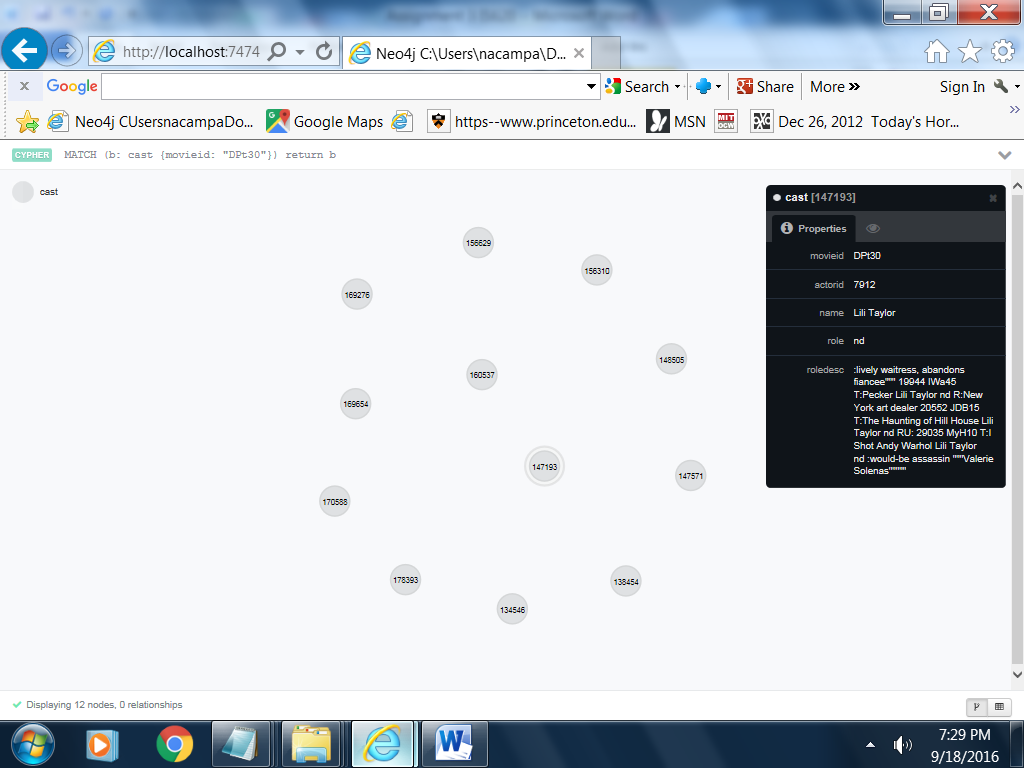


**Mystic Pizza (Movie id: “DPt30”)**

**Cypher command: MATCH (a: cinema {movieid "DPt30"}) return a**

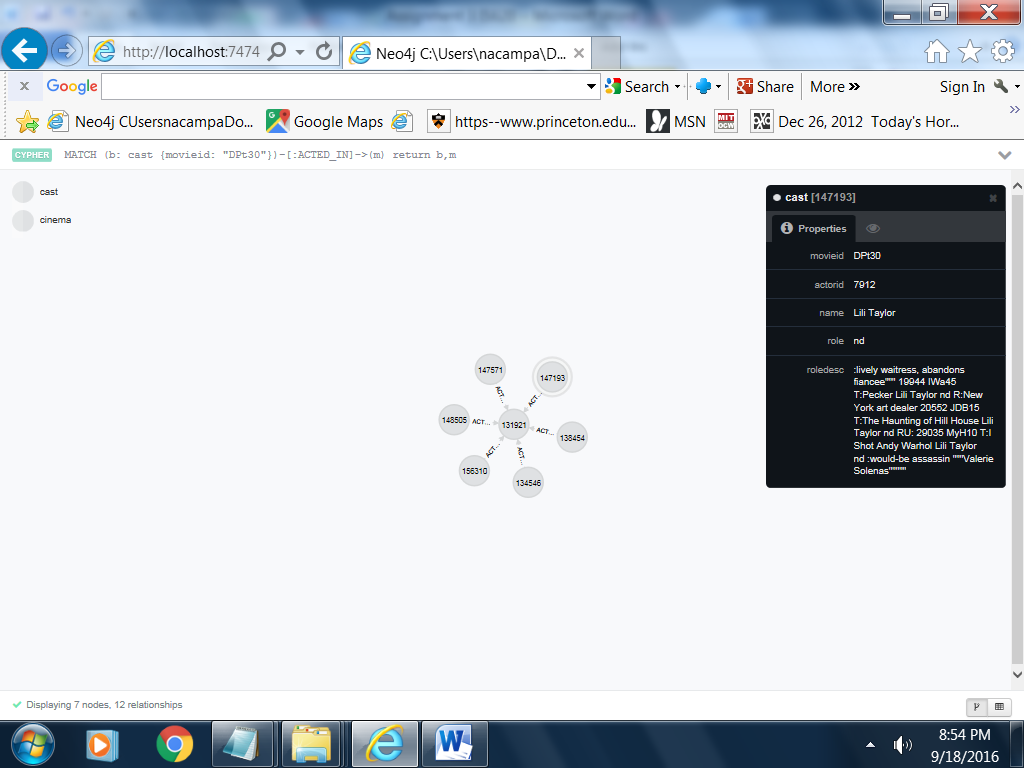


**Mystic Pizza Cast (Movie id: “DPt30”)**



**Mystic Pizza Cast (Movie id: “DPt30”)**

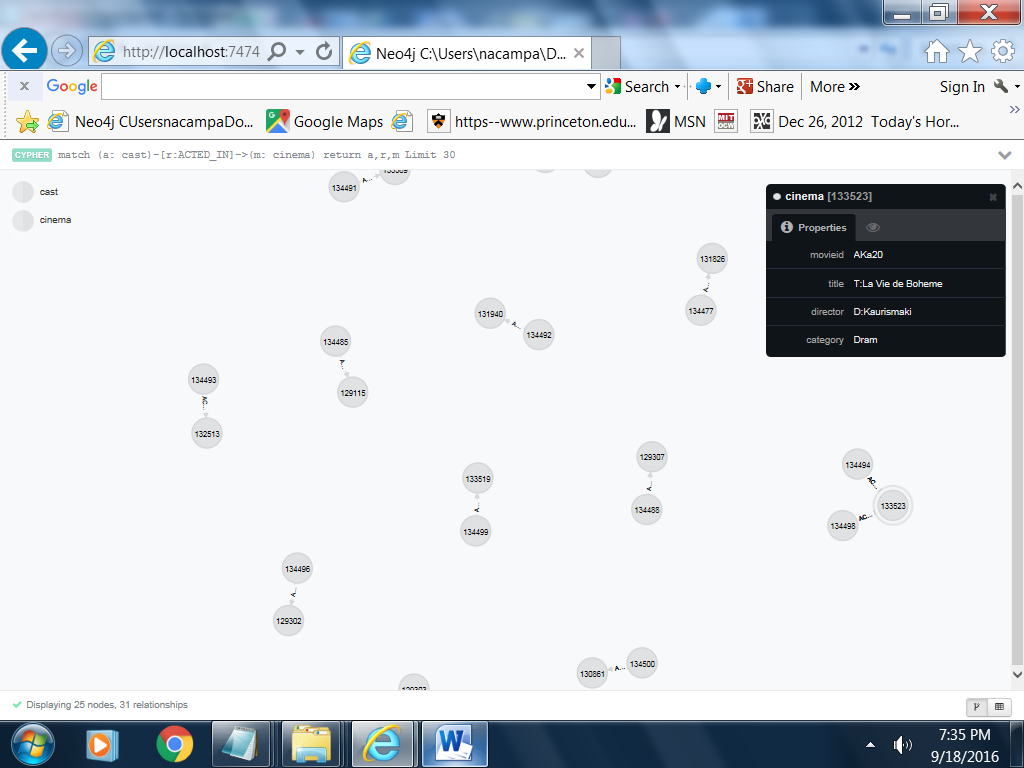
**Actors for a subset of data (6)**



**Example of Relationships:**

**Cypher command:**

**match (a: cast)-[r:ACTED\_IN]->(m: cinema) return a,r,m Limit 30**



**Metrics:**

**The number of Movie Nodes**: Match (m: cinema) return count(m)

**5,656**

The number of **Cast Nodes**: Match (c: cast) return count(c)

**44,166**

**The number of edges from cast to movie:** match (m: cast)-[r]->() return count(r)

**12,770**

The number edges from movie to cast: match (m: cinema)-[r]->() return count(r)

**261**

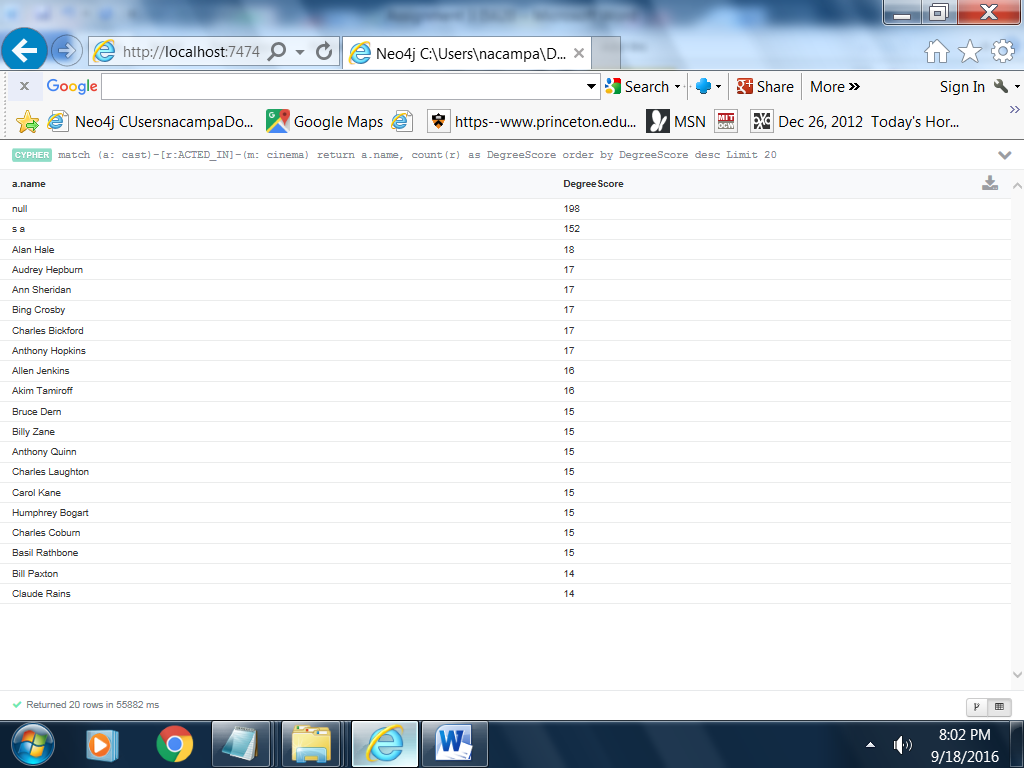
**Degree Centrality**: Measure of maximum number of edges incident on a node.

The results show the actors who acted in the most movies (from the subset of actors).

**These actors are or were very much in demand.**

match (a: cast)-[r:ACTED\_IN]-(m: cinema) return a.name, a.roledesc, count(r) as DegreeScore order by DegreeScore desc Limit 20

**Anthony Hopkins, Bruce Dern (among living actors) are/were in demand**.



Graph Diameter: Measures the longest shortest path between two cast member nodes.

match(n: cast), (m: cast) where n <> m with n, m

match p = shortestPath((n)-[r]->(m)) return n.name, m.name, length(p)

order by length(p) desc limit 1

match(n: cinema), (m: cinema) where n <> m with n, m

match p = shortestPath((n)-[r]->(m)) return n.title, m.title, length(p)

order by length(p) desc limit 1

**Between Centrality**: Measures the number of times a node is in the path that passes through the shortest path between two nodes.

The actor is important because of the connections between other actors.