International Center for Law & Economics

Comments of the International Center for Law & Economics

Comments, RE: Monitoring DISH's Compliance with Conditions Granting an Extension of Time to Complete Construction of Facilities and Buildout Commitments, GN Docket No. 22-212

June 5, 2025

Authored by:

Kristian Stout (Director of Innovation Policy, International Center for Law & Economics)

June 5, 2025
VIA ECFS
Marlene H. Dortch
Secretary
Federal Communications Commission
445 12th Street SW
Washington, DC 20554

Re: Reply Comments from the International Center for Law & Economics, Monitoring DISH's Compliance with Conditions Granting an Extension of Time to Complete Construction of Facilities and Buildout Commitments, GN Docket No. 22-212

Dear Ms. Dortch:

We submit this letter and the attached study in response to the Public Notice seeking supplemental comments on VTel's Petition for Reconsideration[1] of the extension of construction deadlines for certain licenses held by EchoStar Corporation. As demonstrated below in our examination of select filings, we identify evidence of technical discrepancies in the showings made by EchoStar that suggest the possibility of a much larger problem with EchoStar's license construction notifications. In our attached appendix we present the code necessary for replicating this work across all BEAs. We believe the FCC should conduct a thorough evaluation of EchoStar's submissions and the network it claims to have constructed. If the FCC's analysis, as the study suggests, concludes that EchoStar failed to construct the network it agreed to build, then the FCC must take immediate action to ensure that EchoStar's spectrum does not continue to remain fallow.

The U.S. wireless market is highly competitive because the Commission has taken meaningful steps to make spectrum available to companies that put the spectrum to work and to provide a high-speed mobile broadband service for consumers. These networks have enabled substantial growth in the U.S. economy and fostered the development of new and novel technologies that benefit all Americans. Spectrum used for modern mobile broadband networks has significant impacts on the U.S. economy and the future economic security of the country and when spectrum is not being put to its highest and best use, the impacts extend far beyond a licensee and its contractors. But the FCC's options to allocate additional spectrum for mobile broadband service are constrained by several factors. First, the inventory of spectrum suitable for mobile broadband service must be identified and licensed. Second, there is no greenfield spectrum available; therefore, incumbent operations must be protected or moved to different bands – all of which takes years. Even then, the FCC's authority to auction spectrum has lapsed, so the FCC does not have general authority to auction spectrum that is newly allocated for mobile broadband service.

---

[1] *Petition for Reconsideration of Vtel Wireless Inc.*, WT Docket No. 22-212 (Oct. 21, 2024), https://www.fcc.gov/ecfs/search/search-filings/filing/1021167931859.

With the dearth of options for acquiring new spectrum, the FCC must ensure that allocated and licensed spectrum is put to use. Failure by any licensee to comply with its obligations to use the spectrum should be met with swift action so the licenses can be reauctioned or sold to an entity that will make use of this valuable and highly constrained resource. This is why it is so important for the FCC to confirm whether or not EchoStar is warehousing spectrum that could otherwise be put to work.

Over many years, EchoStar has amassed substantial, valuable spectrum assets with a history of repeatedly seeking extensions of time for putting those resources to use in the public interest. In 2017 and 2018, EchoStar accepted penalties rather than meeting interim deployment milestones, effectively delaying its obligation to deploy initial service.[2] In 2019 and 2024, it sought extensions of time to allow it to further delay deployment.[3] Now there is substantial doubt as to whether EchoStar has met even its delayed obligations. In our review of the record in this docket, we are particularly persuaded by some technical discrepancies in the EchoStar data identified by Spectrum Financial Partners, LLC (Spectrum Financial):[4]

- *First*, Spectrum Financial noted that EchoStar demonstrated coverage for mid-band spectrum licenses exclusively using coverage data for the 600 MHz band, which has significantly better propagation characteristics than the mid-band spectrum licensed to EchoStar.[5] This approach seems inherently inconsistent with the Commission's construction requirements.
- *Second*, Spectrum Financial points out that there is a question regarding whether the link budget for the coverage compliance filings made by EchoStar—which rely on a 56 kbps uplink target—are compliant with its commitments to provide 5G "broadband" service. Spectrum Financial highlights that the data used for the construction filings is much lower than the 320 kbps target used as the limit for voice coverage in EchoStar's Broadband Data Collection filings.
- *Third*, Spectrum Financial provides actual field measurements throughout ICLE's home state of Oregon showing minimal activity on EchoStar licensed bands. These measurements are not consistent with the robust network EchoStar claims to operate.

Spectrum Financial's findings regarding EchoStar's use of spectrum in Oregon appear to be supported by ICLE's independent review of public data. As Spectrum Financial documents,

---

[2] *See Letter from Donald K. Stockdale, Jr., Chief, Wireless Telecommunications Bureau, Federal Communications Commission to Jeffrey H. Blum, Senior Vice President & Deputy General Counsel, DISH Network LLC* (Jul. 9, 2018), *available at* https://docs.fcc.gov/public/attachments/doc-352379a1.pdf.

[3] *See Wireless Telecommunications Bureau Seeks Supplemental Comment on Vtel's Petition for Reconsideration of the Extension of Construction Deadlines for Certain Licenses Held by Echostar Corp.*, WT Docket No. 22-212 (May 12, 2025), *available at* https://docs.fcc.gov/public/attachments/DA-25-404A1.pdf.

[4] *Comments of Spectrum Financial Partners LLC*, WT Docket No. 22-212 (May 27, 2025), https://www.fcc.gov/ecfs/search/search-filings/filing/10528678810841.

[5] It is not entirely clear what EchoStar's statements concerning its coverage mean, as it uses the term "Uplink Limited" when describing them, without further explanation. At a minimum, the commission should require clarification from EchoStar before accepting its filings.

EchoStar's construction certifications tell—at most—half of the story. The AWS-3 I Block build showing for BEA055 attached by Spectrum Financial, for example, contains *no* coverage data for the license for which it is filed. Instead, it shows coverage for an entirely different spectrum band.[6] In fact, in each coverage map reviewed by ICLE, EchoStar certifies that the "Minimum Received Signal Level (RSRP) is Uplink Limited" and "value shown/plotted is for n71," where n71 is the 600 MHz band. Thus, as the best can be ascertained, for each map reviewed by ICLE, it appears that EchoStar is not demonstrating coverage solely of its AWS-3 I Block spectrum license, but instead is showing coverage using multiple licenses. To put it another way, implicit in EchoStar's presentation is the claim that the areas of 600 MHz uplink coverage—represented in the RSRP plots—also define the areas where service exists for its other licensed spectrum. In effect, EchoStar is asking the Commission to treat its 600 MHz uplink map as not just one component of service coverage, but as the complete intersection of uplink and downlink coverage across all bands. This would only be valid if, in every case, the downlink signal from AWS-3, AWS-4, H Block, or 700 MHz is weaker than—or at best equal to—the 600 MHz uplink signal. Given the known differences in propagation, building penetration, and spectrum characteristics, this outcome is highly unlikely.

