In [28]:
```
import fitsio
import numpy as np
import astropy.table
import matplotlib.pyplot as plt
from lsst.rsp import get_tap_service
rsp_tap = get_tap_service("tap")
```

In [4]:
```
f = fitsio.read('spAll-mini-v6_2_1.fits')
```

In [36]:
```
irandom = np.arange(len(f), dtype=np.int32)
np.random.shuffle(irandom)
irandom = irandom[0:100000]
fsmall = astropy.table.Table(f[irandom])
```

In [37]:
```
query = """
        SELECT objectId, coord_ra, coord_dec,
        ut1.RACAT AS ut1_ra, ut1.DECCAT AS ut1_dec, ut1.SDSS_ID AS ut1_sdss_
        FROM dp1.Object, TAP_UPLOAD.ut1 AS ut1
        WHERE CONTAINS(POINT('ICRS', coord_ra, coord_dec),
        CIRCLE('ICRS', ut1.RACAT, ut1.DECCAT, 0.00027))=1
        ORDER BY coord_ra ASC
        """
```

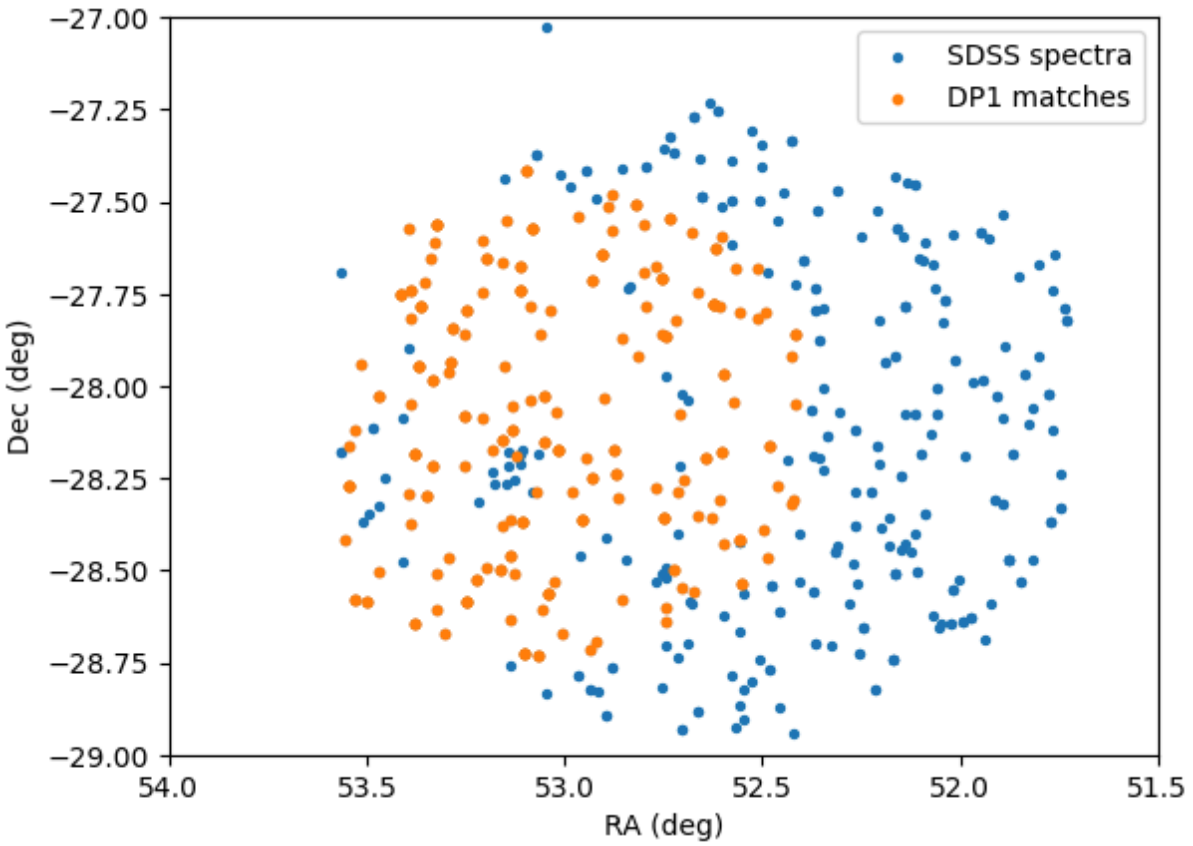
In [38]:
```
job = rsp_tap.submit_job(query, uploads={"ut1": fsmall})
job.run()
job.wait(phases=['COMPLETED', 'ERROR'])
print('Job phase is', job.phase)
if job.phase == 'ERROR':
    job.raise_if_error()

assert job.phase == 'COMPLETED'
results = job.fetch_result()
print(len(results))
```

```
Job phase is COMPLETED
235
```

