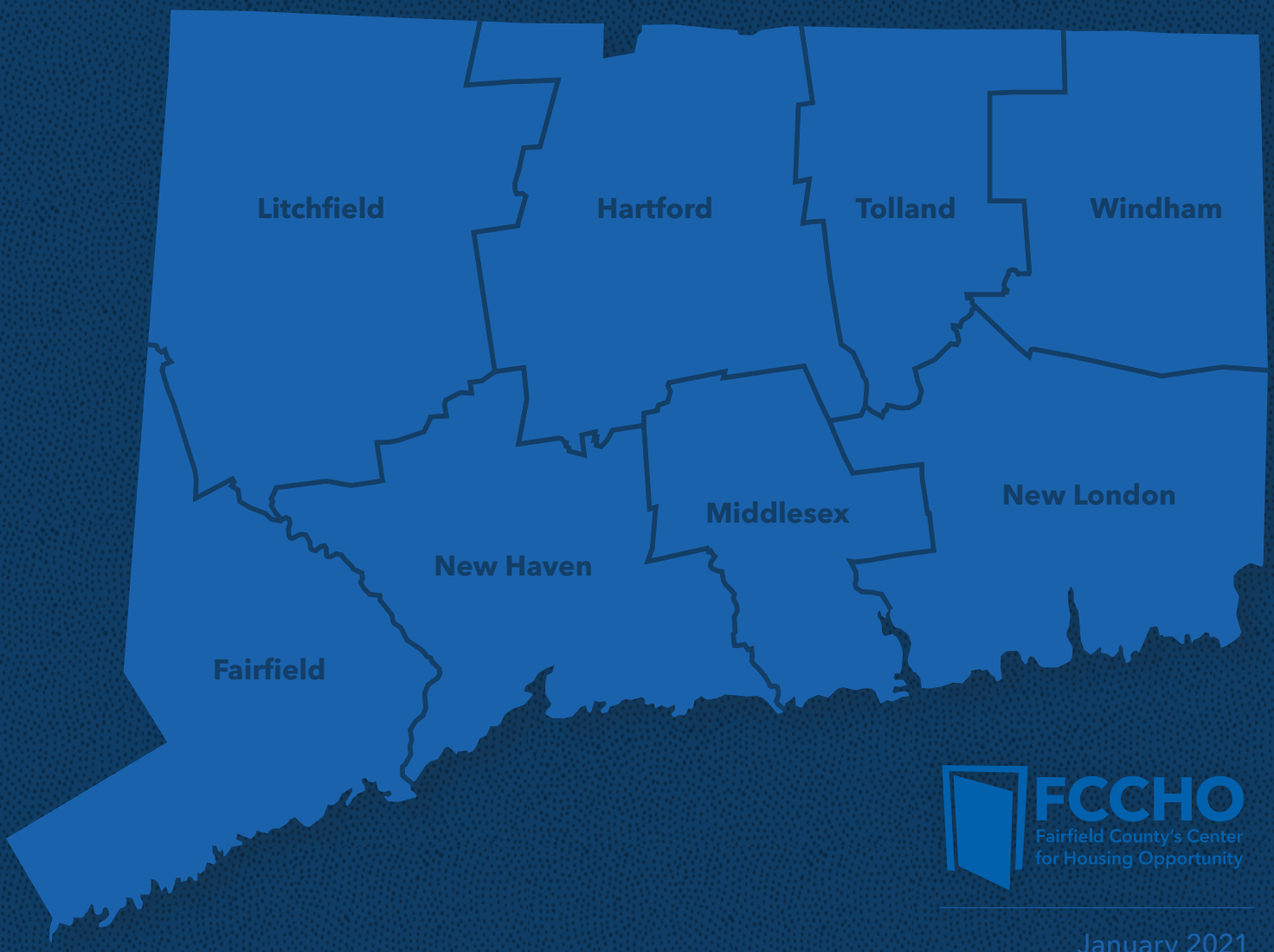
AffordCT

Housing Database User Guide



FCCHO
Fairfield County's Center for Housing Opportunity

January 2021

FCCHO
Fairfield County's Center for Housing Opportunity

Fairfield County's Center for Housing Opportunity (FCCHO) facilitates the intentional production, preservation, and protection of a full spectrum of housing that fosters communities of opportunity for all Fairfield County residents. A strategic partnership between Fairfield County's Community Foundation, Partnership for Strong Communities, Regional Plan Association and Supportive Housing Works, FCCHO utilizes a collaborative, data-driven framework, aligning regional resources to deliver impactful systems change and equitable housing solutions.

Acknowledgements

Fairfield County's Center for Housing Opportunity thanks the many organizations and individuals who contributed to the creation of this critical resource.

FUNDERS

Apple Pickers Foundation
Connecticut Department of Housing
Connecticut Department of Social Services
Fairfield County's Community Foundation
JP Morgan Chase

ENGINEERS:

Source Development Hub
- Billy Huang, CEO
- Nelson Lau
- Michelle Jones

DATA SOURCES:

American Community Survey / IPUMS- USA
Connecticut Department of Housing
Connecticut Department of Social Services
Connecticut Housing Finance Authority
Corporation for Supportive Housing
Data Haven
Housing & Urban Development Housing Inventory Count
Public & Affordable Housing Research Corporation
Urban Institute

FAIRFIELD COUNTY HOUSING ALLIANCE DATA TEAM

Kara Capone
Adhlere Coffy
Jenita Hayes
Billy Huang
John Warburg
Lauren Zimmerman

Special Thanks

Christopher Brechlin, Senior Program & Data Analyst, Connecticut Housing Finance Authority

Erin Boggs, Executive Director, Open Communities Alliance

Jonathan Cabral, Interim Director -Planning, Research & Evaluation, Connecticut Housing Finance Authority

Ellis Calvin, Regional Plan Association Data Research Manager

Adhlere Coffy, Director of Strategic Initiatives, Dalio Philanthropies

Finnuala Darby-Hudgens, Director of Operations, Connecticut Fair Housing Center

Kelly Davila, Data Haven

Steve DiLella, Director of Individual & Family Support Programs Connecticut Department of Housing

Danielle Dobin, Town of Westport Planning & Zoning

Moses Gates, Vice President for Housing & Neighborhood Planning, Regional Plan Association

Sean Ghio, Policy Director, Partnership for Strong Communities

James Horan, Executive Director, Local Initiative Support Corporation -CT

Alanna Kabel, CPD Director, Hartford, US Department of Housing & Urban Development

Melissa Kaplan Macey, Vice President of State Programs & Connecticut Director, Regional Plan Association

Monique King-Viehland, director of State & Local Housing Policy, Urban Institute

Dara Kovel, Chief Executive Officer, Beacon Communities

Alyssa Languth, Corporation for Supportive Housing

Lydia Lo, Research Analyst, Metropolitan Housing & Communities Policy Center , Urban Institute

Steven Martin, Senior Research Associate in the Center on Labor, Human Services & Population, Urban Institute

Kelly McElwain, Research Analyst, Public & Affordable Housing Research Corporation

Mark McNulty, Communications Associate, Regional Plan Association

Terry Nash, Community Engagement Manager, Connecticut Housing Finance Authority

Suzanne Piacentini, Field Office Director, Hartford, US Department Housing & Urban Development

David Rich, Executive Director, Supportive Housing Works

Jeff Rieck, Executive Director, Danbury Housing Authority

Carmen Rodriguez, Management Analyst, Office of Field Policy & Management, Hartford, US Department of Housing & Urban Development

Yasmmyn Salinas, Assistant Professor, Yale School of Public Health

Michael Santoro, Director, Office of Policy, Research & Housing Support, CT Department of Housing

Keely Stater, Director, Research & Industry Intelligence, Public & Affordable Housing Research Corporation

Kim Stevenson, Director of Strategic Initiatives, Inspire Prosperity Capital

Peter Tatian, Senior Fellow, Urban Institute

Jack Tsai, Professor & Campus Dean, UT Health School of Public Health

Fay Walker, Research Analyst, Metropolitan Housing & Communities Policy Center, Urban Institute

John Warburg, Principal, Apple Pickers Foundation

Laura Watson, Office of Policy, Research & Housing Support, CT Department of Housing

Carla Weil, Director of Commercial Lending, Capital for Change

Dave Zackin, Graphic Designer, Regional Plan Association

AffordCT Housing Database User Guide

Introduction

Since its launch in 2019, FCCHO has recognized the need for an aggregated online inventory of affordable housing units throughout the state as a means of identifying and aligning regional and statewide housing goals and resources and facilitating shared accountability among housing practitioners, policy-makers, funders, and advocates.

As part of our efforts to deliver tools and resources that support the data-driven production, protection, and preservation of affordable housing, FCCHO leveraged private and public funding and assembled a project team led by engineering partner Source Development Hub to deliver this statewide online inventory of assisted housing units.

An open source, online platform for the state's current affordable housing data is critical to ensuring (1) a fluid, shared understanding of Connecticut's low-moderate income housing needs and how to meet them; and (2) measuring Connecticut's collective progress towards meeting those needs.

It is our hope that this new tool provides policy-makers and affordable housing practitioners alike, a means to make targeted decisions about project siting and funding, and the ability to more strategically deploy housing resources throughout the state.

Finally, the development of this tool remains an iterative process which we will continue to refine and enhance as additional data become available. Your feedback will assist us in ensuring all users derive as much value as possible from this platform.

This user guide is a tutorial for AffordCT, the data dashboard and file sharing platform hosted on **affordablehousing.tools**. It consists of two sections: (1) a guide on uploading and managing datasets, and (2) a guide on usage of the associated dashboard.

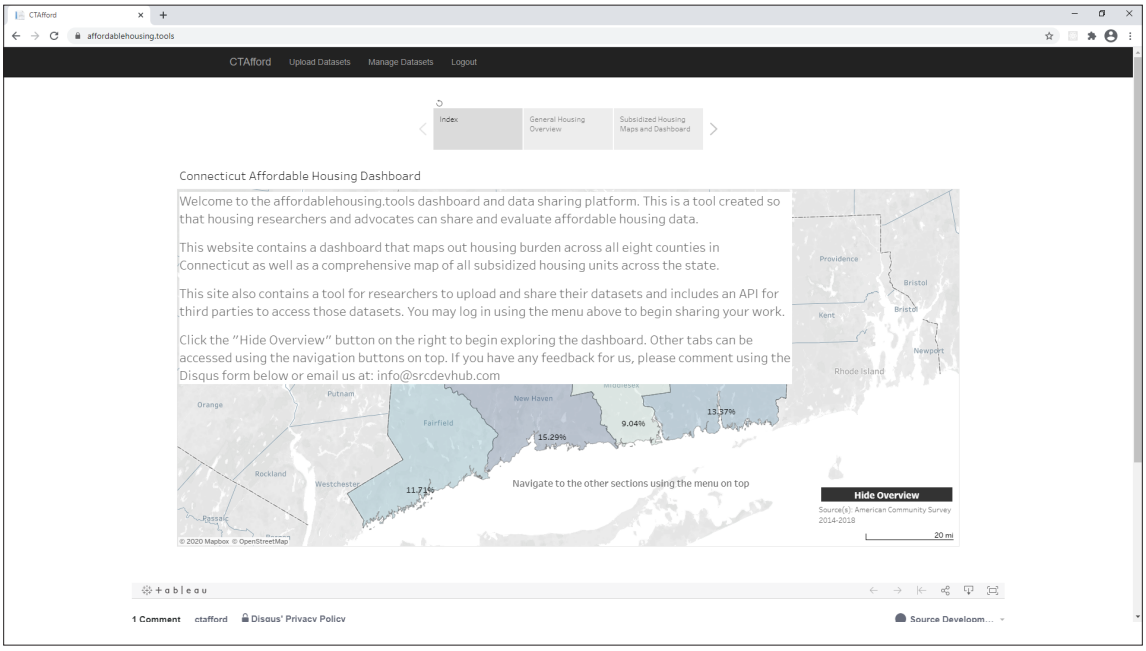
The first section provides details on navigating the affordable housing dashboard. This is based off work completed for the Connecticut Department of Housing's 2020 Study of Affordable and Accessible Housing.

The second section provides details for how to use the file upload tool and how to manage your uploaded datasets for sharing with other users. Once the datasets are uploaded, we ask that your routinely check on them to make sure that they have been approved and available for sharing. This section also provides a table of available datasets that users can access through our API as well as details on how to write code to access the API.

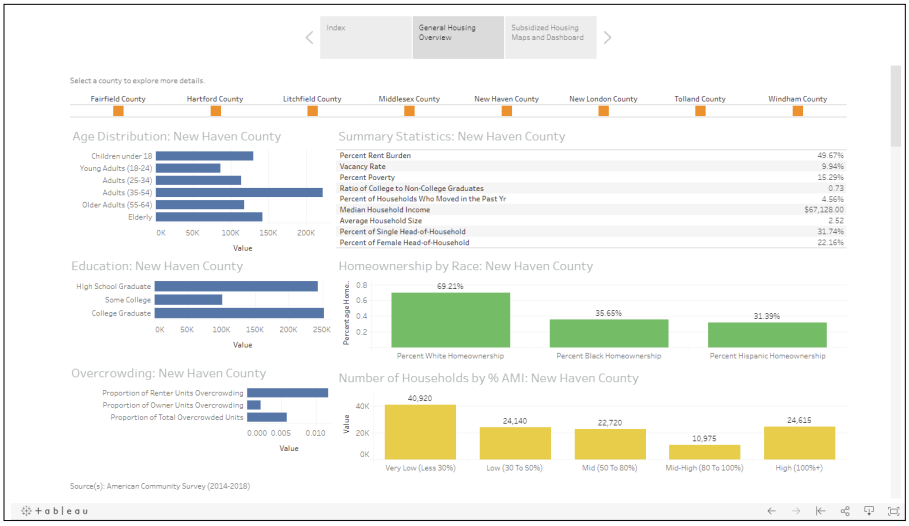
If you have any questions, please address them to **info@srcdevhub.com**.

Sincerely,
The Source Development Hub Team
v1, January 2021

Section 1: Dashboard Navigation



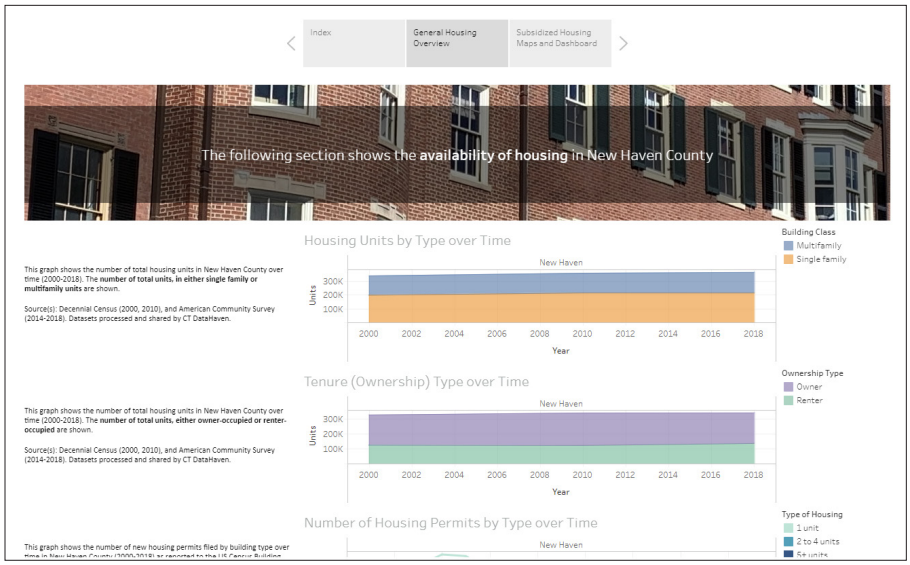
When the website is loaded, the first tab loaded is a map with a dropdown menu of several select housing indicators. An introduction to this site is automatically loaded and can be toggled using the button on the right. Navigate to the other pages of this dashboard using the bar on top.

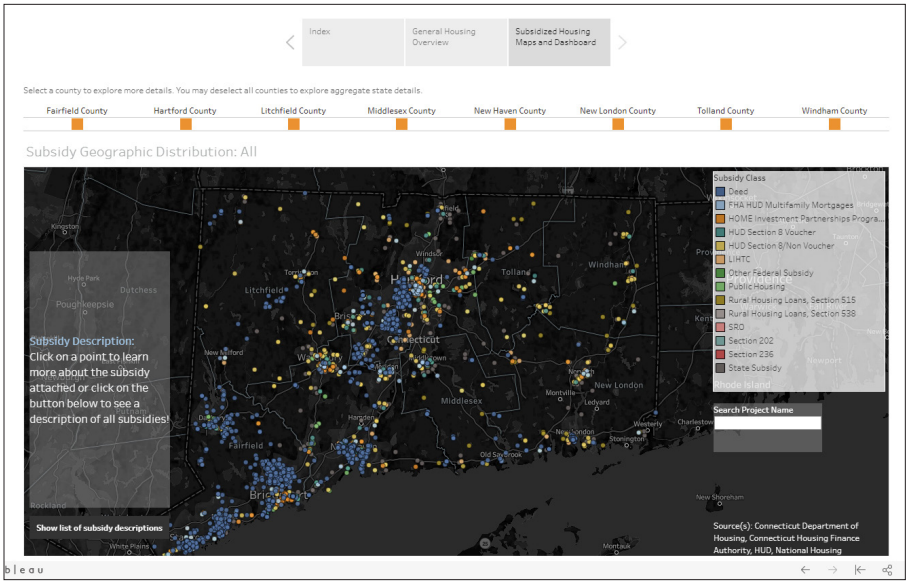


The second tab of the dashboard shows a list of general housing indicators. The top section shows a range of indicators. Use the bar on top to select statistics for a given county. The other graphs tables below will filter accordingly.

Scrolling downward, the next section describes housing cost burden by income, race, and disability status.

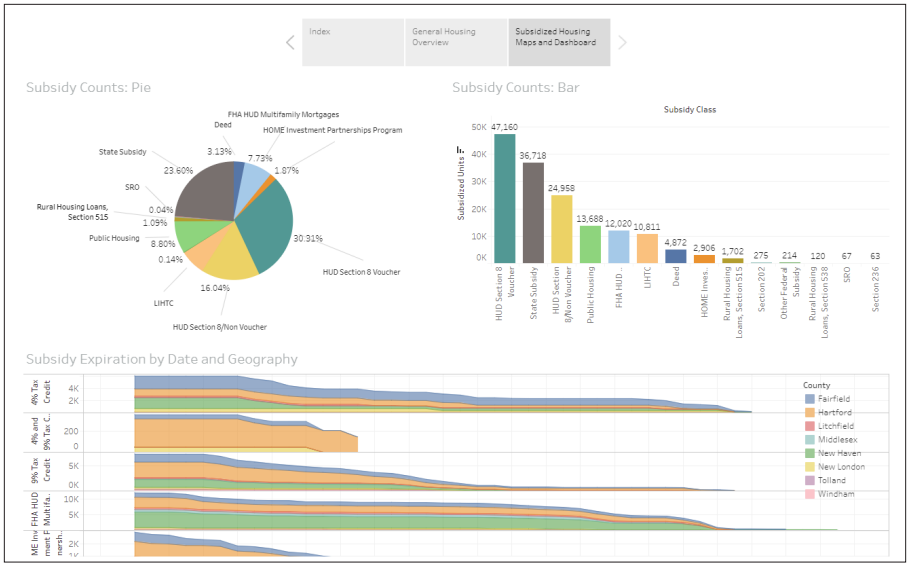
Scrolling downward, the final section describes the supply of housing and the gap in supply and demand by income band.

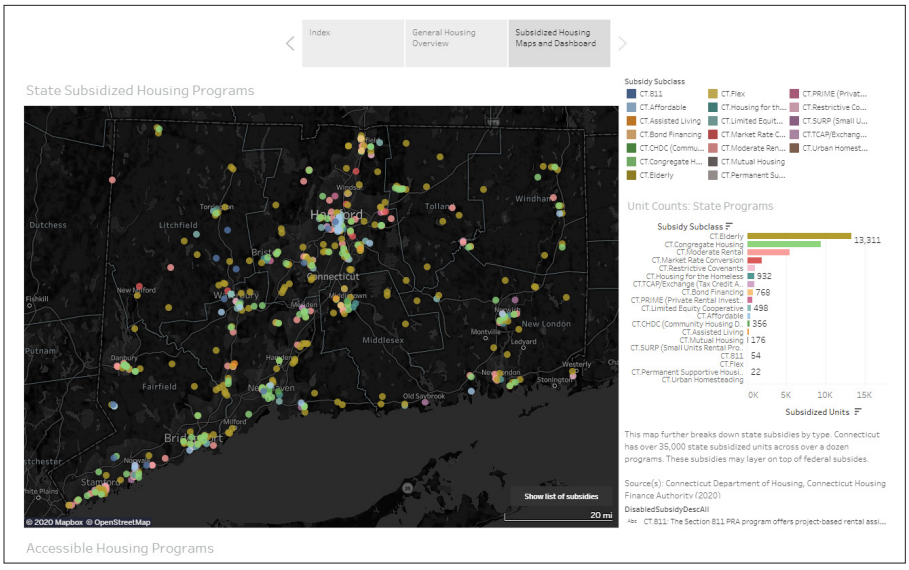




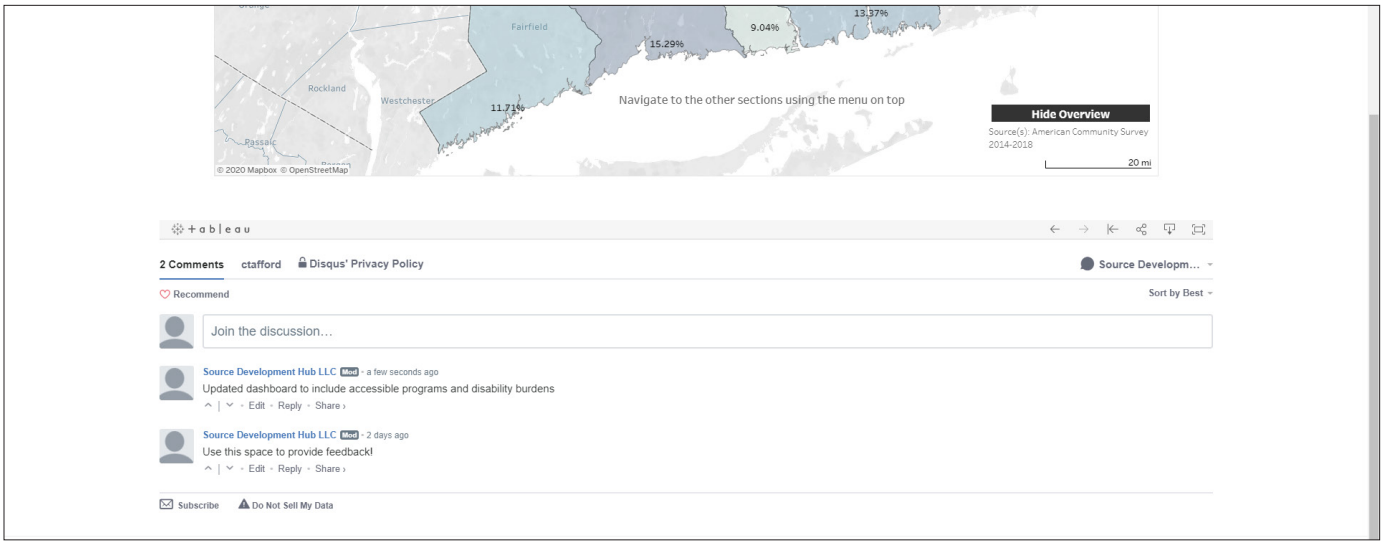
The third tab of the dashboard shows several maps and charts of subsidized housing. The top section shows a range of indicators. Use the bar on top to select statistics for a given county. The other graphs and maps below will filter accordingly. The first map shows list of all subsidy classes, grouped into either federal or state subsidies. You can search for project names or highlight a subsidy type through the legend to the right. There is a toggle on the left for more information about each subsidy.

Scrolling downward, there are a few visualizations summing up the number of subsidized units and their expiration dates.





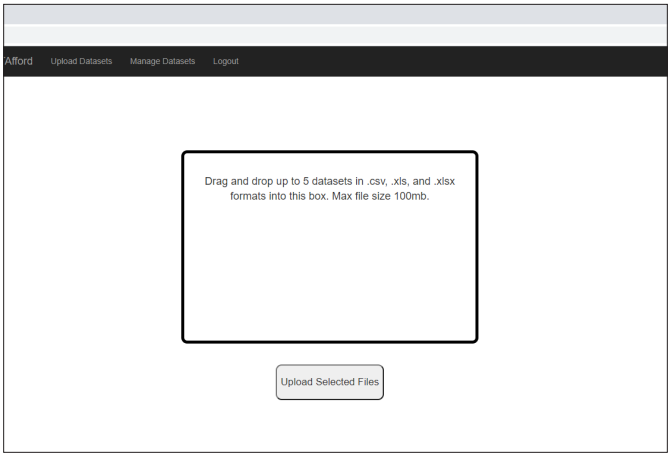
There are additional maps breaking down the subsidy programs by different classes. The example illustrated here is a further breakdown of state subsidies. On every map there is a button toggle for more information about the subsidy programs.



Questions/Feedback? Use the Disqus form at the bottom of every page.

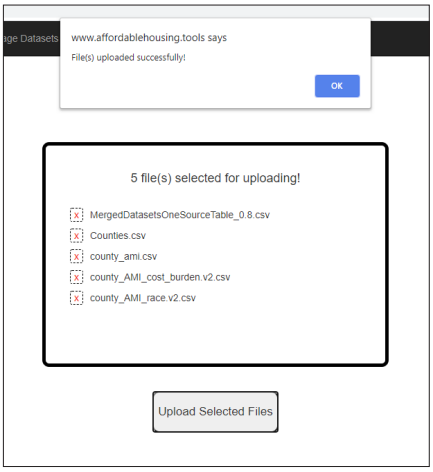
Section 2: Data Sharing Platform and API Access

Login to upload a dataset from the following URL:
https://www.affordablehousing.tools/Account/Login



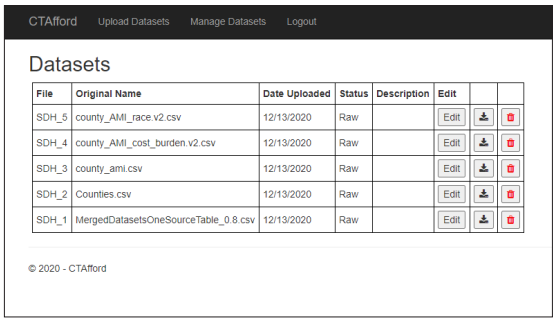
Drag and drop your files into the box. Allowable file types include: .csv, .xls, and .xlsx formats. You can drop either a single file or multiple files at a time. The maximum total allowable file upload size is 100 mb.

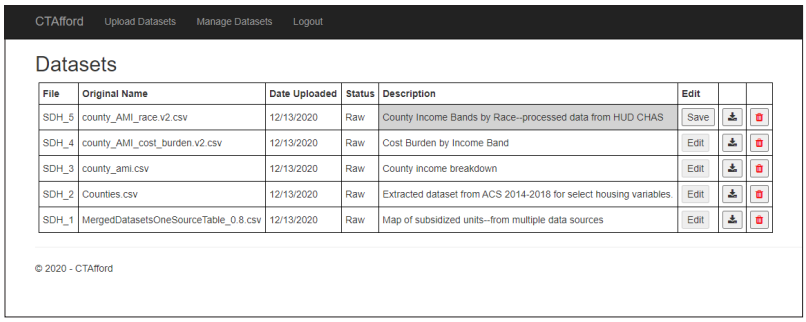
The maximum number of files that can be uploaded at one time is 5.



You can remove your files as necessary. When you're ready, submit your request through the upload button.

It will take a few seconds to upload your file depending on file size. When the files have been successfully uploaded to the server, a dialog box should appear. The server will also notify of any errors in the uploading process.





Click on the Manage Datasets to view your uploaded datasets. We will review your datasets and if it passes moderation, we will make it available for other users to access through our API. It is important to check your datasets after upload and update the "Description" field with a description of your dataset. If you do not, we cannot guarantee that we will review your dataset. If your dataset is approved and made available for access through the API, we will update the status in the "Status" tab. All datasets are labelled as "Raw" until they are moderated and approved for release. You may download or delete your datasets at any time. However, once your datasets are approved and released for use, they will remain on our servers and through the API even if you delete the files afterward. Please inquire if that happens and you no longer wish for your dataset to be shared.

It is also helpful to upload a data dictionary alongside your dataset so that we can cross-reference the fields and ensure that the data is well-formatted. If you do upload a data dictionary, be sure to label it as such in the description. In general, please adhere to tidy data principles when uploading your datasets.

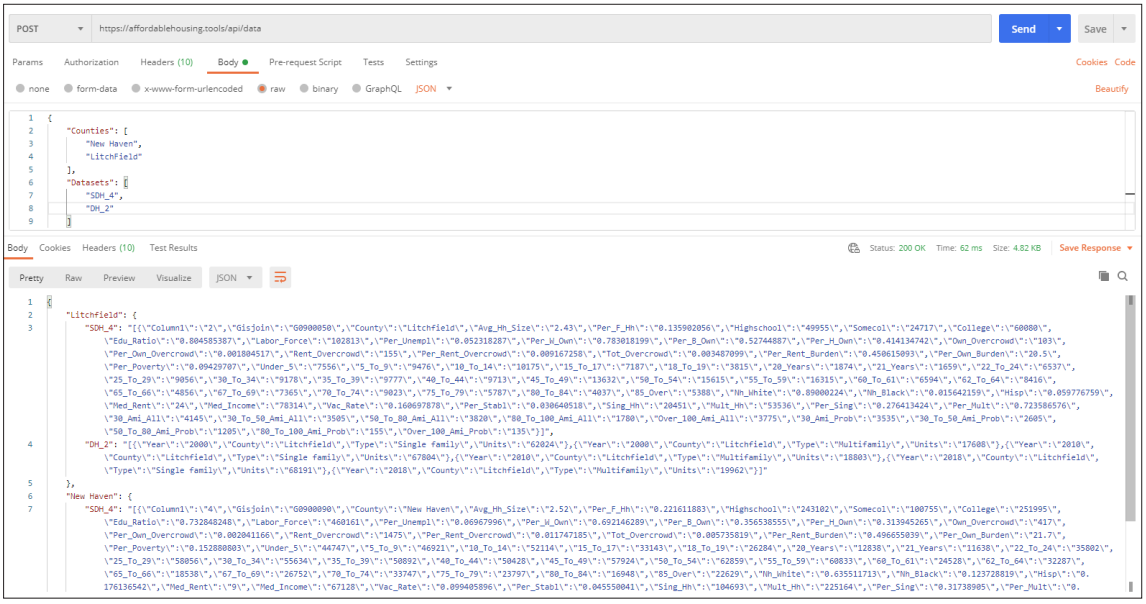
Type 2 Datasets

Conserved Indexing Cols.			Geography Cols.		Subsidy Cols.			General Data Cols.		
Project Name	ProjectID	RowID	Municipality	Address	Subsidy Class	Subsidy Subclass	Subsidy Units	Total Units	Owner	...
ABC Phase 1	ID1	1	...	...	Value 1	Value 5	...	...	...	...
ABC Phase 2	ID1	2	...	...	Value 2	Value 6	...	...	...	...
DEF	ID2	3	...	...	Value 3	Value 7	...	...	...	...
GHI	ID3	4	...	...	Value 4	Value 8	...	...	...	...

Type 1 Datasets

Identical to Type 2 Datasets			Subsidy Value 1 Block				Subsidy Value 2 Block			Additional blocks...
Indexing Cols.	Geography Cols.	General Data Cols.	Subclass 1	Subclass 1 Units	Subclass 2	Subclass 2 Units	Subclass 1	Subclass 1 Units	Subclass 2	Subclass 2 Units
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...
...	...	...	...	...	...	...	...	...	...	...

When uploading datasets to our platform, please remove unnecessary whitespace and incongruous rows. Ensure that all rows of a given column are formatted in the same way and with the same type of data (e.g. strings, numeric, datetime, etc). When uploading data on subsidized or affordable units, please follow one of the above formats. For all datasets please ensure that you have an indexing column and a geography column.



Our API can be used to access datasets that have been uploaded and approved on our database. This Postman illustration shows how the API works: the requesting user submits, using their language of choice, a POST request that includes a JSON body in the following format: “Counties” : {“geography 1”, “geography 2”,…}, “Datasets”: {“dataset 1”, “dataset 2”,…}. The API will retrieve all relevant datasets and their respective rows for the given geographies specified. Currently our API can only work with datasets with fields specifying “Counties”.

```python
# Example Python script to access affordablehousing.tools API endpoint.
import urllib.request, urllib.error
import json
import pandas as pd

# user inputs geography and datasets (updated 12/2020)
# geography choices: Fairfield, Hartford, Litchfield, Middlesex, New Haven, New London, Tolland, Windham
# dataset choices:
# SDH_1, SDH_2, SDH_3, SDH_4, SDH_5, SDH_6, SDH_7, DH_1, DH_2, DH_3, DH_4, DH_5, DH_6, DH_7
geo = ['New Haven', 'Litchfield']
datasets = ['SDH_4', 'DH_2']
url = 'https://affordablehousing.tools/api/data'

# encoding and preparing the HTTP request
req_dict = {'counties': geo, 'datasets': datasets}
json_encode = json.dumps(req_dict)  # converts dictionary to json string
json_encode = json_encode.encode('utf-8')  # converting to bytes
http_req = urllib.request.Request(url, data=json_encode)  # request using the POST method
http_req.add_header('Content-Type', 'application/json')  # adding headers

# returns HTTP object
http_object = urllib.request.urlopen(http_req)
data_read = http_object.read()  # reads HTTP object to bytes
dict_read = json.loads(data_read)  # convert bytes to raw dictionary

# extracts saved to converted pandas dictionary where keys = geography, values = tables -> content
geo_dict = {}
for geography, tables in dict_read.items():
    tables_dict = {}
    for table, content in tables.items():
        # creating a dataframe from table and appending it to tuple of dataframes
        table_df = pd.DataFrame.from_dict(json.loads(content))
        tables_dict.update({table: table_df})
    # appending to dictionary of dataframes, indexed by geography
    geo_dict.update({geography: tables_dict})

# user analysis continues here using geo_dict...
```

```r
# Example R script to access affordablehousing.tools API endpoint.
library(httr)

# user inputs geography and datasets (updated 12/2020)
# geography choices: Fairfield, Hartford, Litchfield, Middlesex, New Haven, New London, Tolland, Windham
# dataset choices:
# SDH_1, SDH_2, SDH_3, SDH_4, SDH_5, SDH_6, SDH_7, DH_1, DH_2, DH_3, DH_4, DH_5, DH_6, DH_7
geo <-  list('New Haven', 'Litchfield')
datasets <-  list('SDH_4', 'DH_2')
url  <-  'https://affordablehousing.tools/api/data'

# encoding and preparing the HTTP request
req_list  <-  list('counties' = geo, 'datasets' = datasets)
json_encode <-  toJSON(req_list, pretty = TRUE, auto_unbox = TRUE)

# returns HTTP object
http_object <- POST(url, body = json_encode, encode = 'raw', content_type('application/json'))
data_read <- content(http_object, "text")  # reads HTTP object to character (text)
data_read <- fromJSON(data_read)  # converts text to list

# extracts saved to nested list where each index name is associated with a geography and values are tables
df_list <- list()
i <- 1
for(tables in data_read) {
  j <- 1
  for (tbl in tables) {
    df_list[[names(data_read[i])]][[names(tables[j])]] <- data.frame(fromJSON(tbl))
    j <- j+1
  }
  i <- i+1
}

# user analysis continues here using df_list...
```

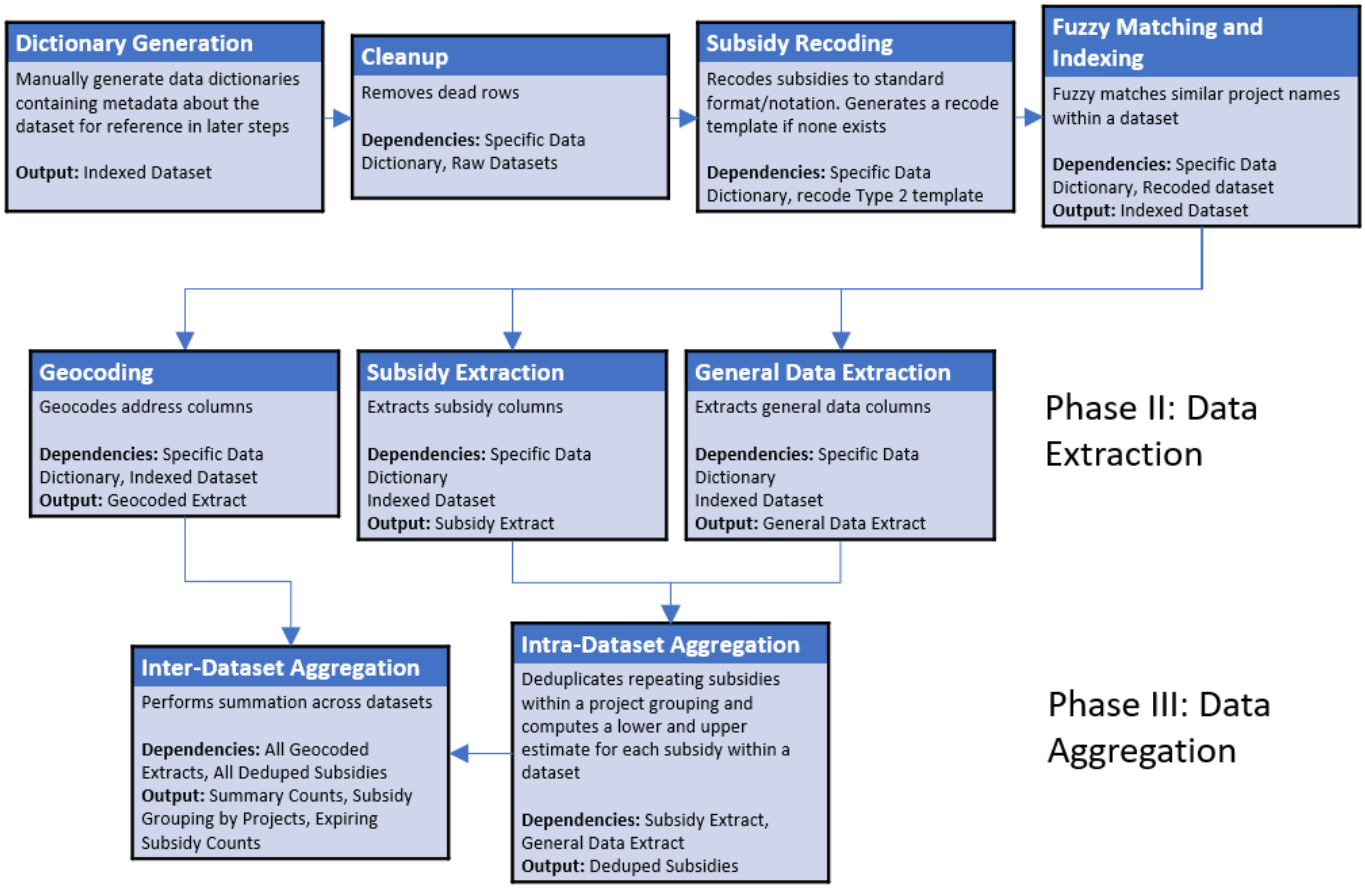
Example code in Python 3 (top) and R (bottom) for accessing datasets through the API.

Dataset Query Name	Dataset Description	Geography Allowed
SDH_1	This is a dataset of subsidized housing locations and details in Connecticut. It was compiled using several datasets from CTDOH, CHFA, HUD, and the National Housing Preservation Database.	Counties
SDH_2	Associated subset of SDH_1, processed for subsidy expirations.	Counties
SDH_3	This is a CTDOH dataset that contains locations and details of future subsidized housing developments in CT.	Counties
SDH_4	This is a curated dataset of select ACS variables related to housing. Several fields have been aggregated and computed relative to the original ACS extract.	Counties
SDH_5	This is an extracted dataset from the HUD CHAS data on affordable housing. It includes a breakdown of income bands relative to area median income for each county in CT.	Counties
SDH_6	This is an extracted dataset from the HUD CHAS data on affordable housing. It includes cost burden at each income band.	Counties
SDH_7	This is an extracted dataset from the HUD CHAS data on affordable housing. It includes a breakdown of cost burden by racial background.	Counties
DH_1	This is a processed dataset from ACS data that summarizes the number of households desiring housing versus the number of housing units available.	Counties
DH_2	This is a processed dataset from Census and ACS data that describes the total number of housing units broken down by county in CT. It has data from the 2000, 2010 Census and ACS 2014-2018.	Counties
DH_3	This is a processed dataset from Census and ACS data that describes homeownership (tenure) broken down by county in CT. It has data from the 2000, 2010 Census and ACS 2014-2018.	Counties
DH_4	This is a processed dataset from ACS data that describes the number of permits issued per year for several categories of buildings and is broken down by county in CT.	Counties
DH_5	This is a processed dataset from Census and ACS data that describes vacancy rates broken down by county in CT. It has data from the 2000, 2010 Census and ACS 2014-2018.	Counties
DH_6	This is a processed dataset from ACS data that describes cost burden by disability status broken down by county in CT. It uses data from ACS 2014-2018.	Counties
DH_7	This is a processed dataset from ACS data that describes number of disabled households in CT broken down by type of disability. It uses data from ACS 2014-2018.	Counties

These datasets are included in this version of the guide. You can access them through the API by specifying "Counties" as a key and list the datasets according to the Dataset Query Name.

Appendix A: Methods

We divided our extraction and analysis pipeline into three phases: the first phase scrubbed the data and indexed it for processing, the second phase extracted relevant information from each dataset in standard form, and the third phase aggregated, and computed sums of subsidy counts with respect to geography.



Phase I: Metadata Generation, Data Cleaning, and Indexing

Dictionary Design

In order to organize the dataset information, we created metadata to index each dataset. We manually developed dataset dictionaries to code for the relevant column data to extract. We chose classification parameters based on examining all the datasets and identifying similar and necessary columns. An example of a dataset dictionary (Governmentally Assisted) is shown below.

Column Variable	Classify	Metadata
RawDatasetName		
DatasetUID		
DatasetVersion		
DatasetType	Type.2	
OrgID		
Funder		
Administration		
Municipality	Address.City	
Project Name	Project.Name	Flag.ProjectID
Total	Unit.Total	
Family	Unit.Family	
Elderly	Unit.Elderly	
Handicapped	Unit.Handicap	
Rent		
Own		
Project Number		
Street Address #1	Address.StreetName	
Street Address #2	Address.StreetName	
Street Address #3	Address.StreetName	
Occ. Date		
Municipality.1		
Project Name.1		
Owner	Owner.Name	
Owner Address	Owner.Address	
City		
State		
Zip Code		
Management	Owner.Name	
Management Address	Owner.Address	
Management Address #2		
City.1		
State.1		
ZipCode		
Owner Type		
Contact		
Phone		
Agency		
Program	Subsidy.Name	

We designed our data dictionaries by noting conserved elements across datasets. We found that each dataset row must have the following minimum column information: project name, address, municipality, and subsidy. Type 2 datasets would have a single subsidy column while Type 1 datasets would have one or more subsidy columns.

Subsidy Standardization

In order for dataset rows to be comparable when combining datasets (i.e. an apples-to-apples comparison), it was crucial to recode subsidies to a standard format. Because each source dataset referred to subsidies according to their own standards, we developed a standard list of subsidies using our own language. This standard list was developed in consultation with both internal partners at the Urban Institute and with external collaborators at DOH and CHFA. The lack of standardization of subsidy names between datasets is a second consideration for a more robust future system.

We manually designed recoding templates, which we called a "categorizer," that would rename each dataset's subsidies to the corresponding standard list value. Subsidies in our standard list corresponded to a given class and subclass. Using our judgement and in consultation with our partners, we manually identified unique subsidy class/subclass values in each dataset and associated them to a standard list value. This process was laborious but crucial. Upon recoding, we expanded each Type 2 dataset to encode extra columns specifying the standard subsidy value for a given project or row. The associated dictionaries for Type 2 datasets were updated with new metadata. Type 1 datasets were unchanged. This was in part because our initial exploratory code used Type 1 datasets as a point of reference.

Row Indexing

Once the dataset subsidies were standardized, we created our own grouping indices, called "ProjectID" for a given row or group of rows. Indexing was a necessary step because it allowed us to further scrub specific row data which may not have a one-to-one correspondence/relationship with our indexing column. The indexing column therefore served as a join column across multiple data extracts. We used the concept of a project or development as the element of analysis and created our grouping index according to matching project names (with the corresponding dataset column specified in the data dictionary). Because multiple project names could refer to the same physical location (such as when a given property has phased projects), we used an inexact or fuzzy string match to group highly similar project names together. We used the union of two string-matching algorithms, Jaro-Winkler and Smith-Waterman, in order to capture the majority of grouped projects. In our initial row indexing, we used a stringent threshold of 0.9 (out of 1) to reduce false positives. We performed an additional step to reduce false positives by eliminating the top two words found across all project names. Finally, we only grouped similar project names within a city (i.e. we

stratified our rows based on municipality/town) in order to further reduce false positives as unrelated projects with the same name could be found in different municipalities/towns. An example of an indexed grouping (from the NHPD) is found below (subsidy columns excluded for simplicity).

NHPD Property ID	Property Name	Property Address	City	Total Unit	RowID	Clean_Proj	Group Flag	ProjectID
1013604	SHELDON COMMON I CO-OP	110 Martin St	Hartford	7	101	sheldon common i coop	Hartford101	e1f74d0c-e861-4af8-b63f-209c93f9429f
1013606	SHELDON COMMON II CO-OP	120 Martin St	Hartford	2	109	sheldon common i coop	Hartford101	e1f74d0c-e861-4af8-b63f-209c93f9429f

Manual reindexing

A log file for all grouped rows was generated for data validation and additional examination. For rows that were incorrectly grouped and need to be reindexed, we used a hardcoded template to regroup or drop specific rows. This step enabled us to fine tune any unnecessarily grouped rows. We found that indexing and reindexing was necessary because not all datasets were internally indexed, and those that were (e.g. the NHPD) did not incorporate our concept of grouping related project names. The need to index within datasets is a third consideration for a more robust future system.

Phase II: Geocoding, Subsidy Data Extraction, General Data Extraction

We performed the next three steps of our data processing simultaneously after indexing. Because not all data encoded in a given column or subset of columns has a one-to-one relationship with those from another subset of columns, extracting this data in parallel with a common join column (i.e. the ProjectID index), allowed us to accurately and cleanly represent each type of extraction. Three types of extractions were performed for each dataset: addresses were extracted for geocoding, subsidy columns were extracted for counting, and general columns (including total unit counts) were extracted for comparison and as references for possible future analysis.

Geocoding

We found that the ways in which address data was recorded varied highly between datasets. Even within datasets we found inconsistencies in the ways in which address data was entered, ranging from misspellings to extensive strings encoding apartment units to the inclusion of special characters, such as parentheses and incorrectly placed zip codes, in the text. Although the National Housing Preservation Database contained geocoded coordinates, we re-geocoded all addresses in order to provide a measure of consistency in our handling.

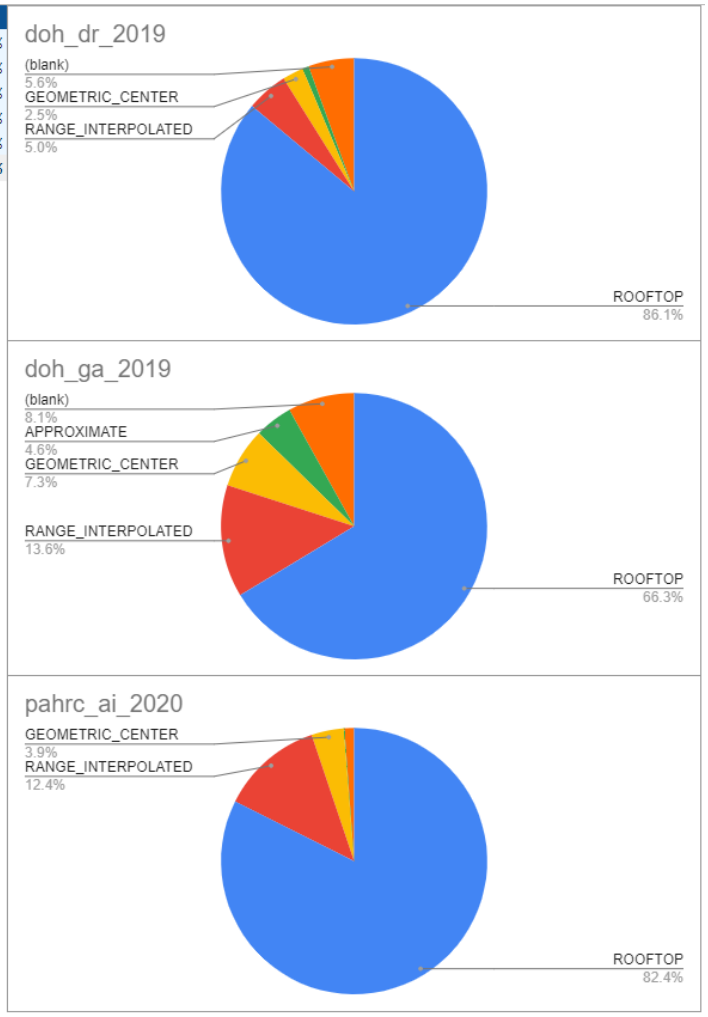
To robustly address these inconsistencies, we first filtered out empty whitespace and removed trailing zip codes which was difficult to geocode. We then utilized a context-free grammar (Python *lark-parser* library) and a series of regular expression rules to parse out addresses. We considered the typical address syntax: street number, street name, city, state. Numbers found in an address string were associated with the nearest subsequent word, assumed to be a street name. Subsequent numbers were associated with the next nearest subsequent word. The parser additionally filtered out optional "decorators" such as units or apartments (e.g. Unit 1, Apt 3). We hardcoded in the decorators we expect to see most often (such as "unit" and "bldg"), which catches the majority of extraneous text encountered.

For geocoding, we utilized Google's Geocoding API, which we found to be user friendly and highly robust, to code the parsed addresses. Each row in the geocoding output corresponded to a single address found within a project's row. For rows that encoded mul-

tiple addresses within the address column cell, we expanded the result such that multiple rows with the same reference ProjectID index were created. Those rows that return errors were logged and flagged. Overall, the variation in which an address is listed, which directly impacted our ability to geocode, is a fourth consideration for a more robust future system.

We logged the type of geocoding result returned for every address string parsed as a read-out of the quality of the address string. We considered the best strings as returning rooftop coordinates, and the worst as returning blanks. A comparison of three of the datasets (NHPD, Governmentally Assisted List, and Deed Restricted List) is seen below:

LocationType	doh_dr_2019		doh_ga_2019		pahrc_ai_2020	
ROOFTOP	2195	86.08%	1674	66.35%	987	82.39%
RANGE_INTERPOLATED	128	5.02%	344	13.63%	149	12.44%
GEOMETRIC_CENTER	65	2.55%	185	7.33%	47	3.92%
APPROXIMATE	20	0.78%	116	4.60%	1	0.08%
(blank)	142	5.57%	204	8.09%	14	1.17%
Total	**2550**	**100.00%**	**2523**	**100.00%**	**1198**	**100.00%**



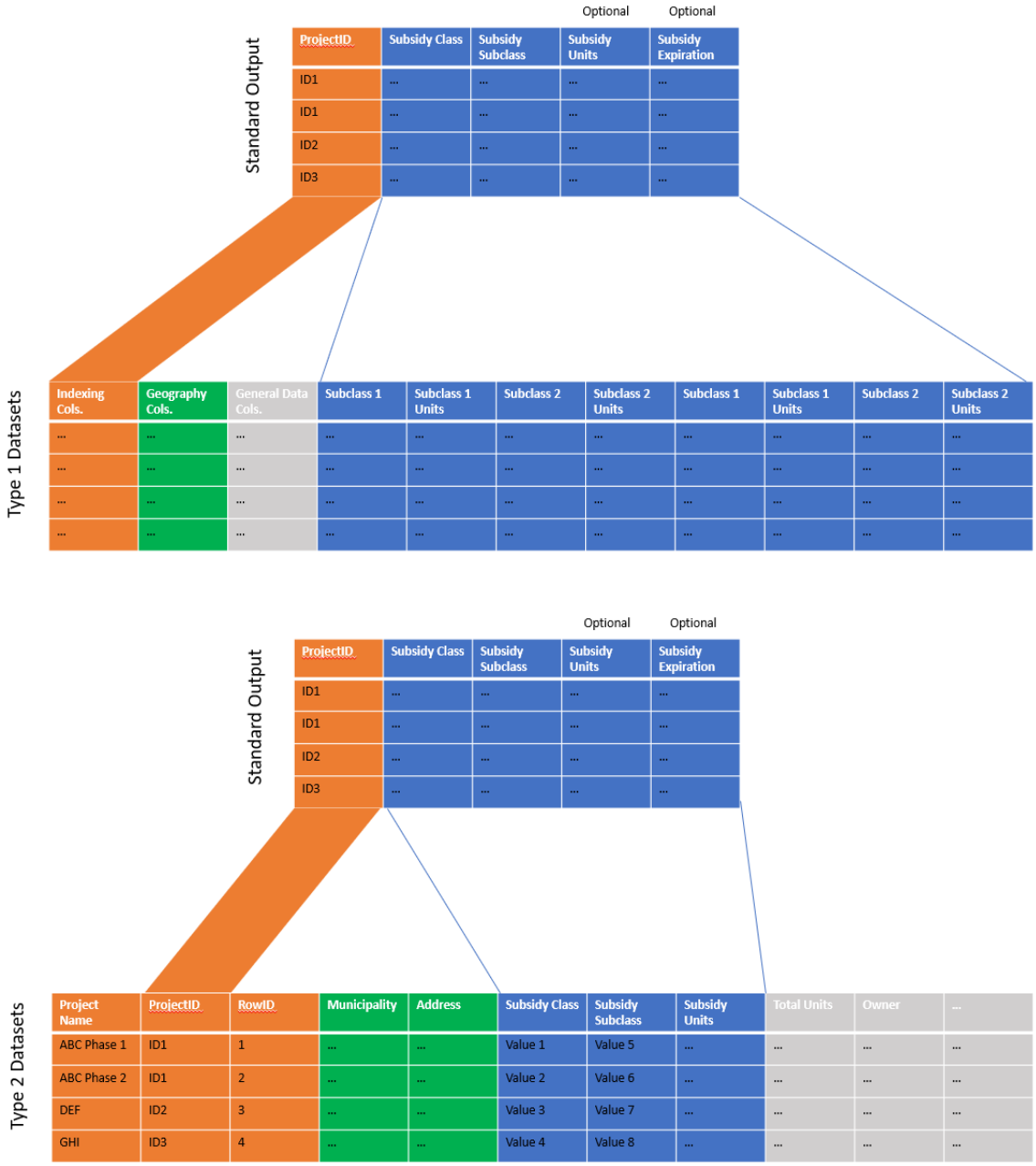
Subsidy Extraction

We extracted subsidy values in the form of subsidy classes and subclasses: these values were already standardized earlier in the process. This step reformatted subsidy column information such that Type 1 and Type 2 datasets subsidy columns remap to a single subsidy class and subclass column for a given subsidy in a given row. For rows that encoded multiple subsidies within the subsidy column(s), we expanded the result such that multiple rows with the same reference ProjectID index were created.

For Type 1 datasets (NHPD and DOH Deed Restricted) the subsidy columns are subdivided into blocks with each row checked for the existence of a given subsidy block. For a given row, if a subsidy block exists, its column value(s) is/are captured. For Type 2 datasets (DOH

Governmentally Assisted, CHFA 8-37bb, and HUD datasets), the designated subsidy class and subclass columns are identified for a given row and the corresponding cell values are captured. Two other optional columns in the output, subsidy unit counts and subsidy expiration dates are encoded if such data is included. The lack of direct or unambiguous subsidy counts in some datasets is a fifth consideration for a more robust future system.

As sourced, we had to manually pre-process both the "2020 Master PBV Log" and the "HUD Affordable Housing List" because the provider (HUD) had encoded multiple bits of information within single columns which should have been split into separate columns. This included combining the total and subsidized unit counts of a given row within a single column as well as cases of inconsistent data entry. The need to pre-process datasets is a sixth consideration for a more robust future system. An illustration of the final reformatted output for Type 1 and Type 2 datasets is shown below.



General Data Extraction

Extraction of other types of data, including the total number of units for each project (if applicable) was completed. This allowed us to separate out data that was potentially useful for future analysis. In this step, we coded for a brief list of exceptions for grouped project names if we believed that the total number of units within that group was not equal to the sum of those units. The inconsistency in conserved column variables between datasets and the need to hardcode total unit count within grouped project names are seventh and eighth considerations for a more robust future system. This concluded the data extraction process.

Phase III: Deduplication and Aggregation

Subsidy Count Aggregation

By formatting and extracting subsidy and general data, we were able to reconstruct data in such a way that our aggregation and analysis did not depend on hardcoded metadata (i.e. the data dictionaries) that pointed to specific locations within a dataset for the final analysis. This enabled us write code that was generalizable in aggregating the total subsidy count.

Intra-dataset aggregation:

We first needed to validate subsidized unit counts within datasets to ensure that we did not double count units and that those counts were reasonable (i.e. that they did not exceed the total number of units, both subsidized and unsubsidized, within a given development or ProjectID grouping). Double counting was primarily a concern for the NHPD dataset which allowed for two instance of a given subsidy subclass, but we developed a generalized subsidy grouping technique that was applicable for all possible future occurrences.

We considered several scenarios in aggregating subsidies within a given dataset since the fidelity of certain datasets was higher than others. While the NHPD data contained both total and subsidized unit counts, other datasets, like the DOH Governmentally Assisted List did not. Yet other datasets, such as the HUD Affordable Housing List contained inconsistent records where some, but not all, records contained the number of subsidized units. As mentioned previously, the inconsistency in the availability of this data makes it crucial to design a better standard. A table of the types of data available within each dataset is shown below.

Dataset	Project Name	Preexisting Indexing		Address	Total Units	Subsidized Units	Owner Information
National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Governmentally Assisted List (2019)	Yes	Some, inconsistent	Yes	Yes	Yes	No	Yes
Deed Restricted List (2019)	Yes	No	Yes	Yes	No	Yes	No
Multifamily 8-37bb Housing Portfolio (2020)	Yes	No	Yes	Yes	No	Yes	Yes
2020 Master PBV Log	Yes	No	Yes	Yes	Yes	No	Yes
HUD Affordable Housing List	Yes	Some, inconsistent	Yes	Yes	Yes	Some	Yes

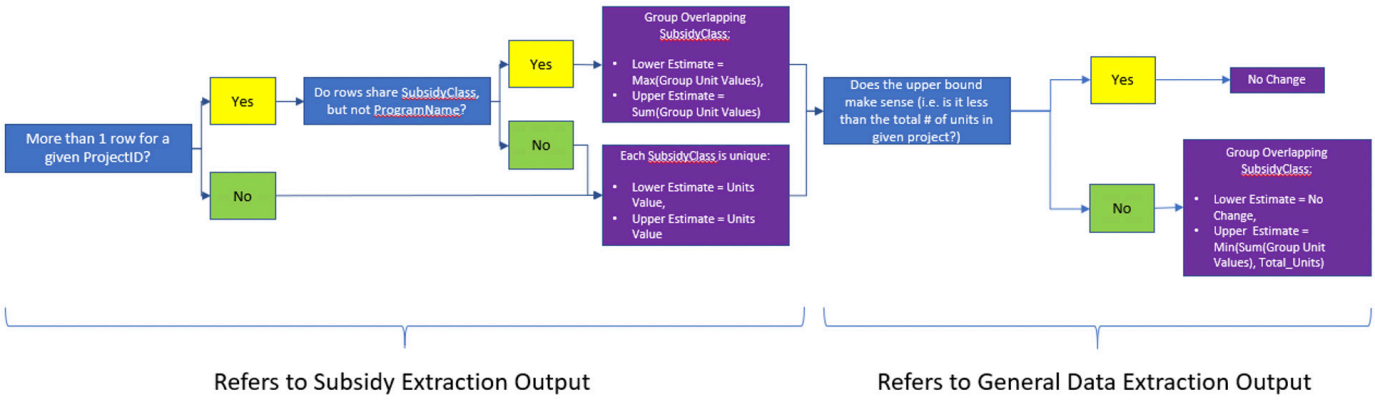
Because some datasets had information only on the total units for a given project or development, we needed to account for/describe the uncertainty of how many of those total units were actually subsidized. To do so, we created a range of estimates, with a lower and upper bound. The lower bound would consider the scenario where there was a minimum number of units on a subsidy, generally 1, and the upper bound would consider the scenario where all the total units were on a subsidy. Lacking additional information, we were unable to create a tighter range without factoring in arbitrary assumptions about the underlying nature of a subsidy. However, for datasets that had much higher fidelity and specified the exact number of subsidized units, we would take those values as the lower and upper bounds.

We first had to account for inconsistencies in missing data for datasets such as the HUD Affordable Housing List. We filled in missing subsidy unit information with a nominal flag value (generally "1") to denote existence of that subsidy.

Next, we created our upper and lower bound estimates for a given subsidy with the above consideration of whether the subsidy unit counts were given in the dataset. We then examined if there was any repeated subsidy class within a ProjectID. To aggregate repeating or duplicated subsidy class values within a given ProjectID, we considered the two scenarios where (1) there was maximum overlap in the number of housing units between those two (or theoretically more) repeated subsidies, and (2) there was minimum overlap between the repeated subsidies. In the first scenario, we coded the aggregated or deduplicated subsidy count to be the maximum value of the set. In the second scenario, we coded the aggregated subsidy count to be the sum value of the set. The exception to this was for repeated subsidies that must be disjoint: we made an exception for deeds, which we considered to be always mutually exclusive of one another and must be summed.

Finally, we compared our ranged estimates with the total unit count from the general data extract if such a count existed for the given dataset. We revised our estimates such that for a given subsidy within a ProjectID, the lower and upper bound estimates must be equal to or less than the total unit count. We computed the total unit count as the sum of the total unit count of all project names within a given ProjectID, with the exception of the hardcoded instances described in the general data extract section above.

A simplified decision tree of the inter-dataset aggregation heuristic is shown below.



Inter-dataset aggregation:

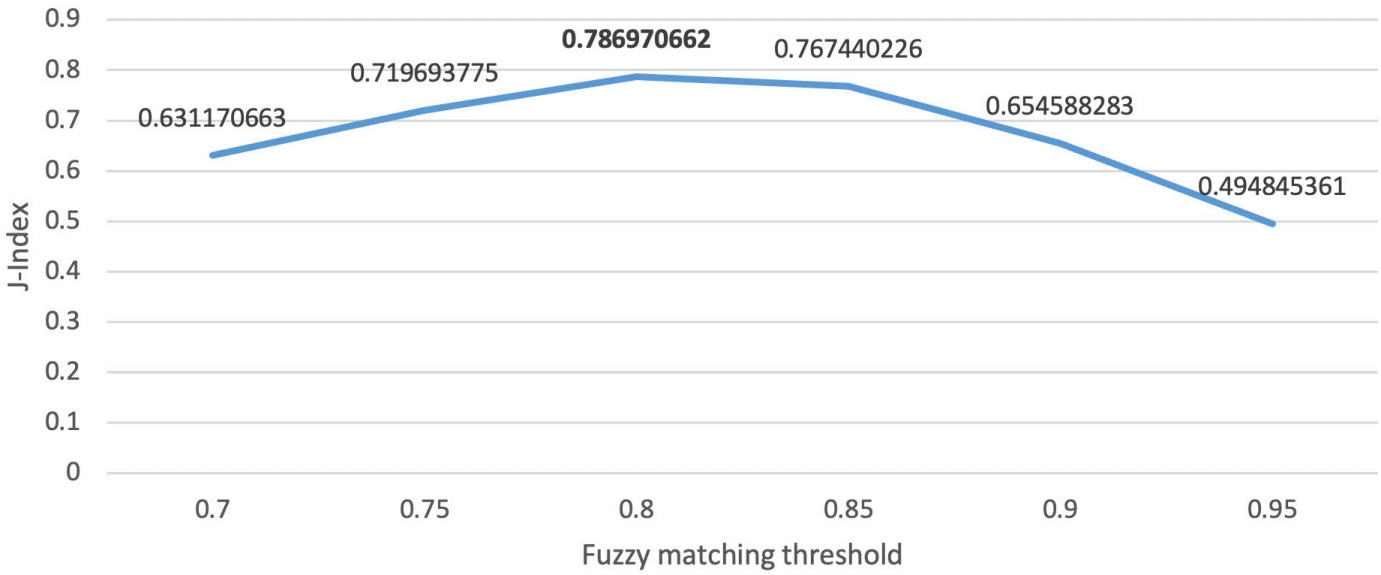
Next, we considered the sum of all subsidies across datasets. Because datasets provided overlapping information on the same subsidies, summing the data would overcount the true number of subsidized units. Instead, we developed a priority tree which specified two key parameters: (1) whether or not to sum (e.g. perform a "group by" function) a given subsidy by its class or subclass, and (2) which dataset to use to aggregate a particular subsidy. This allowed us to have granular control over which subsidies classes to be grouped together and which dataset to use for the summation. A table describing the prioritization and summation is shown below.

Subsidy Class	Dataset to Prioritize/Use	Summation Method (Group By)
FHA HUD Multifamily Mortgages	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
HOME	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
LIHTC	Multifamily 8-37bb Housing Portfolio (2020)	Subclass
Public Housing	HUD Affordable Housing List	Class
Rural Housing Loans (Section 515)	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
Rural Housing Loans (Section 538)	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
Section 202	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
Section 236	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
Section 8 Non-Voucher Programs	National Housing Preservation Database: Active and Inconclusive Properties CT (2020)	Class
Section 8 Voucher Programs	HUD Affordable Housing List, 2020 Master PBV Log	Subclass
State Subsidies	Governmentally Assisted List (2019), Multifamily 8-37bb Housing Portfolio (2020)	Subclass
Deed Restrictions	Deed Restricted List (2019)	Subclass
Single Residency Occupancy	HUD Affordable Housing List	Class
Other Federal Subsidies	Multifamily 8-37bb Housing Portfolio (2020)	Subclass

We performed summation of counts at two geographic levels: counties and towns (denoted as Connecticut county subdivisions by the Census). This summation was extracted for (1) the total merged data, and (2) only for data within the NHPD dataset, which we used as a reference. Because of inconsistency in naming conventions for towns between datasets (i.e. some datasets used informal town names), we standardized the values for address columns using reference 2010 Census county and town shapefiles from the University of Connecticut's MAGIC library. We performed spatial joins of each geocoded point to associate unique addresses to the correctly formatted county and town. Our final summation function took geography as an input argument such that we had the flexibility to sum across either a county or town. We also identified specific rows in the HUD datasets that was not geographically linked to any point but was attached to a given town, which ultimately traced back to Housing Choice Voucher counts and had to be specially considered and added to the total summation. The inconsistency in naming convention for cities/towns between datasets and the existence of specially coded rows in certain datasets are ninth and tenth considerations for a more robust future system.

We also performed a likewise geographic summation looking at temporal changes in subsidy counts. For this, we relied on the subset of data that encoded expiration dates. We iteratively filtered for, summed, and extracted rows from this subset where a given subsidy had not yet expired. Our unit of measurement was one year, so we created data output slices with one-year increments from 2020 (present day) until 2060 (the last known instance of an expiring subsidy).

To understand how subsidies related to one another across datasets, we recreated Project ID associations to complete our final summations of all project names across datasets. This allowed us to identify the specific bundle or permutation of subsidies associated with a physical project or property. To do so, we repurposed earlier code written for fuzzy matching. Given the observation that there was higher variability and less consistency in the naming conventions across datasets, we intuitively understood to lower the fuzzy matching threshold from 0.9. We empirically determined this lower threshold by grouping using a range of thresholds and performing a manual binary classification of validity. We then performed a sensitivity-specificity analysis by identifying the maximum Youden's Index (J) as the optimal value for thresholding. For this analysis we used the DOH Governmentally Assisted Dataset as a reference because it appeared the least well-behaved dataset. The range of J-indices is shown below for this training set.



Appendix B: Recommendations

We recommend a complete standardization in the ways in which data is collected and stored. As noted in the methodology section and summarized below, there are multiple pieces of evidence to suggest that a robust and automated statewide housing database is impossible without restructuring the ways in which data providers collect, organize, and submit information. These limitations include:

- The existence of variation between dataset column structure.
- The lack of standardization of subsidy names between datasets.
- The need to index within datasets.
- The variation in which an address is listed, which directly impacted our ability to geo-code.
- The lack of direct or unambiguous subsidy counts in some datasets.
- The need to pre-process datasets.
- The inconsistency in conserved column variables between datasets.
- The need to hardcode total unit counts within datasets.
- The inconsistency in naming convention for cities/towns between datasets.
- The existence of specially coded rows.

The difficulties of automating a standardized inventory of subsidized units lie primarily in the fact that dataset providers organize their data in highly varied formats. There appears to be little pre-processing on some of the providers' ends and some datasets appear to be better formatted than others. Overall, there appears no single way that the providers validated their data before submitting the datasets to us.

There is currently no way for data providers to understand how data from each other looks like and how they should structure their data to be comparable to those of other providers. Interagency data validation does not appear to be present, although there were highly conserved dataset structure elements. For instance, all datasets included a column for "project names," indicating that the elemental unit of analysis was a housing project or development. Additionally, there were columns for addresses, municipalities, subsidies, and units which further indicated the importance of the geographic location of a project and its associated subsidies and units. Finally, there was often peripheral information encoded within each dataset, including information about the owners and/or the managers of a given project as well as subsidy expiration dates. These well-conserved columnar data could be further improved and standardized to provide a comprehensive and comparable comparison.

To address the above difficulties, we recommend the creation of a new type of dataset template with clearly defined subsidy classification standards that accounts for both federal and state subsidies. Without the creation of this standardized dataset for all applicable housing data providers, it is prohibitively complicated to provide ongoing subsidy tabulation accurately and consistently. We suggest the following design considerations for a new database:

Limitation	Solution
The existence of variation between dataset column structure.	Single dataset column structure. If possible, we recommend a Type 1 structure like the National Housing Preservation Database. The US Census Datasets (e.g. ACS) are similarly structured.
The lack of standardization of subsidy names between datasets.	Standardized and publicly available codebook using the National Housing Preservation Database as a reference but including state subsidies and HUD programs. All housing data providers should have copies and references to this
The need to index within datasets.	Single ruleset for indexing projects/developments. We recommend combining phased developments within a single physical property address.
The variation in which an address is listed, which directly impacted our ability to geocode.	Standard formats for addresses with separate columns for decorators such as apartment unit values.
The lack of direct or unambiguous subsidy counts in some datasets.	Correct for all missing data.
The need to pre-process datasets.	Adherence to Tidy Data conventions.
The inconsistency in conserved column variables between datasets.	Dataset must encode a minimum of: address, subsidy unit total, total units in property, and subsidy expiration date. Entries should not be null if possible.
The need to hardcode total unit counts within datasets.	Specify the total number of units within one physical property address.
The inconsistency in naming convention for cities/towns between datasets.	Standardize naming of cities/towns to the exact names given by the US Census.
The existence of specially coded rows.	Adherence to Tidy Data conventions. Eliminate all non-stratified rows such that every row must be comparable to another row.

FCCHO
Fairfield County's Center for Housing Opportunity

We welcome collaborators!

To become more involved in this effort, visit fccho.org or contact:

Fairfield County's Center for Housing Opportunity
815 Main Street, Bridgeport, CT 06604

FCCHO PARTNERS

FAIRFIELD COUNTY'S COMMUNITY FOUNDATION

Fairfield County's Community Foundation
TOGETHER WE THRIVE

As a nonprofit partner and thought leader since 1992, Fairfield County's Community Foundation brings together passionate people and trusted resources to solve our region's challenges through innovative, collaborative solutions.

PARTNERSHIP FOR STRONG COMMUNITIES

PARTNERSHIP FOR STRONG COMMUNITIES

Partnership for Strong Communities (PSC) is a statewide nonprofit policy and advocacy organization dedicated to ending homelessness, expanding affordable housing, and building strong communities in Connecticut.

REGIONAL PLAN ASSOCIATION

RPA

Regional Plan Association (RPA) is one of America's oldest urban research and advocacy organizations. RPA works to improve the prosperity, infrastructure, sustainability and quality of life of the New York-New Jersey-Connecticut metropolitan region.

SUPPORTIVE HOUSING WORKS

supportive housing WORKS

Supportive Housing Works' (SHW) mission is to end homelessness in Fairfield County by advancing a collective impact approach through dedicated staff, committed partners, and effective leadership.