This raises significant legal and policy concerns:

- As ICLE understands it, EchoStar is using the band for which it is filing for the base-to-mobile downlink and using the 600 MHz band for the mobile-to-base uplink. In the case of paired spectrum, like AWS-3 Blocks G-J and the AWS H Block, EchoStar thus appears to not use half the spectrum for which it is licensed—the uplink portion of the band for which it is filing a construction certification.
- EchoStar's construction filings for the AWS-3 A1 and B1 blocks claim to be uplink limited and use Reference Signal Received Power (RSRP) plots for the 600 MHz band. RSRP plots typically are used to measure downlink signal strength and quality. But the AWS-3 A1 and B1 blocks are uplink only bands. Aside from that, there is also no explanation how the coverage of the uplink transmissions from the AWS-3 band could be limited by the uplink transmissions from the 600 MHz band.
- EchoStar is asking the Commission to take on pure faith—while providing zero empirical data in support—that there is not a single spot in the U.S. where, due to some propagation anomaly or localized conditions, EchoStar's 700 MHz, AWS-3, AWS-4 and H Block downlink coverage does not equal or exceed its 600 MHz uplink coverage (using a low edge speed of 56 kbps). The only way EchoStar's claimed coverage boundaries could be correct is if, in every situation, the downlink signal from higher bands was always weaker than the uplink signal over 600 MHz — which defies both physics and field experience. This belies credulity, especially given the comparative network coverage included by Spectrum Financial that contrasts EchoStar's claimed "AWS" coverage against the AWS coverage of the more mature T-Mobile and AT&T mid-band networks in the same area.

---

[6] *See, e.g.*, FCC ULS File Number 0011376642.

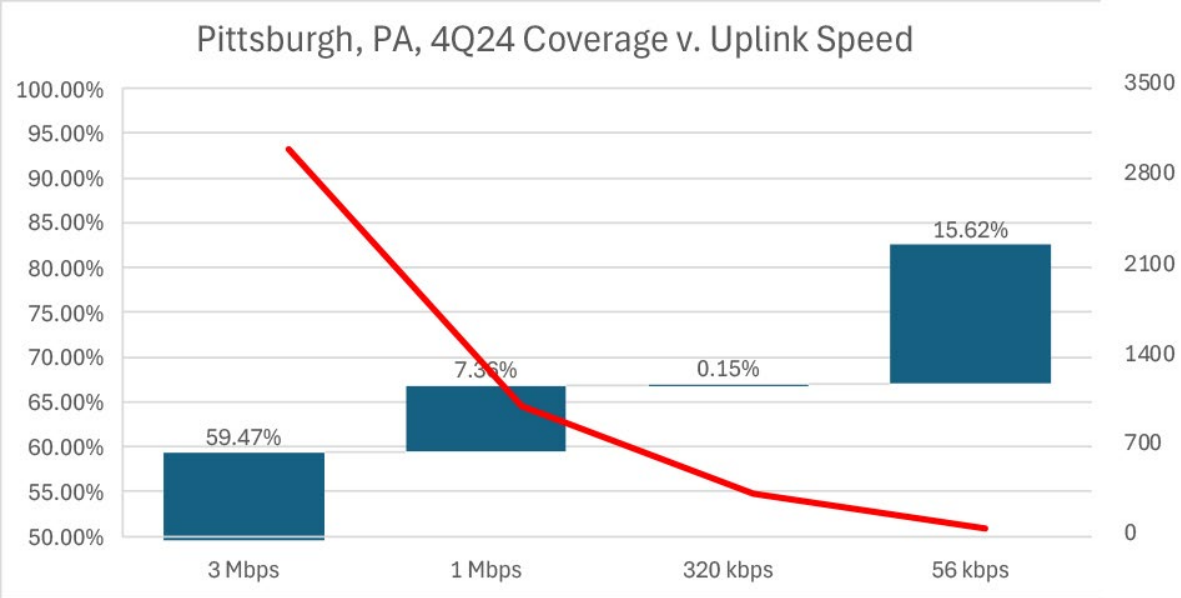
Concerned by these discrepancies, ICLE reviewed certain December 2024 ULS Construction showings by EchoStar which were filed to demonstrate EchoStar met the 80% accelerated coverage requirements. Each of these filings raised significant concerns. For example, EchoStar filed five license construction showings for BEA053—Pittsburgh, PA. Even though each filing purports to show coverage that is 600 MHz-limited (and therefore each filing should be identical), the coverage figures in the certifications ranged from 82.1% for WQTX252 to 82.6% for WQJY993 (although a variance of 0.5% may not seem large, consider that EchoStar purported to have met its national coverage obligation with a margin of only 0.08%).[7]

ICLE also compared the coverage data in those filings to EchoStar's BDC voice coverage data.[8] ICLE overlaid the population data with Pennsylvania counties using Census Bureau GIS data, prorating the population by county overlap and H3 resolution 9 overlap. ICLE found that EchoStar's coverage—aggregating 5G and voice—covered only 66.98% of the population in BEA053. EchoStar presumably will point to its use of a 56-kbps uplink threshold for coverage as explaining the difference in percentage of population covered. However, the incremental coverage difference between 5G with a 1 Mbps uplink and the 320-kbps voice uplink was only 0.15% of the population (4,337 people). Meanwhile, EchoStar claims that using a 56 kbps uplink threshold allows it to reach 82.6% of the population—an increase of approximately 15.62 percentage points over the 66.98% covered at the 320 kbps threshold (59.5% + 7.36% + 0.15%). It is implausible that reducing the uplink speed from 1 Mbps to 320 kbps yields only a 0.15% coverage increase, but reducing it further to 56 kbps would result in a tenfold increase in coverage. This sharp inflection suggests not a genuine expansion of usable service, but a use of technical thresholds to inflate coverage figures without meaningfully improving network performance. This inconsistency is illustrated in the chart below showing a waterfall of covered population versus uplink speed, with the line showing the decrease in speed.

---

[7] *See, e.g.*, *Letter from EchoStar Corp.*, WT Docket No. 22-212 (Jan. 10, 2025), https://www.fcc.gov/ecfs/document/1011082930532/1.

[8] The FCC recently released data as of Dec. 31, 2024—the exact date that EchoStar's coverage obligations under the September 2024 extension matured. ICLE downloaded EchoStar's Pennsylvania H3 coverage files from the BDC website for 5G-NR (35/3 mbps), 5G-NR (7/1 Mbps) and mobile voice, as well as 2023 H3 400m population data from the Humanitarian Data Exchange: https://data.humdata.org/dataset/kontur-population-united-states-of-america.



ICLE also compared EchoStar's six showings[9] in BEA011 (and PEA048, which comprises the identical set of counties) for the Harrisburg, PA market with its BDC filings for both December 31, 2024, and June 30, 2024. Applying the same population coverage methodology, EchoStar's BDC data for the second quarter—three months before it asked for an extension—shows EchoStar had already met its "accelerated" requirement and that EchoStar's coverage was only 0.06% less (738 people) than it was as of the December deadline. As much as EchoStar now states it made "significant investments" in reliance on the extension and that its extension was not a "simple plea for more time," EchoStar's commitment with respect to Harrisburg—and ICLE presumes many other markets—was simply the fulfillment of an empty promise.

That suggests there may have been no *quid* in the *quid pro quo* EchoStar alludes to when it represents "the milestone extensions that EchoStar received were *in exchange* for the acceleration of the milestones of other licenses as well as substantial new commitments" (emphasis in original).[10] If EchoStar were not making the substantial commitments it was purporting to make to justify the extension—and instead was relying on investments made in the ordinary course of business—the FCC has good cause to reconsider the extension that was previously granted.

ICLE recognizes that there may be good reason for companies to seek and obtain extensions of construction deadlines. However, the basis for which EchoStar sought the extensions and the process the FCC undertook to grant them do not give any comfort that EchoStar's history of spectrum warehousing will stop anytime soon. Indeed, even in the best case, if EchoStar complies

---

[9] These call signs were taken from EchoStar's September 2024 request for an extension. *See Letter from EchoStar Corp.*, WT Docket No. 22-212 (Sep. 18, 2024), https://www.fcc.gov/ecfs/document/1091867842711/1.

[10] *Comments Of EchoStar Corp.*, WT Docket No. 22-212 (May 27, 2025), https://www.fcc.gov/ecfs/document/105280597605377/1.

with each of its Commitments, the FCC (and the American people) have no guarantee that EchoStar will put the spectrum to use for another three years. Companies are starving for spectrum and the only viable course at the moment is acquisition through secondary markets. In light of the value of these assets to the American public, it is thus entirely appropriate for the Commission to reconsider its continued tolerance for EchoStar's warehousing practices—especially in light of what appear to be wildly exaggerated claims about what it has actually accomplished and what it committed to accomplish. Spectrum is simply too important to the U.S. economy and to U.S. security interests to be permitted remain unused in EchoStar's hands.

In the attached appendices we include the output of our code analysis, visualizations of coverage maps generated, and the Python source code that can be used to recreate this analysis for any set of BEAs in the United States.

Thank you for the opportunity to provide these reply comments.

Respectfully submitted,

Kristian Stout
Director of Innovation Policy
The International Center for Law & Economics

Appendix A – Full Output for Pittsburgh and Harrisburg

```
============================================================
EchoStar Coverage Comprehensive Analysis
============================================================
Loading population data...
Loaded 4217518 H3 population hexes (res 8)
Loaded 26 counties
Processed 315236 H3 hexes with population data

Creating H3 hex geometries for visualization...
Loaded 26 counties
Created geometries for 365097 H3 hexes
Visualizations will be saved to: [redacted]

Processing Pittsburgh, PA BEA...
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

4Q24 Analysis:
EchoStar 4Q24 POP Coverage for Pittsburgh, PA BEA:
        2845151 total POPs
        1691843 (59.46%) 5G 35/3 covered POPs
        209425 (7.36%) incremental 5G 7/1 covered POPs
        4337 (0.15%) incremental voice covered POPs
        1905606 (66.98%) total covered POPs

Visualizing 242330 hexes for Pittsburgh, PA BEA 4Q24 5G_35_3
Saved: Fig 1 - 4Q24 PittsburghPA 5G 35-3 Coverage.png
Visualizing 242330 hexes for Pittsburgh, PA BEA 4Q24 Any_Service
Saved: Fig 2 - 4Q24 PittsburghPA Any Coverage.png

2Q24 Analysis:
EchoStar 2Q24 POP Coverage for Pittsburgh, PA BEA:
        2845151 total POPs
        1559990 (54.83%) 5G 35/3 covered POPs
        228477 (8.03%) incremental 5G 7/1 covered POPs
        4186 (0.15%) incremental voice covered POPs
        1792654 (63.01%) total covered POPs

Visualizing 242330 hexes for Pittsburgh, PA BEA 2Q24 5G_35_3
Saved: Fig 3 - 2Q24 PittsburghPA 5G 35-3 Coverage.png
Visualizing 242330 hexes for Pittsburgh, PA BEA 2Q24 Any_Service
Saved: Fig 4 - 2Q24 PittsburghPA Any Coverage.png

2Q23 Analysis:
EchoStar 2Q23 POP Coverage for Pittsburgh, PA BEA:
        2845151 total POPs
        731598 (25.71%) 5G 35/3 covered POPs
        214185 (7.53%) incremental 5G 7/1 covered POPs
        3326 (0.12%) incremental voice covered POPs
        949108 (33.36%) total covered POPs
```

Visualizing 242330 hexes for Pittsburgh, PA BEA 2Q23 5G_35_3
Saved: Fig 5 - 2Q23 PittsburghPA 5G 35-3 Coverage.png
Visualizing 242330 hexes for Pittsburgh, PA BEA 2Q23 Any_Service
Saved: Fig 6 - 2Q23 PittsburghPA Any Coverage.png

Processing Harrisburg, PA BEA...
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

4Q24 Analysis:
EchoStar 4Q24 POP Coverage for Harrisburg, PA BEA:
1362347 total POPs
1176985 (86.39%) 5G 35/3 covered POPs
28130 (2.06%) incremental 5G 7/1 covered POPs
31558 (2.32%) incremental voice covered POPs
1236674 (90.78%) total covered POPs

Visualizing 86040 hexes for Harrisburg, PA BEA 4Q24 5G_35_3
Saved: Fig 7 - 4Q24 HarrisburgPA 5G 35-3 Coverage.png
Visualizing 86040 hexes for Harrisburg, PA BEA 4Q24 Any_Service
Saved: Fig 8 - 4Q24 HarrisburgPA Any Coverage.png

2Q24 Analysis:
EchoStar 2Q24 POP Coverage for Harrisburg, PA BEA:
1362347 total POPs
1173304 (86.12%) 5G 35/3 covered POPs
30595 (2.25%) incremental 5G 7/1 covered POPs
32037 (2.35%) incremental voice covered POPs
1235936 (90.72%) total covered POPs

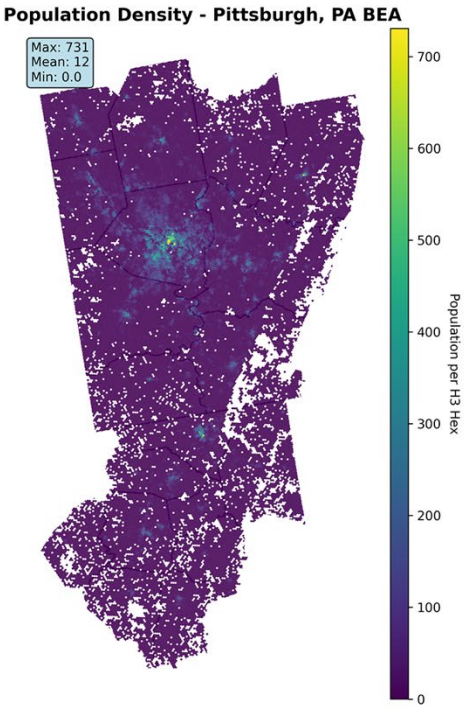
Visualizing 86040 hexes for Harrisburg, PA BEA 2Q24 5G_35_3
Saved: Fig 9 - 2Q24 HarrisburgPA 5G 35-3 Coverage.png
Visualizing 86040 hexes for Harrisburg, PA BEA 2Q24 Any_Service
Saved: Fig 10 - 2Q24 HarrisburgPA Any Coverage.png

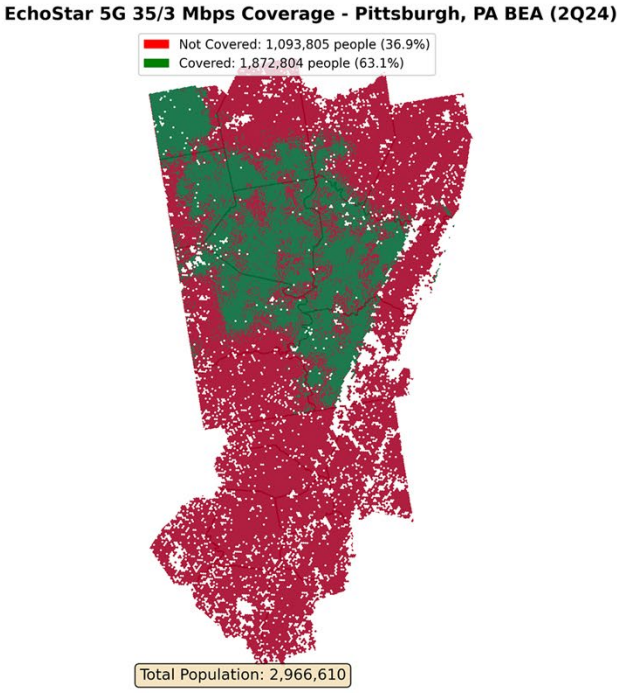
2Q23 Analysis:
EchoStar 2Q23 POP Coverage for Harrisburg, PA BEA:
1362347 total POPs
1166295 (85.61%) 5G 35/3 covered POPs
35735 (2.62%) incremental 5G 7/1 covered POPs
56347 (4.14%) incremental voice covered POPs
1258377 (92.37%) total covered POPs

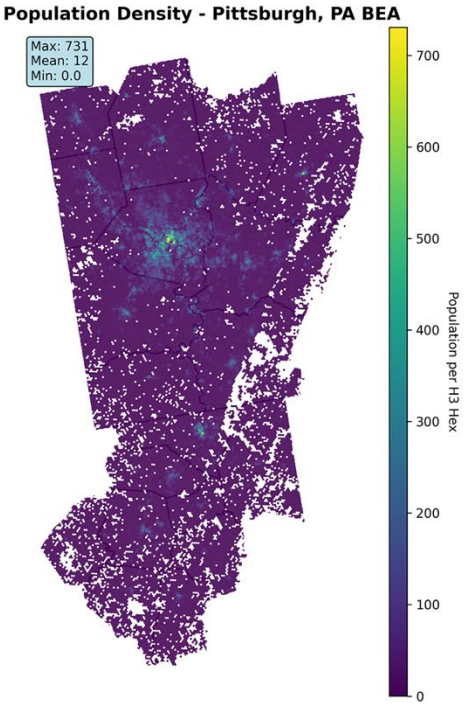
Visualizing 86040 hexes for Harrisburg, PA BEA 2Q23 5G_35_3
Saved: Fig 11 - 2Q23 HarrisburgPA 5G 35-3 Coverage.png
Visualizing 86040 hexes for Harrisburg, PA BEA 2Q23 Any_Service
Saved: Fig 12 - 2Q23 HarrisburgPA Any Coverage.png
==
Analysis Complete!
All visualizations saved to: [redacted]
==
~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~~