In [60]:
```
plt.scatter(fsmall['RACAT'], fsmall['DECCAT'], s=8, label='SDSS spectra')
plt.scatter(results['coord_ra'], results['coord_dec'], s=10, label='DP1 matc
plt.xlim([54, 51.5])
plt.ylim([-29., -27.])
plt.legend()
plt.xlabel('RA (deg)')
plt.ylabel('Dec (deg)')
```

Out[60]:
```
Text(0, 0.5, 'Dec (deg)')
```



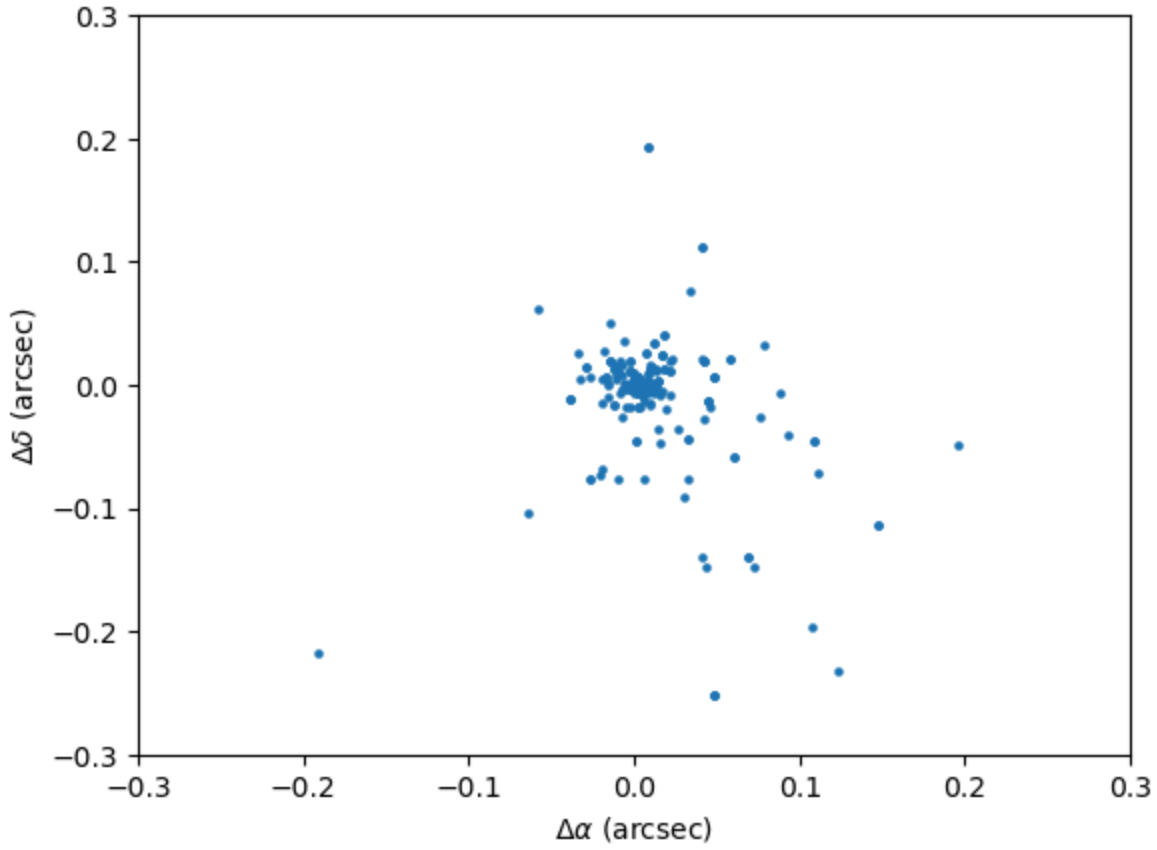
In [57]:
```
print(results.fieldnames)
```

```
('objectId', 'coord_ra', 'coord_dec', 'ut1_ra', 'ut1_dec', 'ut1_sdss_id')
```

In [65]:
```
dra = (results['coord_ra'] - results['ut1_ra']) * np.cos(results['coord_dec'
ddec = (results['coord_dec'] - results['ut1_dec']) * 3600.
```

In [68]:
```
plt.scatter(dra, ddec, s=5)
plt.xlabel(r'$\Delta\alpha$ (arcsec)')
plt.ylabel(r'$\Delta\delta$ (arcsec)')
plt.xlim([-0.3, 0.3])
plt.ylim([-0.3, 0.3])
```

Out[68]:
```
(-0.3, 0.3)
```



In [ ]: