In [1]:
```
def set_hadoop_config(credentials):
    prefix = "fs.swift.service." + credentials['name']
    hconf = sc._jsc.hadoopConfiguration()
    hconf.set(prefix + ".auth.url", credentials['auth_url']+'/v3/auth/tokens')
    hconf.set(prefix + ".auth.endpoint.prefix", "endpoints")
    hconf.set(prefix + ".tenant", credentials['project_id'])
    hconf.set(prefix + ".username", credentials['user_id'])
    hconf.set(prefix + ".password", credentials['password'])
    hconf.setInt(prefix + ".http.port", 8080)
    hconf.set(prefix + ".region", credentials['region'])
    hconf.setBoolean(prefix + ".public", True)
```

In [5]:
```
credentials_2 = {
  'auth_url':'https://identity.open.softlayer.com',
  'project':'object_storage_f962b5f8_4788_49ff_aa47_5e4673e53a2b',
  'project_id':'1b4094970a544940859cdd585d0f462c',
  'region':'dallas',
  'user_id':'cf7b734f80d54703bddedc60fe77bc33',
  'domain_id':'4212beab9a7f469391135e26f7219597',
  'domain_name':'1141491',
  'username':'admin_c68b1bc189f64a5099a7c50bfd7621dc0de4dbd2',
  'password':"""WX.Lkgg8z_Ud#P6l""",
  'filename':'License_Applications.csv',
  'container':'notebooks',
  'tenantId':'saa0-89a6bc0b359e28-b159885f0f89'
}
```

In [6]:
```
credentials_2['name'] = 'keystone'
set_hadoop_config(credentials_2)
```

In [41]:
```
from __future__ import division
import numpy as np

from pyspark.sql import SQLContext
sqlContext = SQLContext(sc)

# adding the PySpark modul to SparkContext
sc.addPyFile("https://raw.githubusercontent.com/seahboonsiew/pyspark-csv/master/pyspark_csv.py")
import pyspark_csv as pycsv

license = sc.textFile("swift://" + credentials_2['container'] + "." + credentials_2['name'] + "/License_Appli
cations.csv")

def skip_header(idx, iterator):
    if (idx == 0):
        next(iterator)
    return iterator

license_header = collisions.first()


license_header_list = license_header.split(",")
license_body = license.mapPartitionsWithIndex(skip_header)

# filter not valid rows
license_body = license_body.filter(lambda line : len(line.split(","))>24)

# create Spark DataFrame using pyspark-csv
license_df = pycsv.csvToDataFrame(sqlContext, license_body, sep=",", columns=license_header_list)
#license_df.cache()
```

In [44]:
```
license_df.printSchema()
```

```
root
 |-- Application ID: string (nullable = true)
 |-- License Number: string (nullable = true)
 |-- License Type: string (nullable = true)
 |-- Application or Renewal: string (nullable = true)
 |-- Business Name: string (nullable = true)
 |-- Status: string (nullable = true)
 |-- Start Date: timestamp (nullable = true)
 |-- End Date: timestamp (nullable = true)
 |-- Temp Op Letter Issued: timestamp (nullable = true)
 |-- Temp Op Letter Expiration: timestamp (nullable = true)
 |-- License Category: string (nullable = true)
 |-- Application Category: string (nullable = true)
 |-- Building Number: string (nullable = true)
 |-- Street: string (nullable = true)
 |-- Street 2: string (nullable = true)
 |-- Unit Type: string (nullable = true)
 |-- Unit: string (nullable = true)
 |-- Description: string (nullable = true)
 |-- City: string (nullable = true)
 |-- State: string (nullable = true)
 |-- Zip: string (nullable = true)
 |-- Contact Phone: string (nullable = true)
 |-- Longitude: double (nullable = true)
 |-- Latitude: double (nullable = true)
 |-- Active Vehicles: string (nullable = true)
```

In [45]:
```
license_df.take(1)
```

Out[45]:
```
[Row(Application ID=u'10447-2016-RGEV', License Number=u'2004323-DCA', License Type=u'Individual', Applicati
on or Renewal=u'Renewal', Business Name=u'HONG BO LIANG', Status=u'Issued', Start Date=datetime.datetime(201
6, 8, 8, 0, 0), End Date=datetime.datetime(2016, 8, 8, 0, 0), Temp Op Letter Issued=None, Temp Op Letter Exp
iration=None, License Category=u'General Vendor', Application Category=u'Special', Building Number=u'60', St
reet=u'ROFF ST', Street 2=None, Unit Type=None, Unit=None, Description=None, City=u'STATEN ISLAND', State=
u'NY', Zip=u'10304', Contact Phone=u'6462807901', Longitude=None, Latitude=None, Active Vehicles=None)]
```

In [46]:
```
license_df.count()
```

Out[46]:
```
3539
```

In [21]:
```
!pip install --user seaborn
```

```
Requirement already satisfied (use --upgrade to upgrade): seaborn in /gpfs/global_fs01/sym_shared/YPProdSpar
k/user/saa0-89a6bc0b359e28-b159885f0f89/.local/lib/python2.7/site-packages
```

In [47]:
```
%matplotlib inline

import matplotlib.pyplot as plt
# matplotlib.patches allows us create colored patches, we can use for legends in plots
import matplotlib.patches as mpatches
# seaborn also builds on matplotlib and adds graphical features and new plot types
import seaborn as sns
import pandas as pd
```

In [48]:
```
license_pd = license_df[['Application ID', 'License Number', 'License Type', 'Application or Renewal',
                         'Business Name', 'Status', 'Start Date',
                         'End Date', 'License Category','Application Category','Building Number',
                         'Street','City','State','Zip','Contact Phone','Longitude','Latitude',
                         'Active Vehicles']].toPandas()
```

In [49]:
```
license_pd.head(2)
```

Out[49]:

	Application ID	License Number	License Type	Application or Renewal	Business Name	Status	Start Date	End Date	License Category	Application Category	Building Number	Street
0	10447-2016-RGEV	2004323-DCA	Individual	Renewal	HONG BO LIANG	Issued	2016-08-08	2016-08-08	General Vendor	Special	60	ROFF ST
1	10033-2015-RDPD	1438257-DCA	Business	Renewal	ALL SOUTH-SHORE MEDICAL SUPPLY INC.	Issued	2015-03-02	2015-03-03	Dealer In Products	Basic	221	MERRICK RD

In [50]:
```
license_pd.tail(5)
```

Out[50]:

	Application ID	License Number	License Type	Application or Renewal	Business Name	Status	Start Date	End Date	License Category	Application Category	Buildir Numbe
3534	14970-2015-RHIC	1234557-DCA	Business	Renewal	BHALLI, MOHAMMED	Issued	2015-04-07	2015-04-08	Home Improvement Contractor	Special	1927
3535	10410-2015-RHIC	1474094-DCA	Business	Renewal	MIHHEIKIN REMODELING LLC	Issued	2015-04-07	2015-04-08	Home Improvement Contractor	Special	213
3536	14794-2015-ACRD	2030472-2-DCA	Business	Application	SAMMY DELI CORP.	Issued	2015-11-16	2015-11-17	Cigarette Retail Dealer	Basic	120
3537	10482-2016-RGEV	1181237-DCA	Individual	Renewal	WILLIAM H SMALLWOOD	Issued	2016-08-09	2016-08-09	General Vendor	Special	417
3538	10194-2015-RHIC	1469724-DCA	Business	Renewal	SM CONTRACTING NY INC.	Denied	2015-07-29	2015-07-29	Home Improvement Contractor	Special	993

In [51]:
```
license_pd.describe()
```

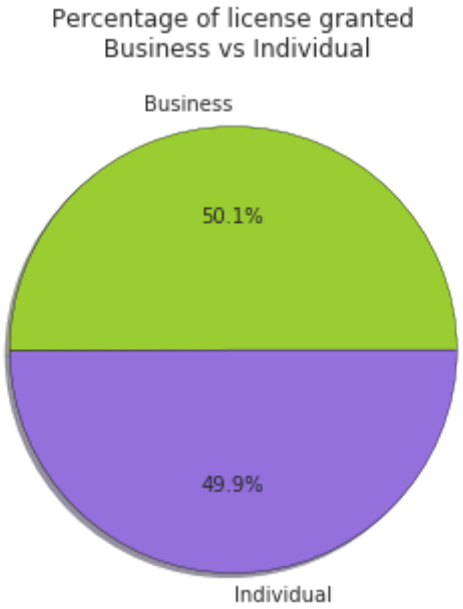
Out[51]:

	Longitude	Latitude
count	1436.000000	1436.000000
mean	-73.929352	40.724674
std	0.085970	0.082434
min	-74.253761	40.502660
25%	-73.984116	40.670039
50%	-73.932102	40.728010
75%	-73.876380	40.774988
max	-73.707479	40.907194

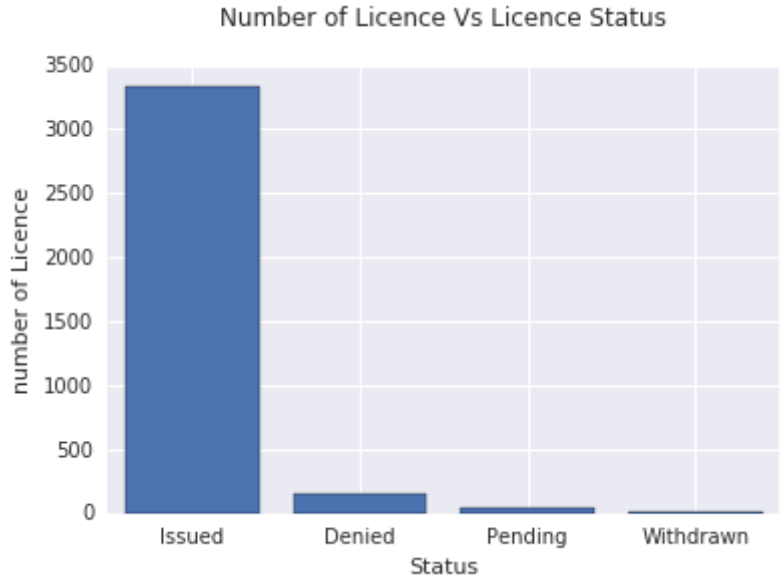
In [52]:
```
import requests, StringIO, pandas as pd, json, re
import matplotlib as plt
%matplotlib inline

d = license_pd['License Type'].value_counts()
print d
```

```
Business      1773
Individual    1766
Name: License Type, dtype: int64
```

In [53]:
```
%matplotlib inline

labels = d.keys()
i = 0
sizes = []
while i < len(d):
    sizes.append(d.get(labels[i]))
    i += 1
colors = ['yellowgreen', 'mediumpurple']
plt.pyplot.pie(sizes,              # data
        labels=labels,      # slice labels
        colors=colors,      # array of colours
        autopct='%1.1f%%',  # print the values inside the wedges
        shadow=True,        # enable shadow
        startangle=0       # starting angle
        )
plt.pyplot.axis('equal')
plt.pyplot.title('Percentage of license granted\n Business vs Individual\n\n')
plt.pyplot.show()
```


Percentage of license granted
Business vs Individual
Business
50.1%
49.9%
Individual