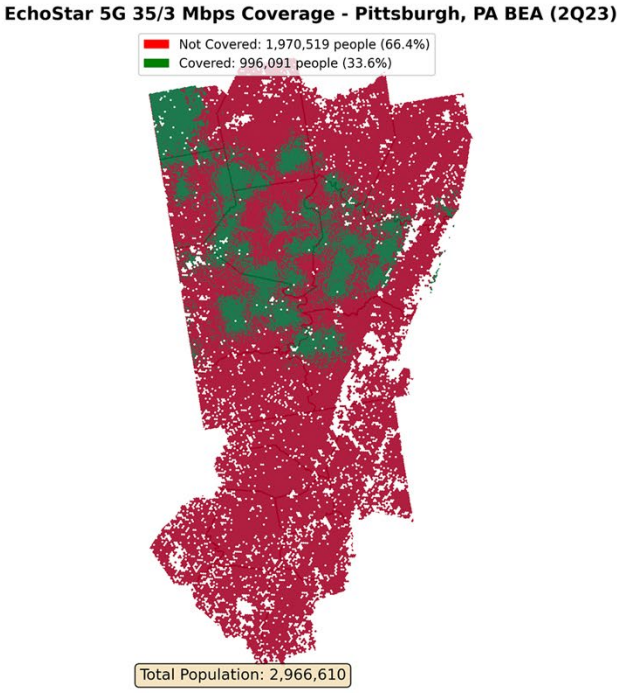
Appendix B – Visualizations of Coverage

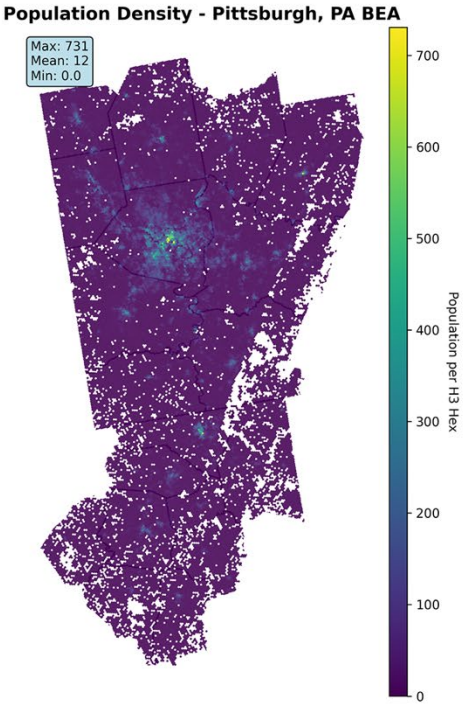
The following visualizations were generated for the above coverage data.

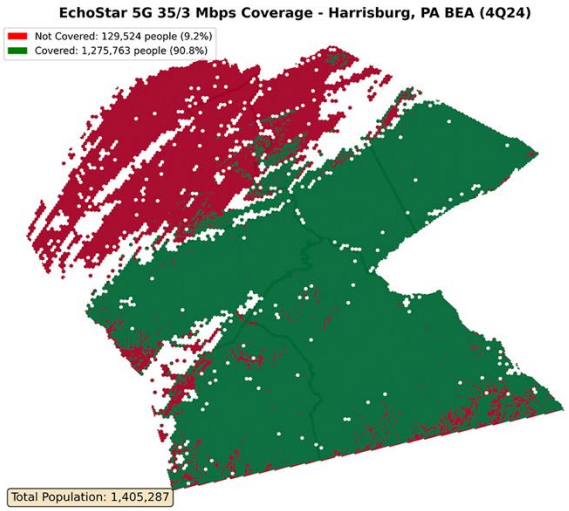


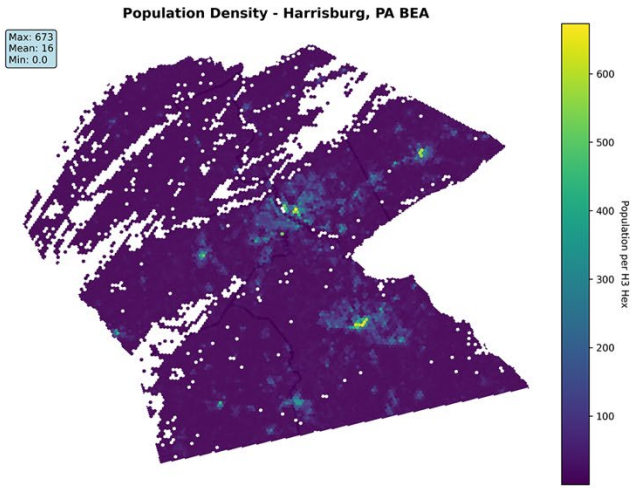

EchoStar 5G 35/3 Mbps Coverage - Pittsburgh, PA BEA (2Q24)
Not Covered: 1,093,805 people (36.9%)
Covered: 1,872,804 people (63.1%)
Total Population: 2,966,610

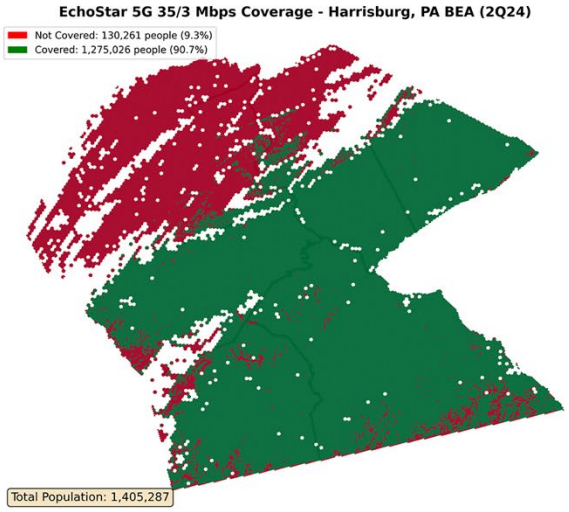

Population Density - Pittsburgh, PA BEA
Max: 731
Mean: 12
Min: 0.0
Population per H3 Hex
0
100
200
300
400
500
600
700

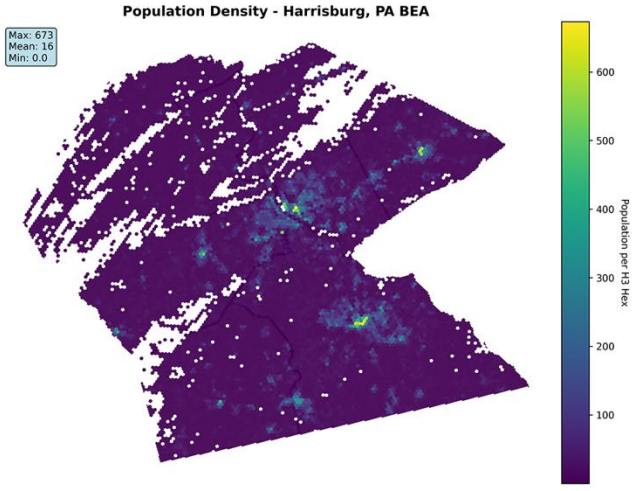

EchoStar 5G 35/3 Mbps Coverage - Pittsburgh, PA BEA (2Q23)
Not Covered: 1,970,519 people (66.4%)
Covered: 996,091 people (33.6%)
Total Population: 2,966,610

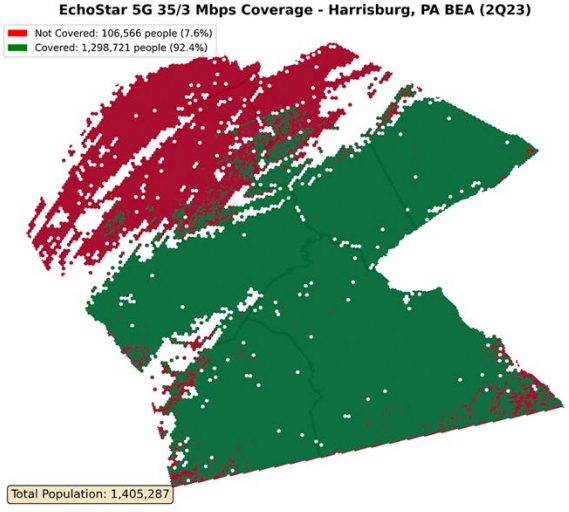

Population Density - Pittsburgh, PA BEA
Max: 731
Mean: 12
Min: 0.0
Population per H3 Hex
0
100
200
300
400
500
600
700

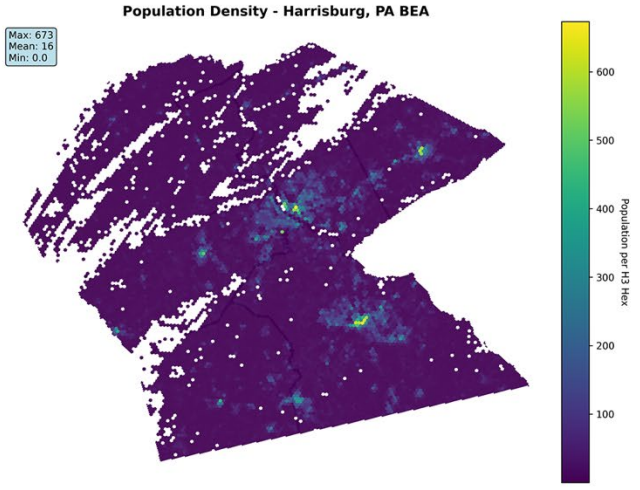

EchoStar 5G 35/3 Mbps Coverage - Harrisburg, PA BEA (4Q24)
Not Covered: 129,524 people (9.2%)
Covered: 1,275,763 people (90.8%)
Total Population: 1,405,287


Population Density - Harrisburg, PA BEA
Max: 673
Mean: 16
Min: 0.0
Population per H3 Hex
600
500
400
300
200
100


EchoStar 5G 35/3 Mbps Coverage - Harrisburg, PA BEA (2Q24)
Not Covered: 130,261 people (9.3%)
Covered: 1,275,026 people (90.7%)
Total Population: 1,405,287


Population Density - Harrisburg, PA BEA
Max: 673
Mean: 16
Min: 0.0
Population per H3 Hex
600
500
400
300
200
100


EchoStar 5G 35/3 Mbps Coverage - Harrisburg, PA BEA (2Q23)
Not Covered: 106,566 people (7.6%)
Covered: 1,298,721 people (92.4%)
Total Population: 1,405,287


Population Density - Harrisburg, PA BEA
Max: 673
Mean: 16
Min: 0.0
Population per H3 Hex
600
500
400
300
200
100

Appendix C – Analysis Code (Python)

```
"""
EchoStar Coverage Comprehensive Analysis
- Processes all markets and timeframes
- Generates visualizations for each combination
- Prints summary reports
"""
#To reproduce, grab the following data sets:
# Download and unzip into tl_2020_us_county directory US County boundaries from Census Bureau:
# https://www.census.gov/geographies/mapping-files/time-series/geo/tiger-line-file.2020.html

# Download and unzip H3 5G-NR (35/3 Mbps), 5G-NR (7/1 Mbps) and Mobile Voice EchoStar gpkg 2Q23, 2Q24 and 4Q24 coverage data for PA from:
# https://broadbandmap.fcc.gov/data-download/data-by-provider

# Download Kontur H3 population dataset from:
# https://data.humdata.org/dataset/kontur-population-united-states-of-america

# Group some fixed data by region and time for looping
# FIPS codes for counties by BEA from h
https://transition.fcc.gov/bureaus/oet/info/maps/areas/cnty1990/beacnty1990.txt

import pandas as pd
import geopandas as gpd
import h3pandas as h3
import h3 as h3_lib
import matplotlib.pyplot as plt
import os
from typing import List, Dict, Tuple

# Setup
temp_dir = '/Path/to/data/files'

# Market definitions
mkt_dict = {
    'pittsburgh': {
        'name': 'Pittsburgh, PA BEA',
        'fips_list': ['42003', '42005', '42007', '42019', '42051', '42059', '42063', '42073',
            '42125', '42129', '54001', '54017', '54033', '54041', '54049', '54061', '54077',
            '54091', '54097']},
    'harrisburg': {
        'name': 'Harrisburg, PA BEA',
        'fips_list': ['42001', '42041', '42043', '42067', '42075', '42099', '42133']}}

timeframe_dict = {
    '4Q24': {
        'name': '4Q24',
        'suffix': '_h3_D24_12may2025.gpkg'},
    '2Q24': {
        'name': '2Q24',
        'suffix': '_h3_J24_08nov2024.gpkg'},
    '2Q23': {
```

```
        'name': '2Q23',
        'suffix': '_h3_J23_05may2024.gpkg'}}

# Constants
PA_FIPS = '42'
H3_RES = 9
NR5G353 = 1
NR5G71 = 2
VOICE = 4

# Data loading functions
def read_h3_pops(file_path: str) -> pd.DataFrame:
    """Load H3 population data"""
    temp = gpd.read_file(file_path)
    print(f"Loaded {len(temp)} H3 population hexes (res 8)")
    return temp[['h3', 'population']].rename(columns={'h3': 'h3_parent'})

def read_cty_data(file_path: str, fips_list: List[str]) -> gpd.GeoDataFrame:
    """Load and filter county boundary data"""
    temp = gpd.read_file(file_path)
    temp['fips'] = temp['STATEFP'] + temp['COUNTYFP']
    temp = temp[temp['fips'].isin(fips_list)]
    print(f"Loaded {len(temp)} counties")
    return temp[['fips', 'geometry']]

def get_pop_by_county(county_list: List[str]) -> pd.DataFrame:
    """Create dataframe of H3 res9 IDs with prorated population by county"""

    # Read data
    h3_pops = read_h3_pops(os.path.join(temp_dir, 'kontur_population_US_20231101.gpkg'))
    county_gdf = read_cty_data(os.path.join(temp_dir, 'tl_2020_us_county', 'tl_2020_us_county.shp'), county_list)

    # Convert to Albers Equal Area for accurate area calculations
    county_gdf = county_gdf.to_crs('ESRI:102003')

    # Buffer counties to ensure complete coverage
    buffered_counties = county_gdf.copy()
    buffered_counties['geometry'] = county_gdf['geometry'].buffer(250)  # 250m > 174m H3 res 9 edge

    # Convert back to WGS84 for H3 operations
    buffered_counties = buffered_counties.to_crs('EPSG:4326')

    # Fill counties with H3 hexes
    h3_counties = buffered_counties.h3.polyfill_resample(H3_RES).reset_index().rename(
        columns={'h3_polyfill': 'h3_res9_id'}).drop(columns={'index'})

    # Back to Albers for area calculations
    h3_counties = h3_counties.to_crs('ESRI:102003')
    h3_counties['h3area'] = h3_counties['geometry'].area

    # Clip to county boundaries and calculate proration
    intersect = h3_counties.overlay(county_gdf[['geometry']], how='intersection').drop_duplicates()
    intersect['subarea'] = intersect['geometry'].area
```

```
    intersect['proration'] = intersect['subarea'] / intersect['h3area']
    intersect = intersect.groupby(['h3_res9_id', 'fips']).agg({'proration': 'sum'}).reset_index()

    # Get H3 res 8 parent for each res 9 hex
    intersect = intersect.set_index('h3_res9_id').h3.h3_to_parent().reset_index()

    # Merge with population data and calculate prorated population
    return_df = pd.merge(intersect, h3_pops, how='inner', on='h3_parent')
    return_df['pops'] = (return_df['population'] / 7) * return_df['proration']
    return_df = return_df.drop(columns=['h3_parent', 'population', 'proration'])

    return return_df[return_df['pops'] > 0]

def read_coverage(coverage_file: str, service: int, bea: pd.DataFrame) -> pd.DataFrame:
    """Read coverage data and merge with population data"""
    try:
        temp = gpd.read_file(os.path.join(temp_dir, coverage_file))
        temp = temp[['h3_res9_id']]
        temp['type'] = service
        return_df = pd.merge(bea, temp, how='left', on='h3_res9_id')
        return_df['type'] = return_df['type'].fillna(0.0)
        return return_df[return_df['pops'] > 0]
    except Exception as e:
        print(f"Warning: Could not load {coverage_file}: {e}")
        return bea.copy()

def process_coverage(suffix: str, bea_fips: List[str], pop_data: pd.DataFrame) -> List[float]:
    """Process coverage data for all service types"""
    market = pop_data[pop_data['fips'].isin(bea_fips)]

    # 5G 35/3 coverage
    coverage = read_coverage(f'bdc_42_460677_5GNR_35_3_mobile_broadband{suffix}', NR5G353, market)
    echostar_5g353_pops = coverage[coverage['type'] == NR5G353]['pops'].sum()

    # 5G 7/1 coverage (incremental)
    temp = read_coverage(f'bdc_42_460677_5GNR_7_1_mobile_broadband{suffix}', NR5G71, market)
    coverage = pd.concat([coverage, temp])
    coverage = coverage.groupby(['h3_res9_id', 'fips', 'pops'])['type'].sum().reset_index()
    echostar_5g71_incremental_pops = coverage[coverage['type'] == NR5G71]['pops'].sum()

    # Voice coverage (incremental)
    temp = read_coverage(f'bdc_42_460677_MobileVoice_mobile_voice{suffix}', VOICE, market)
    coverage = pd.concat([coverage, temp])
    coverage = coverage.groupby(['h3_res9_id', 'fips', 'pops'])['type'].sum().reset_index()
    echostar_voice_incremental_pops = coverage[coverage['type'] == VOICE]['pops'].sum()

    return [echostar_5g353_pops, echostar_5g71_incremental_pops, echostar_voice_incremental_pops], coverage

def show_output(caption: str, data_list: List[float], denominator: float):
    """Print formatted coverage statistics"""
    print(caption)
    print(f'\t{denominator:.0f} total POPs')
    pct_5g353 = (data_list[0]/denominator) * 100
```

```
    print(f'\t{data_list[0]:.0f} ({pct_5g353:.2f}%) 5G 35/3 covered POPs')
    pct_5g71 = (data_list[1]/denominator) * 100
    print(f'\t{data_list[1]:.0f} ({pct_5g71:.2f}%) incremental 5G 7/1 covered POPs')
    pct_voice = (data_list[2]/denominator) * 100
    print(f'\t{data_list[2]:.0f} ({pct_voice:.2f}%) incremental voice covered POPs')
    any_coverage = data_list[0] + data_list[1] + data_list[2]
    pct = (any_coverage/denominator) * 100
    print(f'\t{any_coverage:.0f} ({pct:.2f}%) total covered POPs')
    print('\n')

def create_hex_geometries_from_counties(county_list: List[str]) -> gpd.GeoDataFrame:
    """Create H3 hex geometries using the same method as the working v3 script"""

    # Load county data
    county_gdf = read_cty_data(os.path.join(temp_dir, 'tl_2020_us_county', 'tl_2020_us_county.shp'), county_list)

    # Convert to Albers Equal Area for buffering
    county_gdf = county_gdf.to_crs('ESRI:102003')

    # Buffer counties to ensure complete coverage
    buffered_counties = county_gdf.copy()
    buffered_counties['geometry'] = county_gdf['geometry'].buffer(250)

    # Convert back to WGS84 for H3 operations
    buffered_counties = buffered_counties.to_crs('EPSG:4326')

    # Fill counties with H3 hexes - this creates the geometries
    h3_counties = buffered_counties.h3.polyfill_resample(H3_RES).reset_index().rename(
        columns={'h3_polyfill': 'h3_res9_id'}).drop(columns={'index'})

    print(f"Created geometries for {len(h3_counties)} H3 hexes")

    # Convert back to projected CRS for visualization
    return h3_counties.to_crs('ESRI:102003')

def visualize_coverage(coverage_data: pd.DataFrame, market_name: str, timeframe: str,
                       service_type: str, hex_geometries: gpd.GeoDataFrame, figure_number: int):
    """Create visualization of coverage data"""

    # Merge coverage data with hex geometries
    viz_data = pd.merge(hex_geometries, coverage_data, on='h3_res9_id', how='inner')

    if len(viz_data) == 0:
        print(f"No data to visualize for {market_name} {timeframe} {service_type}")
        return

    print(f"Visualizing {len(viz_data)} hexes for {market_name} {timeframe} {service_type}")

    # Create coverage indicator and titles
    if service_type == '5G_35_3':
        viz_data['covered'] = (viz_data['type'] >= NR5G353).astype(int)
        title = f'EchoStar 5G 35/3 Mbps Coverage - {market_name} ({timeframe})'
        service_label = '5G 35-3 Coverage'
```

```
elif service_type == '5G_7_1':
    viz_data['covered'] = (viz_data['type'] >= NR5G71).astype(int)
    title = f'EchoStar 5G 7/1+ Mbps Coverage - {market_name} ({timeframe})'
    service_label = '5G 7-1 Coverage'
elif service_type == 'Voice':
    viz_data['covered'] = (viz_data['type'] >= VOICE).astype(int)
    title = f'EchoStar Voice Coverage - {market_name} ({timeframe})'
    service_label = 'Voice Coverage'
else:
    viz_data['covered'] = (viz_data['type'] > 0).astype(int)
    title = f'EchoStar Any Coverage - {market_name} ({timeframe})'
    service_label = 'Any Coverage'

# Create visualization
fig, (ax1, ax2) = plt.subplots(1, 2, figsize=(18, 8))

# Calculate coverage statistics
covered_pop = viz_data[viz_data['covered'] == 1]['pops'].sum()
uncovered_pop = viz_data[viz_data['covered'] == 0]['pops'].sum()
total_pop = covered_pop + uncovered_pop
covered_pct = (covered_pop / total_pop * 100) if total_pop > 0 else 0
uncovered_pct = (uncovered_pop / total_pop * 100) if total_pop > 0 else 0

# Coverage map with custom legend
covered_plot = viz_data.plot(column='covered', cmap='RdYlGn', ax=ax1, legend=False)
ax1.set_title(title, fontsize=14, fontweight='bold')
ax1.axis('off')

# Add custom legend for coverage with population stats
from matplotlib.patches import Patch
legend_elements = [
    Patch(facecolor='red', label=f'Not Covered: {uncovered_pop:,.0f} people ({uncovered_pct:.1f}%)'),
    Patch(facecolor='green', label=f'Covered: {covered_pop:,.0f} people ({covered_pct:.1f}%)')
]
ax1.legend(handles=legend_elements, loc='upper left', bbox_to_anchor=(0, 1))

# Add total population text box
textstr = f'Total Population: {total_pop:,.0f}'
props = dict(boxstyle='round', facecolor='wheat', alpha=0.8)
ax1.text(0.02, 0.02, textstr, transform=ax1.transAxes, fontsize=12,
        verticalalignment='bottom', bbox=props)

# Population density map with colorbar
pop_plot = viz_data.plot(column='pops', cmap='viridis', ax=ax2, legend=False)
ax2.set_title(f'Population Density - {market_name}', fontsize=14, fontweight='bold')
ax2.axis('off')

# Add colorbar for population
from mpl_toolkits.axes_grid1 import make_axes_locatable

divider = make_axes_locatable(ax2)
cax = divider.append_axes("right", size="5%", pad=0.1)
```

```
    # Create colorbar with population values
    sm = plt.cm.ScalarMappable(cmap='viridis', norm=plt.Normalize(vmin=viz_data['pops'].min(),
vmax=viz_data['pops'].max()))
    sm.set_array([])
    cbar = plt.colorbar(sm, cax=cax)
    cbar.set_label('Population per H3 Hex', rotation=270, labelpad=15)

    # Add population density statistics to the right plot
    pop_stats = f'Max: {viz_data["pops"].max():.0f}\nMean: {viz_data["pops"].mean():.0f}\nMin:
{viz_data["pops"].min():.1f}'
    ax2.text(0.02, 0.98, pop_stats, transform=ax2.transAxes, fontsize=10,
             verticalalignment='top', bbox=dict(boxstyle='round', facecolor='lightblue', alpha=0.8))

    plt.tight_layout()

    # Create viz directory if it doesn't exist
    viz_dir = os.path.join(temp_dir, '..', 'viz')
    os.makedirs(viz_dir, exist_ok=True)

    # Create filename with figure number
    market_clean = market_name.replace(',', '').replace(' BEA', '').replace(' ', '')
    filename = f"Fig {figure_number} - {timeframe} {market_clean} {service_label}.png"
    filepath = os.path.join(viz_dir, filename)

    plt.savefig(filepath, dpi=300, bbox_inches='tight')
    print(f"Saved: {filename}")

    plt.show()

def main():
    """Main analysis function"""
    print("=" * 60)
    print("EchoStar Coverage Comprehensive Analysis")
    print("=" * 60)

    # Load population data for all counties
    all_fips = mkt_dict['pittsburgh']['fips_list'] + mkt_dict['harrisburg']['fips_list']
    print("Loading population data...")
    county_pops_by_h3 = get_pop_by_county(all_fips)
    print(f"Processed {len(county_pops_by_h3)} H3 hexes with population data\n")

    # Create hex geometries for visualization (once for all markets)
    print("Creating H3 hex geometries for visualization...")
    hex_geometries = create_hex_geometries_from_counties(all_fips)

    # Create viz directory
    viz_dir = os.path.join(temp_dir, '..', 'viz')
    os.makedirs(viz_dir, exist_ok=True)
    print(f"Visualizations will be saved to: {viz_dir}\n")

    figure_number = 1

    # Process each market and timeframe
```

```
    for mkt in ['pittsburgh', 'harrisburg']:
        market_name = mkt_dict[mkt]['name']
        market_fips = mkt_dict[mkt]['fips_list']
        mkt_pops = county_pops_by_h3[county_pops_by_h3['fips'].isin(market_fips)]['pops'].sum()

        print(f"\nProcessing {market_name}...")
        print("-" * 50)

        # Get hex geometries for this market
        market_hexes = hex_geometries[hex_geometries['fips'].isin(market_fips)]

        for timeframe in ['4Q24', '2Q24', '2Q23']:
            timeframe_name = timeframe_dict[timeframe]['name']
            suffix = timeframe_dict[timeframe]['suffix']

            print(f"\n{timeframe_name} Analysis:")

            # Process coverage data
            coverage_stats, coverage_data = process_coverage(suffix, market_fips, county_pops_by_h3)

            # Show summary report
            title = f'EchoStar {timeframe_name} POP Coverage for {market_name}:'
            show_output(title, coverage_stats, mkt_pops)

            # Create visualizations for different service types
            market_data = coverage_data[coverage_data['fips'].isin(market_fips)]

            if len(market_data) > 0:
                # 5G 35/3 visualization
                visualize_coverage(market_data, market_name,
                                   timeframe_name, '5G_35_3', market_hexes, figure_number)
                figure_number += 1

                # Combined coverage visualization (any service)
                visualize_coverage(market_data, market_name,
                                   timeframe_name, 'Any_Service', market_hexes, figure_number)
                figure_number += 1

    print("=" * 60)
    print("Analysis Complete!")
    print(f"All visualizations saved to: {viz_dir}")
    print("=" * 60)

if __name__ == "__main__":
    main()
```