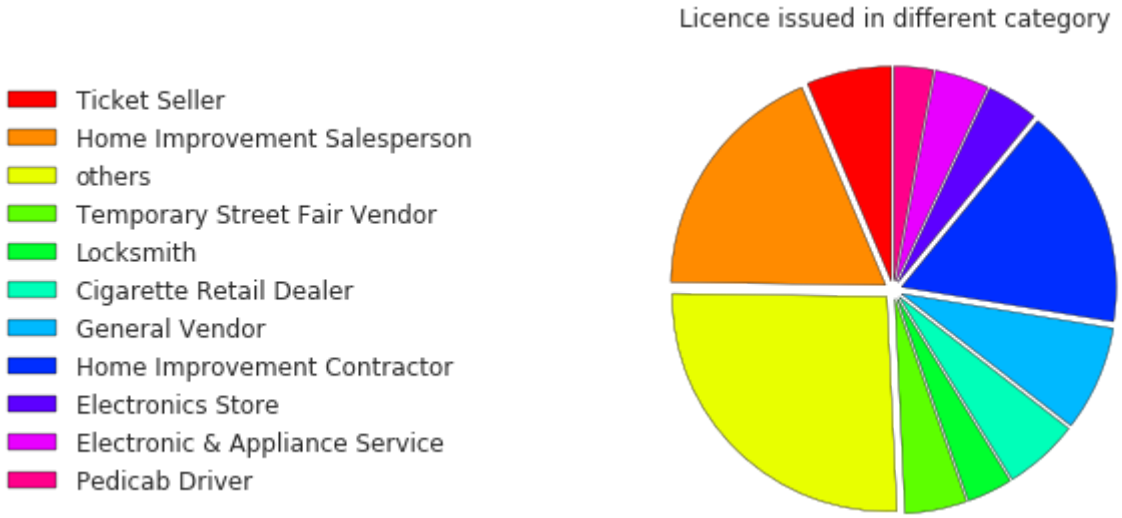
In [55]:
```
d = license_pd['Status'].value_counts()
valList = []
for val in d.keys():
    valList.append(d[val])
plt.pyplot.bar(range(len(d)), valList, align='center')
plt.pyplot.xticks(range(len(d)), d.keys())
plt.pyplot.grid(True)
plt.pyplot.xlabel('Status')
plt.pyplot.ylabel('number of Licence')
plt.pyplot.title('Number of Licence Vs Licence Status\n')
plt.pyplot.show()
```



In [56]:
```
license_pd['State'].value_counts()
```

Out[56]:
```
NY             3310
NJ              115
PA               22
CT               10
TX                9
VA                6
CA                5
DE                5
OH                5
FL                4
NC                4
MA                2
CO                2
MD                2
NH                2
IL                2
MI                2
MO                2
PHILIPPINES       1
OK                1
Cheshire          1
GA                1
AZ                1
UT                1
IN                1
NE                1
SURREY            1
KS                1
SC                1
KY                1
LONDON            1
WA                1
Name: State, dtype: int64
```

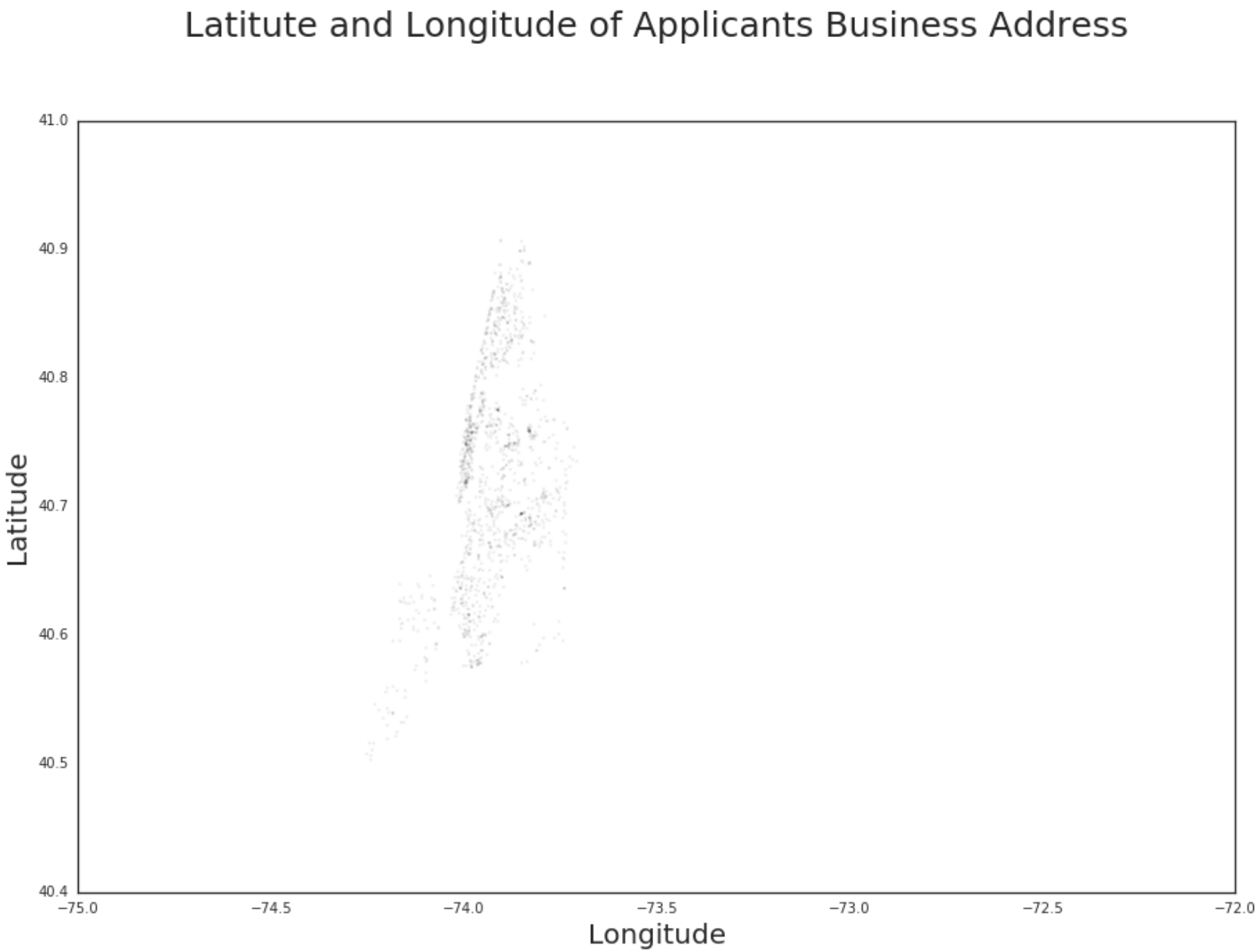
In [60]:
```
%matplotlib inline
import colorsys

df = license_pd['License Category'].value_counts()
others=0
list ={}
for val in df.keys():
    if df[val] < 100:
        others = others + df[val]
    else:
        list[val] = df[val]
list['others'] = others

labels = list.keys()
i = 0
values = []
while i < len(list):
    values.append(list.get(labels[i]))
    i= i+1

explode = []
for k in labels:
    explode.append(0.05)

HSV_tuples = [(x*1.0/i, 1, 1) for x in range(i)]
RGB_tuples = map(lambda x: colorsys.hsv_to_rgb(*x), HSV_tuples)

patches, texts = plt.pyplot.pie(values,colors= RGB_tuples, explode=explode, startangle=90, radius=1.2)
plt.pyplot.legend(patches, labels, loc='best', bbox_to_anchor=(-0.1, 1.),fontsize=12)
plt.pyplot.axis('equal')
plt.pyplot.title('Licence issued in different category\n')
plt.pyplot.show()
```



In [61]:
```
#Application Status Distribution by Group

df1 = license_pd[['Application or Renewal','Status','State']]
counts = df1.groupby(['Application or Renewal','Status']).size();
counts
```

Out[61]:
```
Application or Renewal  Status
Application             Denied        120
                        Issued       1605
                        Pending        20
                        Withdrawn       9
Renewal                 Denied         39
                        Issued       1726
                        Pending        20
dtype: int64
```

In [64]:
```
import seaborn as sns
import matplotlib as plt

%matplotlib inline

#adjust settings
sns.set_style("white")
plt.pyplot.figure(figsize=(15,10))

#create scatterplots
plt.pyplot.scatter(license_pd.Longitude, license_pd.Latitude, alpha=0.05, s=4, color='black')

#adjust more settings
plt.pyplot.title('Latitute and Longitude of Applicants Business Address\n\n', size=25)
plt.pyplot.xlim((-75,-72))
plt.pyplot.ylim((40.4,41))
plt.pyplot.xlabel('Longitude',size=20)
plt.pyplot.ylabel('Latitude',size=20)

plt.pyplot.show()
```


Latitute and Longitude of Applicants Business Address
Latitude
Longitude
41.0
40.9
40.8
40.7
40.6
40.5
40.4
−75.0
−74.5
−74.0
−73.5
−73.0
−72.5
−72.0

In [ ]: