8-2018

GIS for Public Health Assessment A CASPER Methodology Framework

Cesar J. Garrido Lecca

University of Redlands

GIS for Public Health Assessment
A CASPER Methodology Framework

A Major Individual Project submitted in partial satisfaction of the requirements
for the degree of Master of Science in Geographic Information Systems

by

Cesar Garrido Lecca Rivera

Ren Fang, Ph.D., Committee Chair

Mark Kumler, Ph.D.

July 2018

GIS for Public Health Assessment

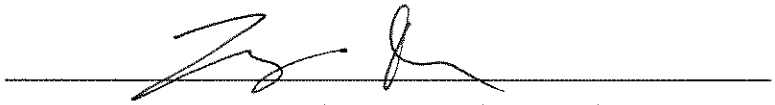
A CASPER methodology framework

by

Cesar Jhonatan Garrido Lecca Rivera

The report of Cesar Jhonatan Garrido Lecca Rivera is approved.

Mark Kumler, Ph.D.



Fang Ren, Ph.D., Committee Chair

July 2018

Acknowledgments

First, of all, I would like to thank God Almighty for His grace in giving me the opportunity to stay here in Redlands with peace of mind, contentment and good health. He has given me the patience and strength to face all my challenges and the wisdom to understand my errors and avoid repeating them.

I would like to thank professor Fang Ren for her contribution to the realization of this project and her encouragement when facing the different challenges that being a foreigner means in this great country. My sincere gratitude to the MS GIS program faculty, their support, and commitment to the quality of our education was a source of encouragement during this year.

I would also like to thank Jack Dangermond, the President of ESRI, and the University of Redlands MS GIS Program for funding my studies through the Jack Dangermond fellowship program. I am in debt to you all, and I hope I can contribute to the society as a means to repay your support of me.

Abstract

GIS for Public Health Assessment

by

Cesar Garrido Lecca Rivera

One of the greatest challenges in managing emergencies is to determine the impact of the disaster and to respond effectively to the primary needs of those affected. The Community Assessment for Public Health Emergency Response (CASPER) is a specific methodology designed to quickly and effectively estimate the health status and basic needs that must be addressed by the health agency. The present study discusses the application of a complete GIS-based framework for the improvement of each of the main phases of the CASPER methodology: preparation, conduction, analyzing data, and writing a report. The results show the GIS approach to implementing CASPER can significantly reduce the time required for data collection and processing, improve the quality of the collected data, and allows the agencies to make a real-time decision based on the situational awareness of the communities.

Table of Contents

Table of Figures

List of Tables

List of Acronyms and Definitions

CASPER	Community Assessment for Public Health Emergency Response.
CDC	Centers for Disease Control and Prevention.
DEM	Digital Elevation Model.
GIS	Geographic Information Science.
GPS	Global Position System (receiver)
WHO	World Health Organization.

Chapter 1 – Introduction

One of the greatest challenges in managing emergencies is to determine the impact of the disaster and to respond effectively to the primary needs of those affected. Throughout the years, natural disasters have indicated that the main needs in these situations are food, medicines, and drinking water (Watt, 1977).

To effectively assist those affected, government agencies must assess the impact of the disaster after it has occurred (Das, Lashkari, Biswas, 2013). The CASPER methodology, used by many US government agencies, allows them to assess and determine the health status and medical needs of the affected community (Centers for Disease Control and Prevention, 2012). At present, many government agencies are applying this methodology; however, this process largely depends on manual work and each phase of the process, including preparation, conduction, analyzing data and writing a report, is done independently. The current workflow is time-consuming and introduces potential human errors (Platz et al., 2017). In addition, due to the manual operation of CASPER, decision makers do not have situational awareness of the field work, which limits the capacity for timely decision-making in the event of a disaster.

Since most of the data collected and analyzed in CASPER methodology are spatial data, a GIS framework would be an appropriate solution for effectively monitoring and executing impact assessment. As such, this project seeks to address these challenges through the implementation of a comprehensive GIS framework that allows agencies to apply the CASPER methodology computationally. Since Esri ArcGIS software is already licensed or accessible to most of the major entities conducting CASPER surveys at a global level, this project chose the ArcGIS suite as the platform.

1.1 Client

The primary client for this project was Esri's Human and Health Services Department. Esri is a worldwide leading GIS company, providing GIS solutions for all types of industries, from emergency management to solutions for private businesses where the location factor is used for decision making. Este Geraghty, the Chief Medical Officer at Esri, is the lead Esri contact for this project. As part of the work team, Jared Shoultz, the Health GIS Pattern Specialist, was designated as the contact in charge of providing access to information and technology needed to develop the project. In addition, the Center for Disease Control and Prevention (CDC) provided support and expertise for implementing the CASPER methodology.

1.2 Problem Statement

Although the GIS toolkit created by CDC to apply CASPER methodology is very useful, the phases implemented for this methodology are all independent, and part of the process is completed manually. This approach leads to a substantial increase in the time required and the complexity of implementing the methodology. Considering that time is a determining factor in the context of an emergency, it is necessary to simplify, as much as possible, those steps that involve an unnecessary expense of time and resources.

1.3 Proposed Solution

The proposed solution is to implement a complete GIS-based framework that creates a standardized set of tasks and processes to integrate and automate (as much as technically possible) the main phases of CASPER Methodology. The main components include automating the cluster methodology, managing the field operations and standardizing the communication of the results with an engaging and interactive medium. The goals, scope,

and methodology used to implement this project will be discussed in the following sections.

1.3.1 Goals and Objectives

The main goal of this project was to reduce the time and complexity in the deployment of the CASPER methodology by health agencies. The project had five objectives:

- Integrating the main phases of the CASPER methodology in a single workflow, thus allowing the information to flow effectively between one phase and another;
- Automating the cluster sampling process;
- Implementing digital surveys enabling real-time monitoring of field activities;
- Analyzing and publishing the results of the survey in an effective way that allows real-time access;
- Making the configurable solution available at either the CDC Portal or the Esri Solutions Portal, so that users from various health agencies can get easy access.

1.3.2 Scope

The scope of this project involved the development of three modules that would integrate the main processes of CASPER. The Preparation and Analysis Module would include the functionalities to perform the cluster sampling design, allowing users to define 30 clusters that the methodology requires. The Fieldwork Module would involve the designing digital surveys with smart rules for mobile devices to ensure the data quality. The Reporting Module would allow stakeholders and decision makers to maintain operational awareness of the execution status of CASPER methodology. Finally, this project would include a user manual as part of the solution templates of Esri.

1.3.3 Methods

The development of the GIS CASPER framework involved different methods. The project management was conducted using Scrum as the agile framework to optimize the management of the multiple teams involved in the project. For the development of the Planning and Analysis Module, Python was selected as the programming language for the creation of the CASPER statistical tools. In addition, the use of ArcGIS Pro task was a key component to create a user-friendly workflow. For the development of the Fieldwork Module, the configuration of a Survey 123 template was chosen. This template was configured to ensure smart rules through SQL code and native survey rules. Finally, for the development of the Reporting Module, Dashboard for ArcGIS items was applied. This dashboard was configured following the CDC standard for showing the rates required for the CASPER methodology.

1.4 Audience

The intended audience for this report includes the CDC employees related to the implementation and training of the CASPER Methodology. Also, health practitioners in local and state health departments, health manager officials interested in both, health prevention and post disasters plans, and any agency that currently is planning to implement CASPER methodology. Also, technical readers related to GIS and post-emergency response are included as part of the audience of this report.

1.5 Overview of the Rest of this Report

This report discusses the implementation of a complete GIS-based framework for the automation of CASPER methodology. Chapter Two describes the background and literature review, including an examination of previous research undertaken related to the

importance of epidemiological information after a disaster, statistical techniques in the collection of data, a digital platform for the automation of data collection, GIS as an effective mean of integration and communication among others. Chapter Three outlines the business problem, the requirements analysis and the system design of the solution. This chapter gives a complete overview of each component of the solution specifying the how they work together to address the business problem. Chapter Four describes the data administration for the project, from the conceptual database model design through the logical model design, data recollection methods, to the data scrubbing and loading into the physical database. Chapter Five refers to the development process taken to create the solution. This chapter describes the development of Python tools, smart forms and operation dashboards that are part of the solution. Chapter Six discusses the analysis and results of the project showing the benefits obtained and the lessons learned. Chapter Seven discusses the conclusions as well as the future works that could improve the scope and results of this project.

Chapter 2 – Background and Literature Review

After an emergency, government health agencies must carefully decide on the strategy they will use to respond effectively. Part of this strategy is to determine which places will be prioritized for aid. By using statistical methods, the Community Assessment for Public Health Emergency Response (CASPER) enables public health practitioners and emergency management officials to determine the health status and basic needs of the affected communities rapidly. Geographic information systems have shown to be the effective platform to integrate, analyze and communicate CASPER information. This chapter discusses the concepts related to the CASPER methodology and how different digital platforms and statistical techniques address the different challenges that a CASPER implementation involves. Finally, this chapter concludes by pointing out GIS as a key platform for the improvement of CASPER management as well as the integration of its processes.

2.1 Community Health Assessment

Epidemiological information collection is well recognized as essential for emergency management and response (de Ville de Goyet & Lechat, 1976). However, health agencies face many challenges in carrying out this work; challenges include the lack of time to collect data in the field, limited access to disaster-affected areas, societal insecurity and instability, cultural and language barriers, and finally the political challenges of data confidentiality (Morton & Levy, 2011). To address these challenges, multiple agencies have developed and applied various techniques in a wide range of emergencies (Bradt & Drummond, 2002). For example, the methodology created to assess the preschool child immunization index in the USA (Serfling, R. E., & Sherman, 1965) or the methodology

implemented to provide community-based information essential for the Smallpox Eradication Program in West Africa (Henderson, Davis, Eddins, & Foege, 1973). Among these data collection and estimation methods, CASPER has been widely adopted and will be detailed in the following section.

2.1.1 CASPER

Throughout the 1970s, many field professionals adapted conventional epidemiological techniques by simplifying them through statistical sampling methods (Henderson & Sundaresan, 1982). In 1990, based on these statistical techniques, the World Health Organization (WHO) published nine protocols describing best practices for rapid health assessment (World Health Organization, 1990). This helped health professionals optimize the strategic planning of health assessment. Based on a modified version of the WHO-published sampling methods, the Centers for Disease Control and Prevention (CDC) developed CASPER as a standardization initiative for health assessment in 2009 (Malilay, Flanders, & Brogan, 1996).

CASPER provides a baseline for the full implementation of this rapid health assessment methodology. CASPER is based on a two-stage cluster sampling design in which 30 clusters that can be census blocks or block groups, according to the area of study size, are selected pseudo-randomly. Then, seven interviews are conducted systematically in each of the 30 clusters. The goal is to create a statistically significant sample that provides the basis for valid estimation of the entire community (Centers for Disease Control and Prevention, 2012).

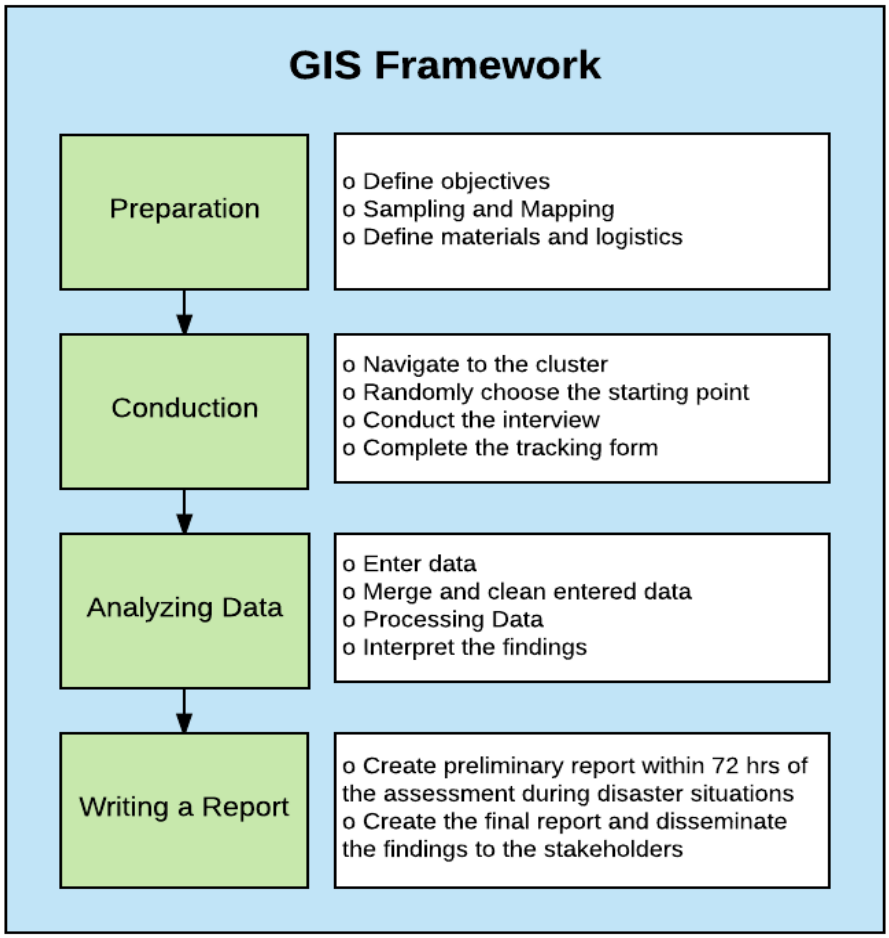


Figure 2-1 CASPER Workflow

As Figure 2-1 shows, the CASPER methodology is made up of 4 phases. In the preparation phase, the study area is defined, and the 30 clusters are selected. Normally the cluster selection is made by a pseudo-random method that takes into consideration an external variable, such as total housing units or total occupied housing units, to prioritize the selected blocks. The conduction phase is about executing a systematic survey schema to sample 7 housing units in each block. The methodology indicates that the housing units must be pseudo-randomly selected to start and then the total number of housing

units in the block must be divided by seven, to determine the sampling interval. Given the importance of preserving statistically sound sampling, the selection of 30 census clusters and a minimum of 7 housing units per block should be maintained as fixed parameters. In the next phase, data collected need to be analyzed to estimate the health situation for the entire community. It is important to create a weighting variable for each survey carried out, so that all the surveys have the same degree of influence on the final estimation. Finally, the last step of CASPER is to prepare a complete report with the estimates calculated for each question asked.

2.2 Digital Platforms and GIS

Different mechanisms have been used to address community health assessment challenges. One of them has been the use of digital platforms for data collection (Morton & Levy, 2011). Traditional methods of data collection involve processes such as the printing and distribution of survey files, as well as subsequent data entry. These processes are time-consuming and increase the complexity of health assessment. The development of communications technologies, global navigation satellite systems, and client-server technology has enabled field users to collect data more efficiently through electronic devices (Nusser, Miller, Goodchild, & Clarke, 2001). In particular, GIS and mobile devices with GPS receiver capabilities have led to not only a big reduction in the use of paper, but also have improved crew navigation capabilities. This represents a considerable saving in the time required for the execution of the surveys (Kaiser, Spiegel, Henderson, & Gerber, 2003). For example, after Hurricane Charley occurring in August 2004, the North Carolina Division of Public Health successfully performed sampling using GIS and handheld computers (Young, Sanchez, & Malilay, 2005). Also, after

Hurricanes Charley, Frances and Ivan, handheld devices were used in conducting rapid need assessment in Florida; this technology helped survey teams to improve their efficacy when maneuvering to selected clusters and moving through disaster-affected areas where navigation was extremely complicated because of the damage of street signs and landmarks (Young et al., 2005).

Situational awareness is an essential element for decision-making during the collection of epidemiological information (MacEachren et al., 2011). One of the challenges that interviewers face is the lack of public safety, so monitoring crews and the survey status is important. It is critical to ensure a bi-directional communication between field staff and staff in the planning units. Field feedback should provide information on duplication, logistical issues, etc. (World Health Organization, 2004). However, many of the agencies are not technologically prepared to reach this point. For example, during the Household Health Assessment conducted by the Florida Department of Health's Disaster Community Health Action Team (DCHAT), which was carried out after Hurricanes Charley, Frances, and Ivan, the DCHAT team used electronic devices to collect information, but progress status was only reported daily (Young et al., 2005). This is a limitation for effective management of reassignments in the surveys, in addition to preventing decision makers from having real-time situational awareness.

Before analyzing the information collected in the survey, data entry is required, either by a manual process if the survey was developed manually or by gathering the digital files from each device in the case of electronic surveys. After Hurricane Ike in Texas (2008), the Texas Department of State Health Services team performed the integration and cleanup of their data using Microsoft Excel and then created a GIS

database using Epi Info. To complete data entry and processing, four people were appointed (Zane et al., 2010). This step adds time and the need for resources to the health assessment process.

It is also imperative for health agencies to present and disseminate the results of the analysis efficiently. The challenges presented in this phase are the concise integration of the results, effective presentation of the conclusions, and the efficient management of the security policies about access to information. GIS allows public health officials to quickly and efficiently understand the many factors associated with a health assessment, in addition to providing the ability to update data constantly. GIS can also process and present the many date and location attributes provided by spatiotemporal data, allowing decision-makers to understand the multiple implications of the data (Young et al., 2005). Additionally, the use of local government or health agencies web pages that allow controlled access to health information has been approved to be effective in protecting data confidentiality (Morton & Levy, 2011).

2.3 Summary

Understanding community health issues and needs is a critical component in every post-emergency response plan. However, the execution of the data collection faces many challenges in a post-disaster scenario. In the recent years, many techniques and technologies have been used to address these challenges. The evolution of GIS set this technology as a powerful ally when it comes to data integration process. As a system of registry, GIS allows users to integrate a different kind of information from multiple sources including mobile devices and sensors, standardizing the process, and reducing its complexity. As a system of insights, GIS can integrate statistical and mathematical

models that are needed for epidemiological analysis and enrich them with location-based information. Finally, as a system of engagement, GIS empowers health agencies to communicate the results of their analysis effectively by using the power of Web GIS.

Chapter 3 – Systems Analysis and Design

To develop and implement a successful GIS solution it is critical to have a clear understanding of the business processes of the client, the value generated for each process, and how GIS can improve or add value to them (Tomlinson, 2007). This requires GIS planners to collect the business requirements and carefully design a GIS system aligned to the user needs.

This chapter discusses the complete process of designing a GIS solution. Section 3.1 describes the business problem to be addressed by the solution. Section 3.2 describes the requirements including functional and non-functional requirements. Section 3.3 presents the system design and how it meets the requirements of the client to add value to the business process. Section 3.4 explains the overall project plan created to implement this solution successfully. Finally, Section 3.5 summarizes the chapter showing the conclusions of the project planning.

3.1 Problem Statement

The main problem addressed in this project was the low efficiency in implementing CDC's CASPER methodology. For example, the majority of the processes defined in this methodology were completed manually, which was not only time-consuming but also error-prone. In addition, all the phases involved in this methodology were implemented independently, making a real-time health needs assessment difficult as the evaluation can only be conducted after all the surveys were collected.

3.2 Requirements Analysis

Effective requirements management is fundamental to project management. According to Randell et al. (2014), at least 70% of software development projects fail due to poor management of user requirements, which in turn has caused an estimated rework cost of at least 45 billion dollars per year. Kusiak and Tang (2006) proposed three states of requirements analysis, including identification, diffusion, and attainment. For this project, a series of user interviews and requirements refinement were conducted to identify the functional and nonfunctional requirements. Functional requirements are those related to the function of the system to complete the business process. Non-functional requirements are those required to support the business process technically. Table 3.1 summarizes the collected requirements.

Category	Requirement
Functional	Query County Level Block Data
	Geo-enrich AOI Layer
	Block Random Sampling
	Publish Layers
	Prepare and Analyze Data
	Conduct Surveys
	Report CASPER Status
Non-Functional	ArcGIS Solutions Deployment Tool Compatibility
	Web GIS Pattern Required
	Base Components Restriction

Table 3-1 List of Requirements

In the following sections, these requirements will be briefly described.

3.2.1 Functional Requirements

The main functional requirements of the project include querying county level block data, generating and enriching the AOI block data, conducting the clustering sampling schema, publishing the results, collecting field surveys, reporting the survey status, and preparing and analyzing the results of the survey. Below is a summary of each requirement.

Query County Level Block Data: For the querying county level block data requirement, the system shall return the block level data for the county defined by the user. This functionality is part of the Planning and Analysis module.

Geo-enrich AOI Layer: The system shall allow the user to define an area of interest (AOI) as a specific study area and apply the geo-enrichment ArcGIS Online tools to the layer. This functionality is part of the Planning and Analysis module. It requires an internet connection and an ArcGIS online account with credits to apply the geo-enrichment tool.

Block Random Sampling: Following the CDC standards, the system shall generate a new layer with no more than 30 selected blocks and display it on the map. This functionality is part of the Planning and Analysis module. It requires an enriched layer mentioned earlier.

Publish Layers: The system shall publish the selected blocks and AOI layers to the ArcGIS Online portal. This functionality is part of the Planning and Analysis module.

Prepare and Analyze Data: The system shall generate an Excel file with the proper format to be imported in EPI Info. The system shall also generate the estimation of a

selected variable from the survey. This functionality is part of the Planning and Analysis module.

Conduct Surveys: The system shall allow the user to use smart forms that show the bank of questions for the specific CASPER survey. The system shall also validate the consistency of the answers and choose the questions according to the answers entered. This functionality is part of the Field Work module.

Report CASPER Status: Based on the CDC Standards (Centers for Disease Control and Prevention, 2012), the system shall show the operational status, demographic information and medical emergencies in the field. This functionality is part of the Reporting module.

3.2.2 Non-Functional Requirements

The non-functional requirements of the project are as follows.

ArcGIS Solutions Deployment Tool Compatibility: All modules shall be packaged so that the complete solution can be deployed using the ArcGIS Solutions Deployment Tool.

Web GIS Pattern Required: All the modules should be connected to the ArcGIS Online/ Enterprise Organizational account of the user. It is required to use ArcGIS Pro as the base solution for the Planning module.

Base Components Restriction: The solution should only include those components that are part of a basic ArcGIS Online / Pro implementation.

3.3 System Design

Based on the processed business requirements, a comprehensive GIS system was proposed, as shown in Figure 3-1 System Design.

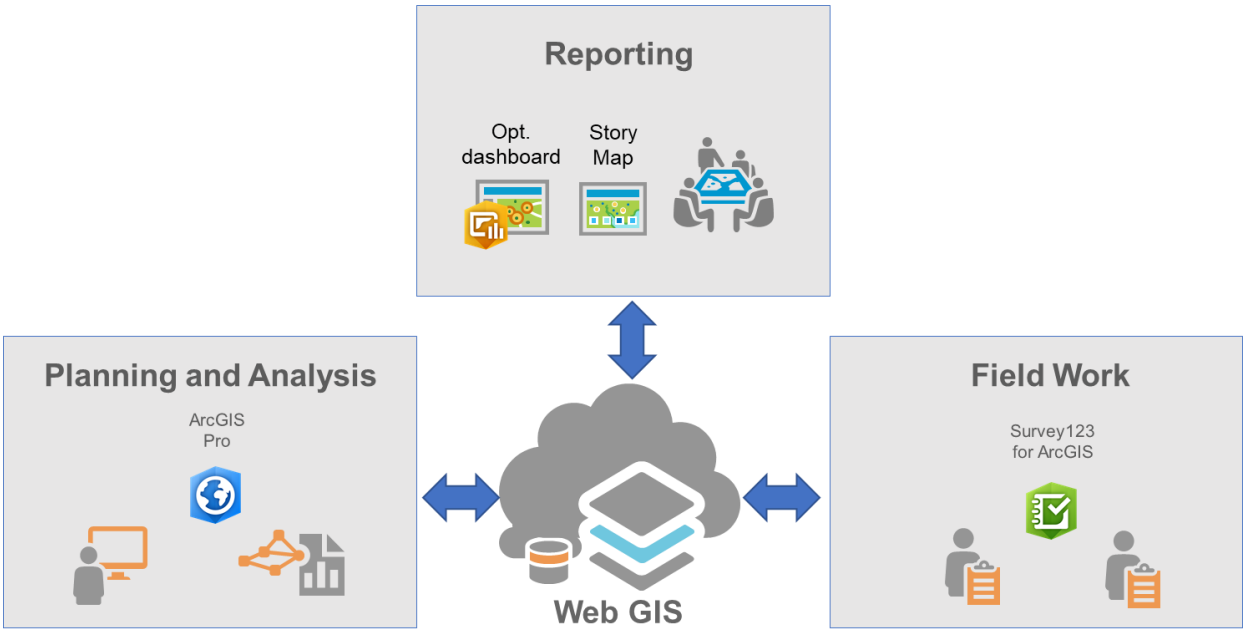


Figure 3-1 System Design

There were three modules and one geodatabase designed in this system. The geodatabase was designed to house local and state boundaries as well as census data of the United States. The geodatabase would be used in the first module, Planning, and Analysis, which includes the functionality to perform the cluster sampling process, allowing users to define 30 clusters that the methodology requires. This module also shares the results of this process with the fieldwork crew administrators and with the stakeholders. This module was created through the configuration of automatic tasks in ArcGIS Pro, as well as the integration of Python code for the automation of the Cluster Sampling methodology.

The second module, Field Work, involves field surveys. Once the households have been selected, the crew administrator can distribute the digital surveys to the fieldworkers. They can conduct the surveys digitally and transmit the progress of the surveys in real time. There is direct communication between the field staff, and the crew administrator in case a reassignment of surveys is necessary, or in case an emergency occurs. This module was created through the configuration of a survey template based on the CDC questions bank by using Survey 123 for ArcGIS.

The third module, Reporting, allows stakeholders and decision makers to maintain operational awareness of the execution status of the CASPER survey. It includes real-time reporting features, such as a summary of the key data. The Reporting module was created through the configuration of Operation Dashboards that summarize the data analysis results. Moreover, a Story Map that presents the entire process followed to achieve these results is part of this module.

A geodatabase was designed to house local and state boundaries as well as census data of the United States.

3.4 Project Plan

Planning is a key task for project management. There are two main methodologies when it comes to managing software development: Waterfall, also known as the traditional linear approach, and Agile, an iterative team-oriented approach that has increased its popularity in recent decades. This project was conducted from the perspective of the Agile development methodology, focusing on providing the client with valuable software in an ongoing and early manner (Hazzan & Dubinsky, 2014). All modules were

implemented iteratively and incrementally with the constant collaboration of the project participants.

3.4.1 Agile Methodology Implementation

Scrum has been chosen as the Agile framework (Schwaber, 1997) to be used. The key roles were defined following the good practices of this methodology (Schwaber, 2013). Figure 3-Scrum roles show the key roles in this framework.

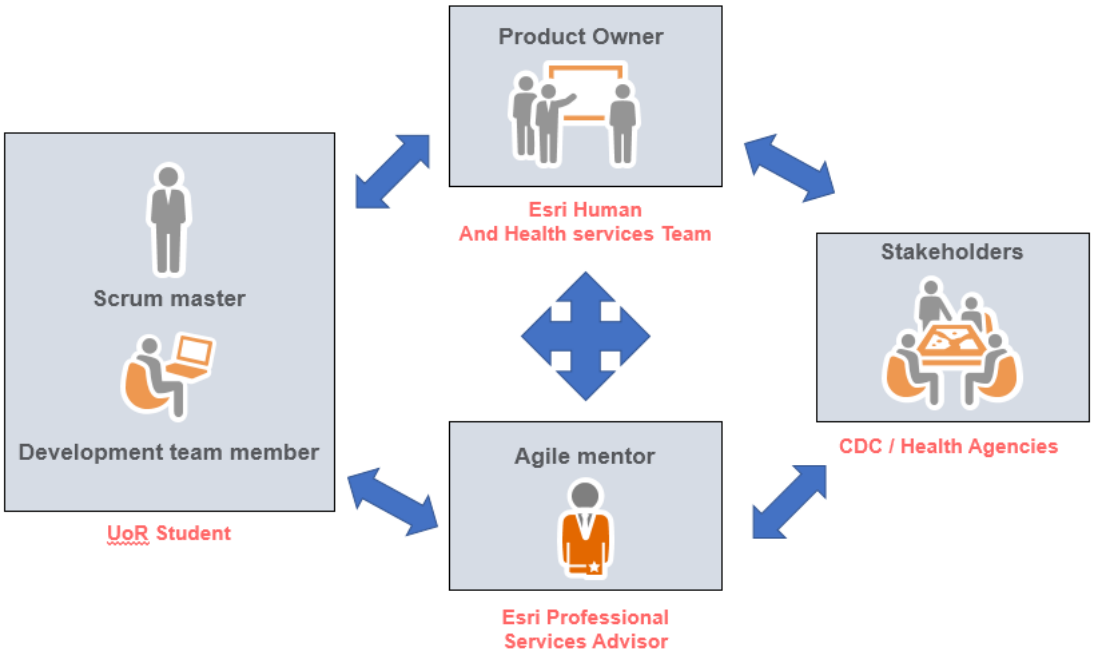


Figure 3-2 Scrum roles

The Product Owner role is responsible for closing the gap between the stakeholders and the person responsible for developing the solution. The product owner is trained to understand the customer's business problems and knows very well how to approach the solution to these problems. For this project, Esri's Health and Human Services team covered this role working closely with the CDC. Understanding that the best way to

transmit information is through face to face meetings (Hazzan & Dubinsky, 2014), there were weekly video calls with the Esri Health GIS Patterns Specialist, Jared Shoultz, who provided support by answering questions about the health sector during development. Also, these meetings included Tanya Bigos, an Esri Solutions Engineer for the Health sector as well as other stakeholders on an as-needed basis.

The Development Team Member role oversees all the tasks of design, programming, configuration, and deployment of the components of the product. The Scrum Master is the person in charge of providing the necessary resources for the development of the product and maintaining the workflow constant. For this project, Jhonatan Garrido Lecca assumed both roles, developing and integrating the three modules of this solution.

The stakeholders are all those who are affected by or who influence the decisions of the project. For the project to be successful, it is necessary to effectively solicit regular business feedback from stakeholders. For this project, the primary stakeholder was the CDC, as it is the leading Public Health institution and the undisputed authority when it comes to the CASPER methodology. Monthly meetings with the CDC, to validate the progress of the solution were planned.

Finally, the Agile mentor is that leader experienced in Agile development projects that provide valuable feedback to the team. His expertise in the application of Agile methodologies provided solid foundations throughout the development process. For this project, the Esri Professional Services team provided an Agile mentor. All these roles interacted in each of the stages of the development of the solution.

3.4.2 Project Plan Phases

The scope of this project was defined by a period of 7 months. The closing date planned was June 20, 2018. However, due to minor changes and in the inclusion of one nonfunctional requirement (Base Components Restrictions), the end date of the schedule changed to July 6, 2018. This project was divided into five main sprints or iterations, which contain a series of weekly sub-sprints to obtain constant feedback. At the beginning of the project, the workload was more focused on finalizing the project proposal, which was approved by Esri and the CDC. After that, this first sprint included the collection of the general technical requirements of the project. The final deliverable of this sprint was the project proposal document as well as the official list of requirements.

The second sprint was focused on the development, testing, and deployment of the Planning and Analysis module. This module contains all the functionalities developed in ArcGIS Pro to automate the process of the two-stage cluster sampling design. The final deliverable of this sprint was a fully functional Planning and Analysis module.

The third sprint included the development, testing, and deployment of the Fieldwork module. This module involved the creation of the digital surveys on Survey 123. This sprint also included the unit tests and the deployment of the module in a test environment. The final deliverable of this sprint was a fully functional Field Work module.

The fourth sprint included the development, testing and deployment of the dashboards and their indicators, and the Story Map that summarizes the processes carried out. The final deliverable of this sprint was a fully functional Reporting module.

Finally, the fifth sprint was focused on integral testing of the solution and the deployment of the final user offices. In addition, the user documentation was created. The final deliverable of this sprint was the official user documentation. All the

deliverables were planned to be tested by the CDC and Esri. However, due to communication problems between Esri and the CDC, each deliverable was tested by the Pasadena Public Health Department and the Vermont Health Department.

3.4.3 Change Management

Each of the deliverables of this project was carried out iteratively. During every weekly meeting, the product owner (Esri) and the scrum master (Cesar Garrido Lecca) evaluated the changes required by the end users to estimate its impact. When a change caused an impact on the time of the project that delayed the final delivery by more than two weeks, that change was included as part of the second phase of the solution. Those requests for change that were accepted were officially communicated by email, and the list of functionalities that was shared with the end users was updated. At the end of each weekly meeting, the approved changes were prioritized for their implementations.

3.4.4 Communication Plan

To manage team communication efficiently, a planned communication schedule was developed:

Table 3-2 Communication Plan

Content	Purpose	*Responsible*	Timing or Periodicity	Methods of Communication
Kick-off or Start-up Meeting	Team presentation.	*Esri Team* Scrum Master	At the beginning of the project	In-person Meeting Formal Presentation
Publication of the solution proposal	Define Project Scope	*Esri* The CDC *Scrum Master*	According to the calendar of deliverables.	Word Document Shared in Google Drive - Formal Report
Progress review	Validate development progress	*Product Owner* Scrum Master	Weekly	e-meeting (Skype)

Methodology review	Validate project management	Scrum Master *Agile Mentor*	Monthly	e-meeting (Skype) Informal Report
Progress presentation	Validate development progress	*Esri*, Health Agencies, *Scrum Master*	Monthly	e-meeting (Skype)
Update of the technical requirements.	Allow all parties to know the status of the requirements	Scrum Master	When changes are approved in the list of requirements.	Excel document shared in Google Drive - Formal Report
Delivery of the module	Formal delivery of the module	Esri, Health Agencies, Scrum Master	According to the calendar of deliverables.	e-meeting (Skype)
Delivery of the final solution	Formal delivery of the solution	Esri, Health Agencies, Scrum Master	According to the calendar of deliverables.	In person meeting
Delivery of user documentation.	Formal delivery of documentation.	Esri, Health Agencies, Scrum Master	According to the calendar of deliverables.	Word Document Shared in Google Drive - Formal Report

3.5 Summary

This chapter laid out the planning and design of the solutions that address the user needs to implement the CASPER methodology. The Agile methodology was selected due to its effectiveness in managing communications among project participants. This was of great benefit since it is possible to counteract certain communication problems with the CDC development team at an early stage of the project, making it possible to carry out the tests of each module with alternative users. Also, the Scrum Master can effectively coordinate the support required for each module with the various Esri teams (Mobile Apps Team, ArcGIS Pro Team, Content Team, Python Team). Finally, the effective management of the changes allowed us to mitigate the increase of the scope of the project and generate potential future enhancements for the next release of the solution.

Chapter 4 – Database Design

Data is a key component in any GIS. To ensure the quality of a GIS system output, it is necessary to define the correct data model that depict all the business goals, and that ensure the quality of the data stored in the database. This chapter discusses the database design process for this project. It begins with Section 4.1 describing the conceptual data model which abstracts real-world elements and its relationships into a simplified diagram to reflect the business process. Section 4.2 describes the transition from the conceptual data model to a logical model where tables and relations of the database are detailed. Section 4.3 details the data sources and the type of metadata standard used for this project. Section 4.4 discusses the data collection methods, and Section 4.5 describes the process of data cleaning and loading into the database. This chapter concludes with a summary in Section 4.6.

4.1 Conceptual Data Model

A conceptual data model is a type of data abstraction that provides the key concepts needed to support the system environment at a very high level (Navathe, 1992). After a series of regular meeting with the CDC staff, the following entities were identified as a part of the CASPER process (Figure 4-1).

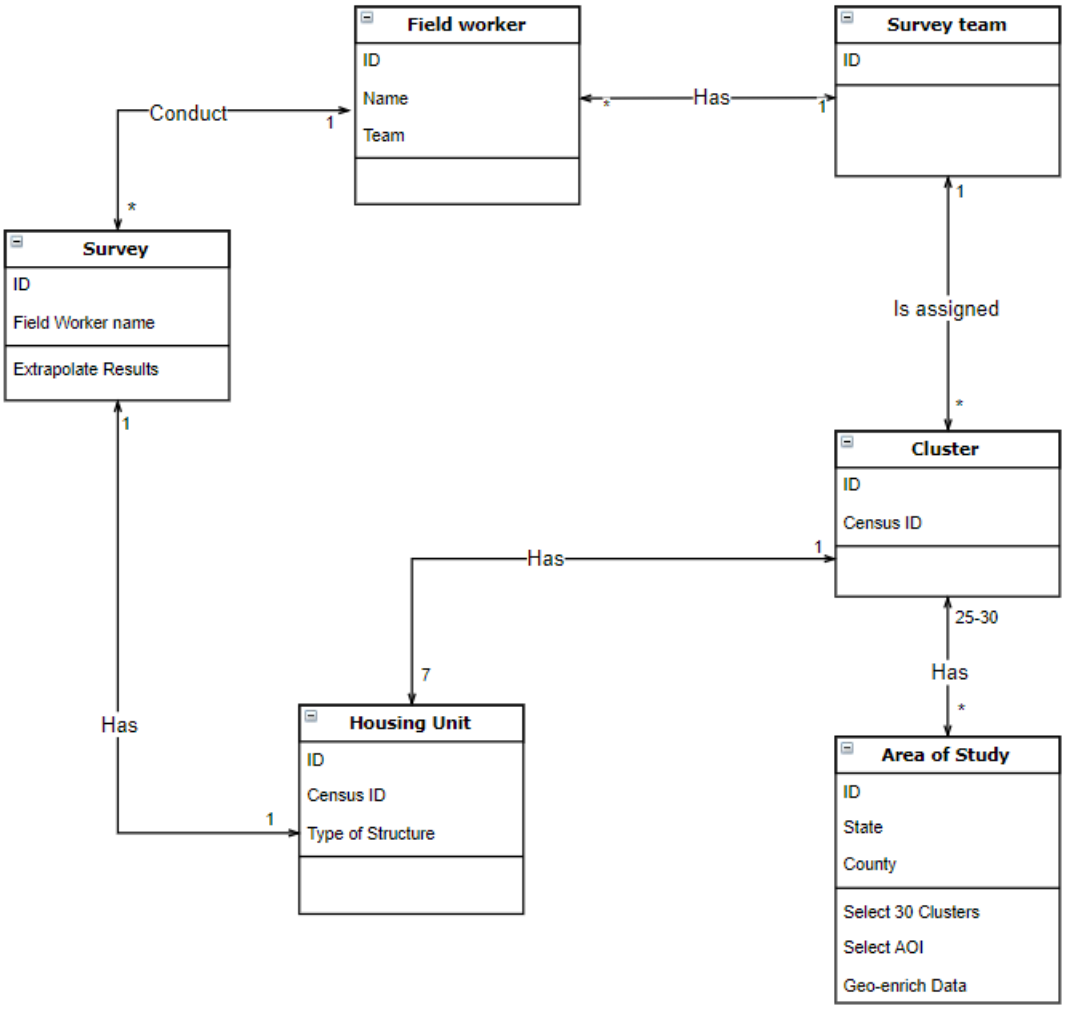


Figure 4-1 Conceptual Model

The study area is composed of census blocks (clusters). Within each study area, it is possible to select a specific Area of Interest (AOI) that is enriched with demographic variables and 30 clusters are pseudo-randomly selected. Each cluster has a maximum of seven selected housing units. The survey teams are assigned to several clusters to conduct the surveys. Each survey team is composed of several field workers who are responsible for surveying the required number of housing units.

4.2 Logical Data Model

The logical data model is a low-level representation of the data. It is also called implementation data model because it has the elements needed for a complete implementation (Navathe, 1992). For this project, a total of ten tables have been defined within a geodatabase implementation. Figures 4-2 and 4-3 show the logical model designed, grouped by module.

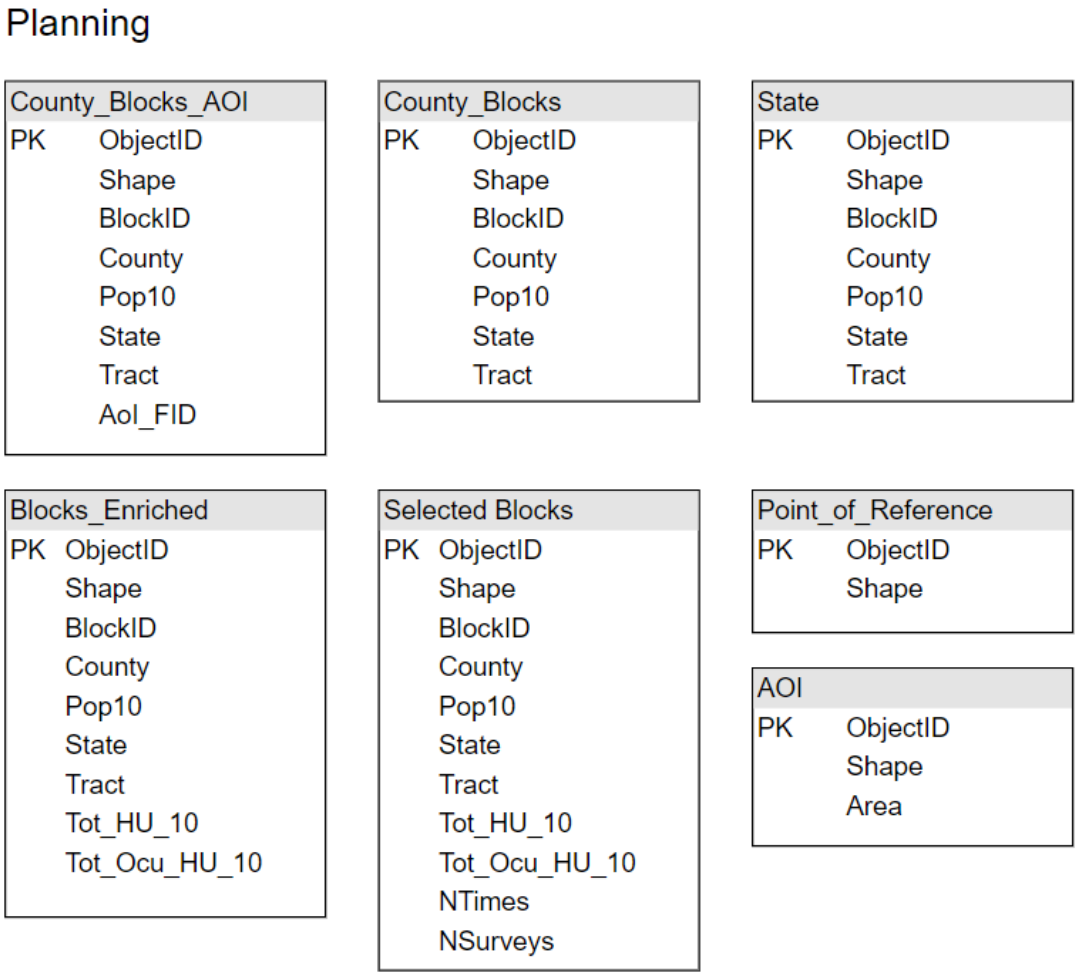


Figure 4-2 Logical Model - Planning

For the planning module, seven tables were generated. The State table contains all the census blocks in the United States grouped by states. The Point of Reference table

stores the location of the reference point used for the creation of the County_Blocks table which contains the census blocks at a county level. The AOI table stores the geometric information of the AOI polygon that will be used to reduce the study area. The County Blocks AOI table stores those blocks filtered spatially by the AOI layer. The County Enriched table stores the census blocks enriched with the variable of total housing unís. Finally, the Selected Blocks table contains the records of the blocks selected by the pseudo-random selection.

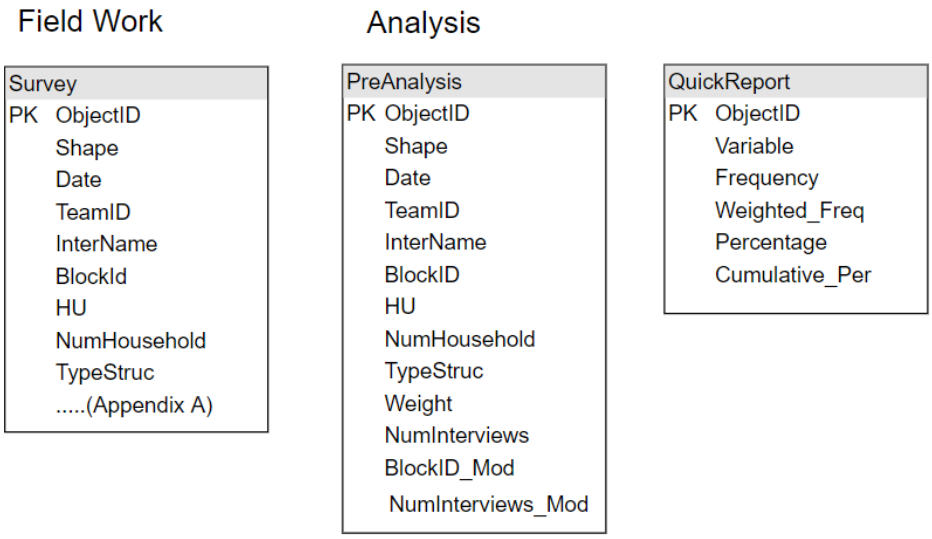


Figure 4-3 Logical Model – Field Work And Analysis

For the field work and the analysis module, three tables were created. The Survey table store all the information related to the surveys and survey teams. The PreAnalysis table contains the information of all the survey results to calculate the weighting field. The QuickReport table contains the estimation results of one selected variable in the system.

In addition to the tables, a complete series of data domains were created through Survey 123; Appendix B shows the detailed list of domains.

4.3 Data Collection and Data Sources

Data were collected from different sources in this project; the first one was the Census Bureau, from which census blocks data were downloaded (Figure 4-4).

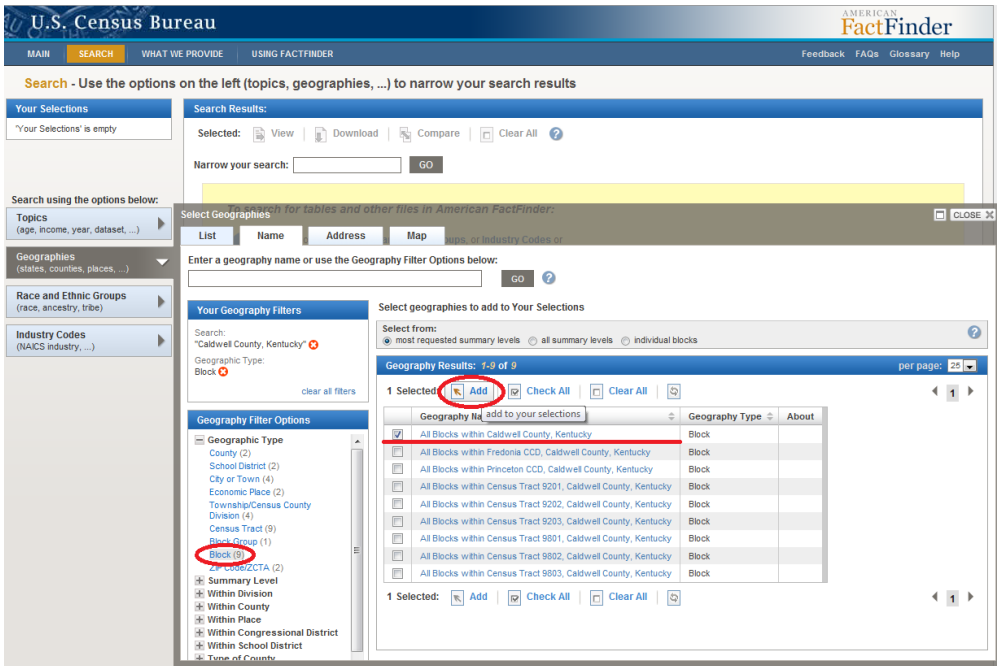


Figure 4-4 Downloading Census Data

For the cluster sampling process, the following demographic information has been required: 2010 Total Housing Units, 2010 Owner - occupied HU, and 2010 Renter - occupied HU. This information has been collected using the Esri's geo-enrichment service (Figure 4-5).

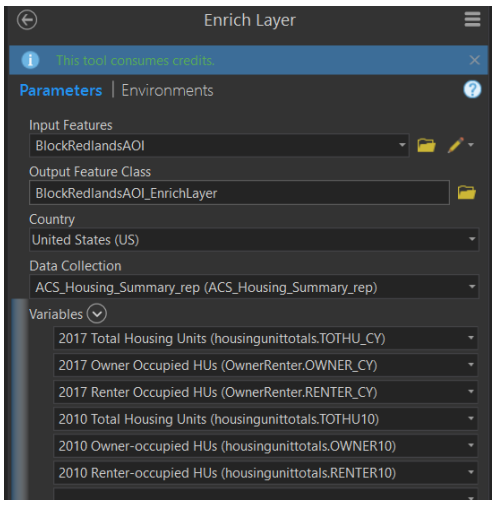


Figure 4-5 Geo - enriching data

Lastly, the survey instrument structure was taken from CDC. All the questions used in the surveys were extracted from the question bank published by the CDC in its CASPER toolkit.

All the data used in the system are licensed under the U.S. Government Work License. It is under this type of license that it is possible to work without any restriction on reproduction, derivative works, distribution, performance, or display of government work. Since changes are not being applied to the geometry of the elements, it is not essential to validate the original attributes of projection and scale. All the data generated is under the standard metadata ISO-19139 by internal regulations of the GIS program and is open license.

4.4 Data Scrubbing and Loading

The process of scrubbing and loading of data consisted of two main steps. The first was the creation of a geodatabase within the ArcGIS Pro project to contain all the necessary data for the system. This geodatabase is composed of 3 feature datasets (Figure 4-6): Base, which contains the base information of blocks nationwide, Planning, which contains the layers necessary for the process of selecting an area of interest in the system, and Results, which contains the resulting layers from the planning and analysis workflow.

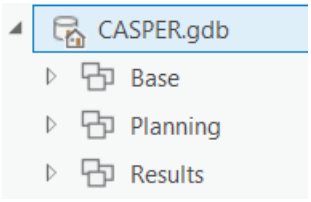


Figure 4-6 Geodatabase

The dataset downloaded from the Census Bureau webpage was in the shapefile format and the NAD 83 geographic coordinate system. The dataset was loaded to the geodatabase and projected to Web Mercator to optimize its efficiency when published in ArcGIS Online.

The second process was the design and publication of the survey layer in ArcGIS Online through Survey123 Connect.

type	name	label
begin group	Identification	① Identification and physical location
date	Date	A1. Date
text	TeamID	A2.Team
text	InterName	A3. Interviewer Name
text	BlockId	A4. Block ID
text	HU	A5. House Unit ID

Figure 4-7 Survey123 - Design

The process of designing and publishing the surveys included the design of the field names, aliases, as well as the domains (Figure 4-7). This entire process was carried out using the CASPER toolkit document published by CDC.

D5d. How many people living in your household aremore than 64 years old?		
 The total sum of people living in your house does not coincide with D3, please check the data entered	${P_LT2} + ${P_2to17} + ${P_16to64} + ${P_65} > ${ResAfter}	

Figure 4-8 Survey123 - Smart Rules

This process also included smart rules for data validation. These are logic rules to validate the consistency of the data entered and arithmetic rules to validate the correspondence between the numerical responses. Figure 4-8 shows an example where the answer is given for question: What is the total number of people in the household? Is compared with the sum of each data entered in the questions about the number of people by age range (less than two years, between 2 to 17 years, etc.).

4.5 Summary

This chapter describes the process of designing, implementing and loading the data that supports the CASPER system. The conceptual model shows how the business concepts relate to support the planning, fieldwork, and analysis process in CASPER. The logical model shows how these concepts are represented in the form of tables that support the CASPER system. Since this solution is based in ArcGIS Pro, the tables were implemented within a File Geodatabase. The data loading was quite simple due to the low variety of base data. However, the process of creating and implementing the survey

table required a considerable investment of time due to the exhaustiveness of the model survey created by the CDC.

Chapter 5 – Implementation

This chapter shows the complete implementation process of automating the CASPER methodology using GIS. The automation was implemented in three different phases. This chapter describes in detail how each of these components was developed and implemented. Section 5.1 focuses on the development of the Planning and Analysis module; this module was implemented with ArcGIS Pro through the creation of Python tools. The second phase of developing the Fieldwork module was described in Section 5.2, which included the configuration of a ready-to-use template in Survey123. Section 5.3 details the configuration of the control panels that show vital information for monitoring the status of the fieldwork according to CDC standards. This chapter ends with a summary of implementation.

5.1 Planning and Analysis Module

To develop the Planning and Analysis module, a series of specialized tools were created and organized into a toolbox as shown in Figure 5-1.

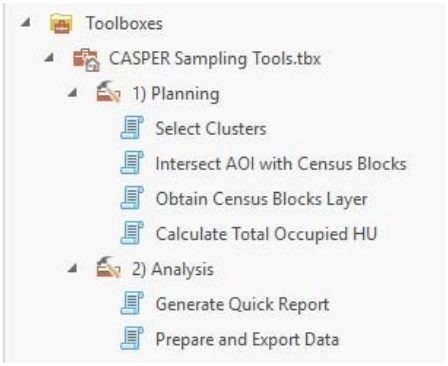


Figure 5-1 CASPER Tools

The planning tools were implemented with Python to automate the defined CDC CASPER workflow, which includes obtaining and integrating the census blocks data, reducing the study area according to the user's needs, geo-enriching the census block layer with the housing units variables, and pseudo-randomly selecting thirty blocks (clusters) for field work (Figure 5-2).

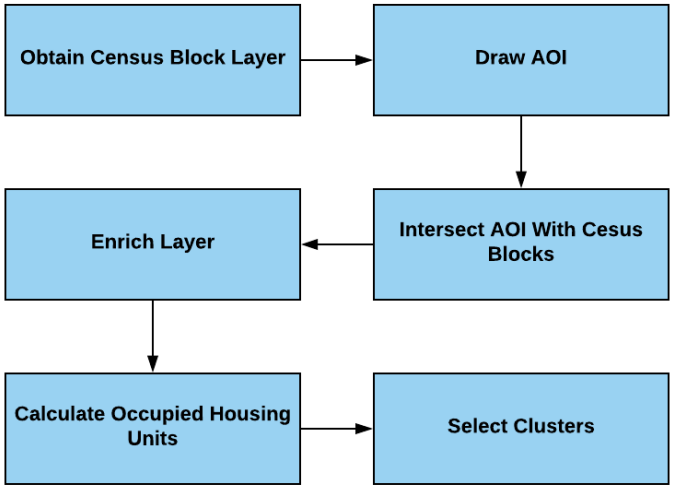


Figure 5-2 Planning Phase - CDC Documentation

The analysis tools allow the preparation of the data (calculating the weighting field) and the generation of an interoperable Excel file for the external analysis using Epi Info or SAS. The analysis tools also allow a quick estimation of the health needs of the community.

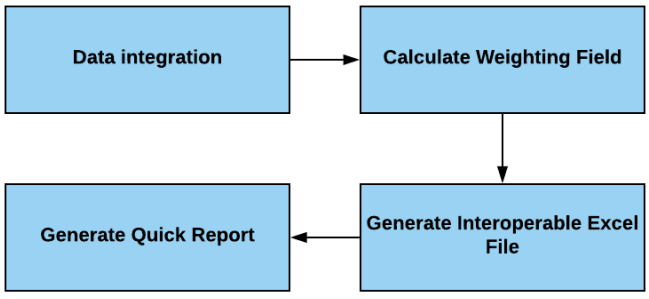


Figure 5-3 Analysis Phase - CDC Documentation

5.1.1 Python Development

For the development of the tools, Wing was chosen as the primary IDE. The configuration of the development environment was simple because the computer used had ArcGIS Pro installed, which also automatically installed Python 3. To configure the Python environment of the IDE, it was only necessary to reference the instance of Python 3 inside the ArcGIS Pro installed folder as shown in Figure 5-4.

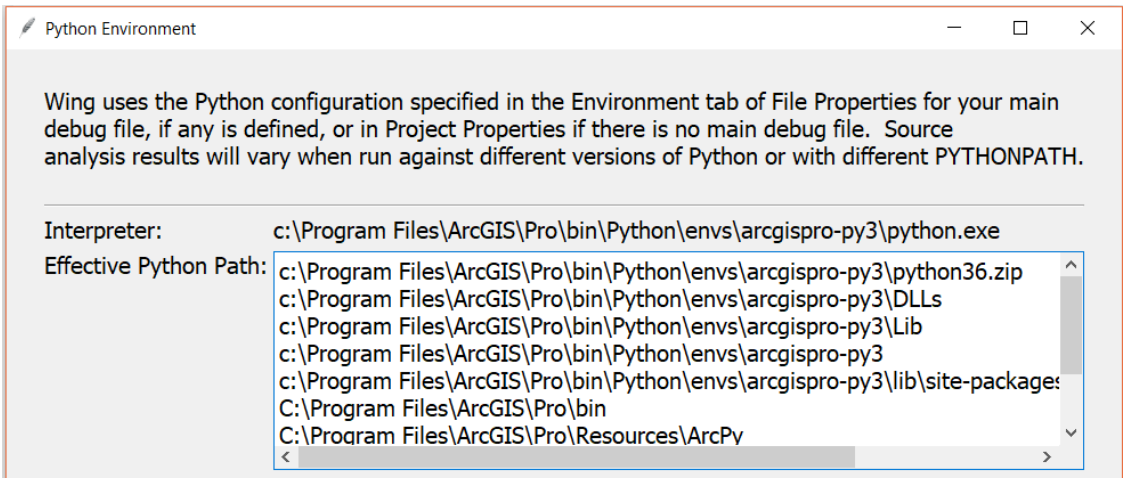


Figure 5-4 Wing IDE Configuration

Once the work environment was configured, the systematic development of the scripts was carried out to automate the CASPER steps. The first script was to obtain census blocks (clusters) for a defined study area. It is important to mention that the handling of variables in memory was vital to avoid the generation of intermediate layers during each of the processes (Figure 5-5). The detailed Python script can be found in Appendix A.

```
20
21  arcpy.AddMessage(arcpy.env.workspace)
22
23  output = arcpy.Intersect_analysis ([ layer , "Base/Countie "], 'in_memory\\output1')
24  fields = ['STATE_NAME', 'CNTY_FIPS','NAME']
25  State = ''
26  County = ''
27  Name = ''
```

Figure 5-5 Variables In-memory Handling

The second Python script was to reduce the area of study in case the CASPER analysis is not intended at the county level. It is important to note that this tool offers the possibility to choose between two options. The first option is to clip the downloaded census blocks to a user-defined geometry, either a user-drawn shape or a city limit layer. The second option, specifically requested by the client, is to maintain all the census blocks partially intersected with the user-defined study area. The output from this option will yield a study area that may extend outsides the user-defined shape. As such, the second option was implemented with the "intersecting" method, while the first option was implemented with the "clipping" method (Figure 5-6).

```
#Creating an temporal Feature Class
arcpy.AddMessage("Intersecting...")
#copy = arcpy.management.CopyFeatures(layer, "in_memory\\CopyEnriched")

if method == 'true':
    #Selecting
    arcpy.AddMessage("true...")
    inter = arcpy.SelectLayerByLocation_management (clusters, "INTERSECT", AOI)
    BlocksAOI =arcpy.CopyFeatures_management(inter, output)
    arcpy.SelectLayerByAttribute_management(clusters, "CLEAR_SELECTION")
else:
    #Clipping
    arcpy.AddMessage("Clipping...")
    BlocksAOI = arcpy.Intersect_analysis ([clusters, AOI], output)
    BlocksAOILayer= arcpy.MakeFeatureLayer_management (BlocksAOI, 'BlocksAOILayer')
    ClusterLayer= arcpy.MakeFeatureLayer_management (clusters, 'ClusterLayer')
```

Figure 5-6 Python Data Intersection

The third Python script emulates the pseudo-random selection of thirty clusters based on a user-chosen demographic variable. To explain the process, the total number of housing units is used as an example. First, a table of housing units was generated, each row representing one housing unit in a particular block. So the number of records equals the total number of housing units. In this table, a census block that has more housing units has more rows so that a random number generated later will be more likely falls into the row number range of this block.

Second, the random library that is part of the standard installation of Python 3 was imported. This library allows generating random numbers based on the Mersenne Twister algorithm. For detailed information about this library, please refer to official Python documentation (https://docs.python.org/3.7/library/random.html)

Using the random library, thirty random numbers within the range from 0 to the total number of housing unit were generated in the study area. If a random number of 201 is generated, the block that is corresponding to the row number of 201 will be selected. Compared to a block that has a fewer number of housing units, say 20, a block of 200 housing units will have ten times higher chance to include the generated random numbers. It is possible that the same random number could be generated multiple times and if that happens, the corresponding block will be selected multiple times. For example, if Block 1742 is selected twice, a number of surveys in this block will become fourteen instead of seven.

After the execution of the surveys process, the information must be analyzed to estimate health needs. To achieve this purpose, a set of analysis tools has been developed. The first tool developed was the data preparation and export. This tool calculates the weighting value to estimate the health needs. This calculation is made by using the following formula defined by CDC:

$$Weight = \frac{Total\ Number\ of\ Housing\ Units\ in\ the\ Study\ Area}{Total\ Interviews\ in\ Block * 30}$$

This formula is applied per each survey and the weight means how many actual housing units each sampled household represents in the study area. So, the sum of the weights will equal to the total housing units in the study area. For example, in a

community where there are 19370 housing units in total, and a cluster where seven surveys were conducted, all the interviews in this cluster will have the same weight of 92.23 calculated as follows:

$$W_i = \frac{19370}{7 * 30} = 92.23$$

This tool also generates the interoperable file that allows the user to analyze the results using a variety of software such as Epi Info or SAS. The development of this script required complex handling of lists and dictionaries as shown in Appendix A.

Finally, the last tool, Generate Quick Report, generates the estimation of a variable that the user selects. The process is based on the calculation of the frequency of the collected answers multiplied by the sum of weights associated with the answers. For example, in a sample of 210 interviews, if the user selects the question: What is your source of electricity? The tool will calculate the frequency of the answers for the unweighted distribution and the sum of the weights of each group of the answers for the weighted distribution (Figure 5-7).

	Unweighted		*Weighted*		
Characteristic	*Frequency*	*Percent*	*Frequency*	*Percent*	*95% CI*
Source of Electricity					
Power company	137	74.1	14190	74.0	61.9-86.0
Gasoline generator	29	15.7	3200	16.7	7.6-25.7
None	19	10.3	1789	9.3	3.8-14.8

Figure 5-7 Unweighted and Weighted Results' Distribution

It is important to mention that this tool does not seek to replace Epi Info and is only recommended for a preliminary analysis of the data. Epi Info has many additional tools for the statistical analysis of data.

5.1.2 ArcGIS Pro Project Components

Once the tools were created, two automatic workflows were created by using tasks in ArcGIS Pro. These workflows optimize the user experience by guiding him/her through each of the processes (Figure 5-8).

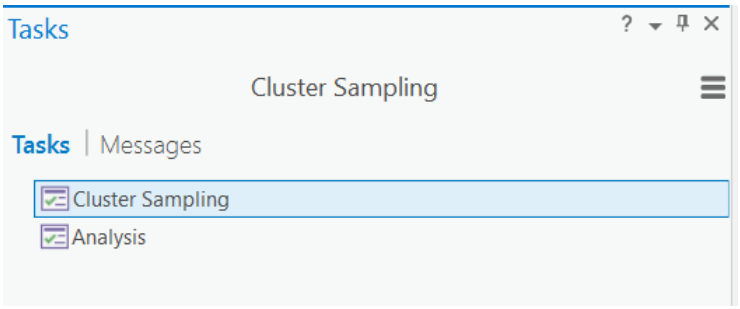


Figure 5-8 ArcGIS Pro Tasks

The first task integrates the planning tools and other complementary functionalities for the execution of the cluster sampling. The second task integrates the analysis tools for estimating CASPER results. The complete design of these tasks was performed using the Task Designer tool of ArcGIS Pro as shown in Figure 5-9.

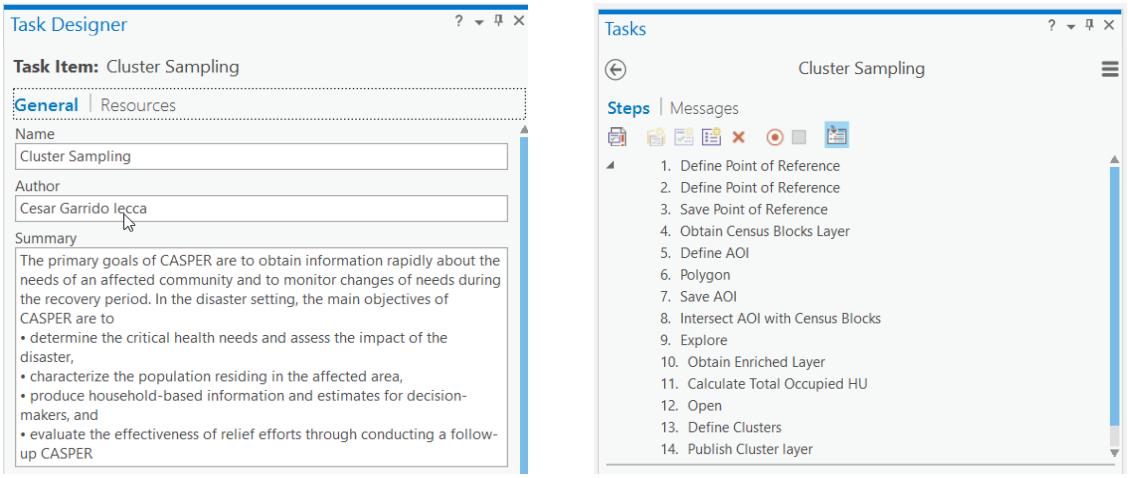


Figure 5-9 Task Descriptions

The Planning and Analysis module also includes a series of printing templates to facilitate the printing of the maps generated by each of the selected clusters.

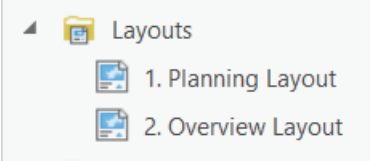


Figure 5-10 ArcGIS Pro Layouts

These templates were implemented through the ArcGIS Pro map series functionality. By using the attribute table of the block layer, this functionality is able to create a complete set of ready to print maps that include dynamic text as shown in Figure 5-11.

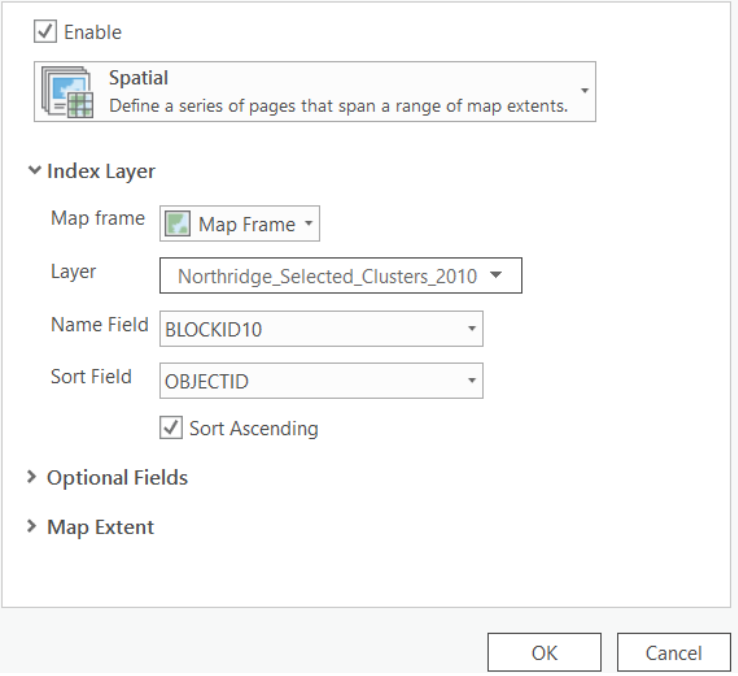


Figure 5-11 Layouts Configuration

5.2 Fieldwork Module

For this module, a template was developed to be deployed by using Survey 123. This template includes more than 180 questions taken from the CASPER data bank created by CDC. This process included the development of logic rules to ensure data quality and consistency. HTML styles and grouping of questions were also applied for the optimization of field surveys as shown in Figure 5-12.

D5	D5. Currently, how many people living in your household are:		
P_LT2	D5a. How many people living in your household are less than 2 years old?		
P_2to17	D5b. How many people living in your household are 2–17 years old?		
P_16to64	D5c. How many people living in your household are 18–64 years old?		
P_65	D5d. How many people living in your household aremore than 64 years old?		
Notification	 The total sum of people living in your house does not coincide with D3, please check the data entered	${P_LT2} + ${P_2to17} + ${P_16to64} + ${P_65} > ${ResAfter}	

Figure 5-12 Survey123 Configuration

Moreover, specific domains were generated for most of the questions. In this way, the possibility of incorrect data entry was minimized as shown in Figure 5-13

A	B	C
list_name	name	label
StayCh	Shelter	Shelter
StayCh	SpecialNeedsShelter	Special needs shelter
StayCh	Hotel/Motel	Hotel/motel
StayCh	Family/FriendHome	Family/friends home
StayCh	Other	Other
Race	AmericanIndian/AlaskanNative	American Indian/Alaskan Native
Race	Asian	Asian
Race	BlackorAfricanAmerican	Black or African American
Race	NativeHawOtherPacificIslander	Native Hawaiian or Other Pacific Islan
Race	Caucasian	Caucasian
Race	Unknown	Unknown
Race	Refused	Refused
Race	Other	Other

Figure 5-13 Survey123 Domains

Finally, the webhooks option was configured within Survey 123. Webhooks are HTTP events that are triggered after an event occurs. In our case, the event that triggers the HTTP call is the registration of a new survey. (Figure 5-14)

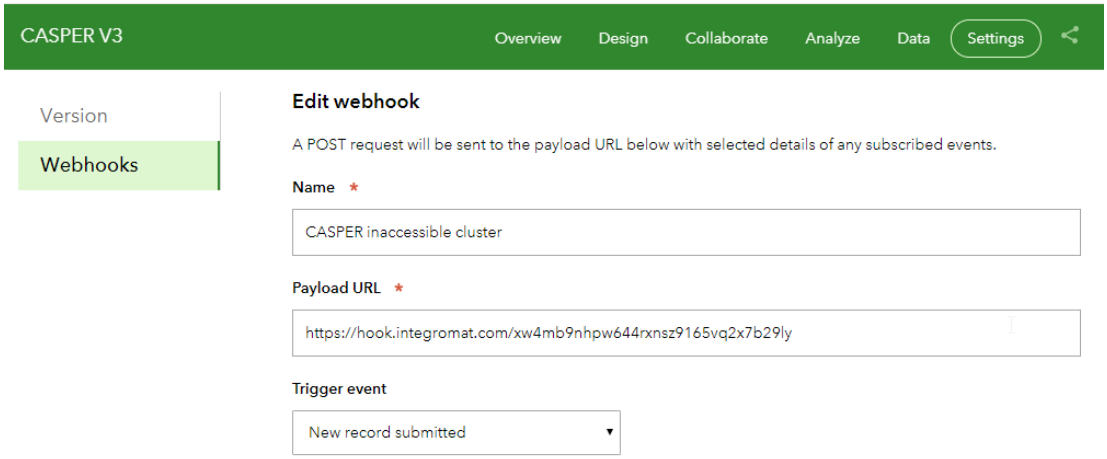


Figure 5-14 Survey123 Webhook

For the administration of the HTTP calls, Integromat was used that let user manage webhooks to interconnect different applications.

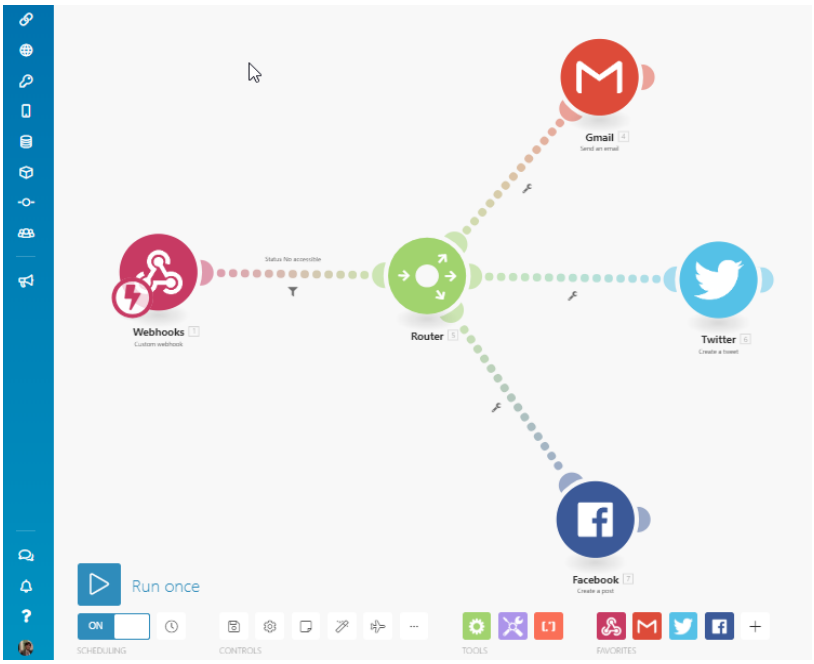


Figure 5-15 Integromat

Through the connection with Integromat, it was possible to send emails and alerts automatically in social networks every time the survey received presented the status field as not accessible (Figure 5-15).

5.3 Reporting Module

The reporting module was created with Operation Dashboards. Three types of dashboards were configured according to CDC standards. The first dashboard, CASPER Operational Status Dashboard, was designed to show the entire survey process. Three important rates were displayed, including the completion rate, the cooperation rate, and the contact rate. The calculations made for each of these rates are shown in Table 5-1.

Table 5-1 Rates defined by CDC Standards

Completion rate	$\frac{Number\ of\ Completed\ Interviews}{Number\ of\ Interviews\ goal\ (usually\ 210)}$
Cooperation rate	$\frac{Number\ of\ Completed\ Interviews}{All\ HUs\ Where\ contact\ was\ made)}$
Contact rate	$\frac{Number\ of\ Completed\ Interviews}{Number\ of\ HUs\ where\ contact\ was\ attempted}$

Various graphics were also configured to show the progress of each team, as well as the spatial distribution of completed and incomplete surveys (Figure 5-16)

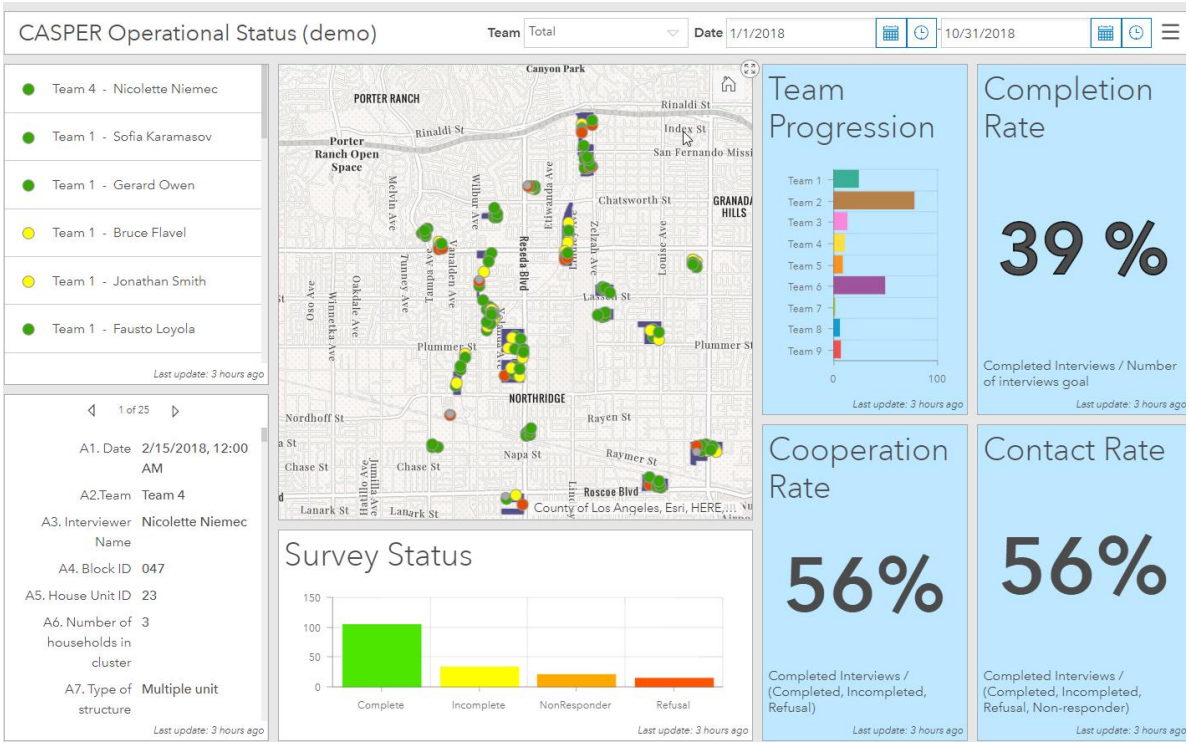


Figure 5-16 Operational Status Dashboard

Likewise, a demographic information control board was set up; this board shows the distribution of ages and races of the samples collected in the study area (Figure 5-17).

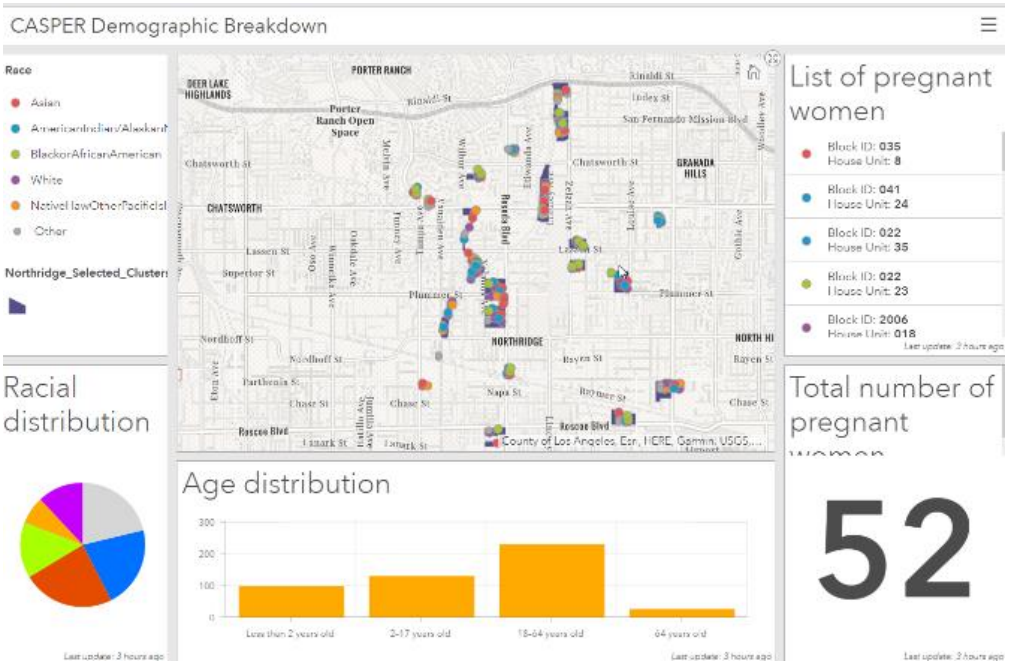


Figure 5-17 Demographics Dashboard

Finally, a medical emergency control board was designed to support the emergency reporting process during the execution of field surveys (Figure 5-18). This board shows the distribution of diseases in the area of study as well as emergencies recorded in the field.

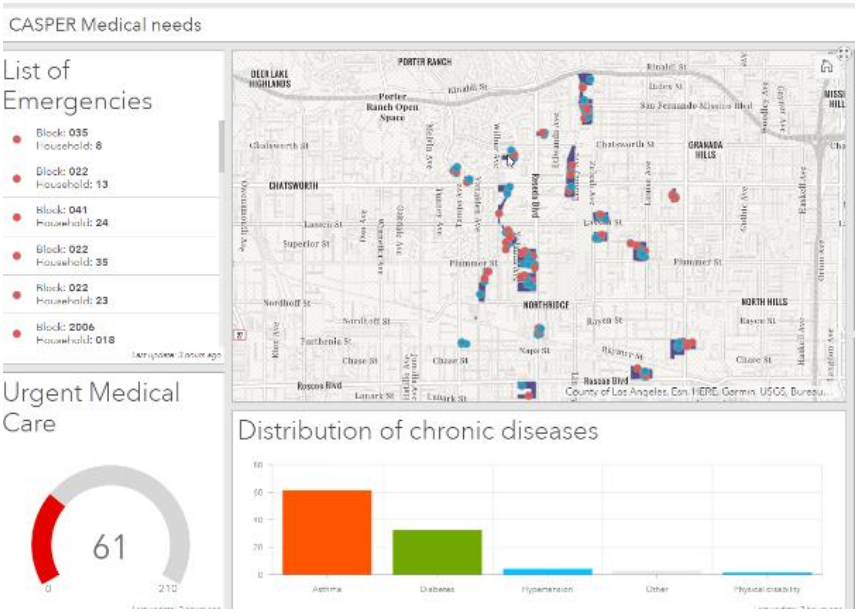


Figure 5-18 Demographics and Medical Emergency Dashboard

The step-by-step configuration of these boards is detailed in Appendix C "Manual of the Dashboards configuration."

5.4 Summary

This chapter describes the complete process of implementing the solution. The Planning and Analysis module was developed through the use of tools developed through Python and the integration of automated tasks in ArcGIS Pro. Python scripts automate the manual processes described in the CASPER guide published by the CDC. The Field Work module was designed under the standards of the CDC. For this module, it required an important investment of time for the digitalization of the questions, the creation of the domains and the data validation rules. Finally, the Report module consisted of the configuration of control boards that provide valuable information for health officers. For the implementation of these boards, three control boards were configured, which collect the main concerns expressed by the stakeholders.

Chapter 6 – Results and Analysis

This chapter describes the results of the project from the point of view of the end user. In particular, the use of each of the modules is detailed, and the results from the analyses are discussed. A summary of the results can be found at the end of the chapter.

6.1 Planning Module

The Planning module, developed on the ArcGIS Pro platform, automates the tasks of downloading, processing, and integrating the necessary data for the CASPER methodology (Figure 6-1). More importantly, this module enables the users to pseudo - randomly sample 30 clusters and output the sampling result for the following fieldwork.

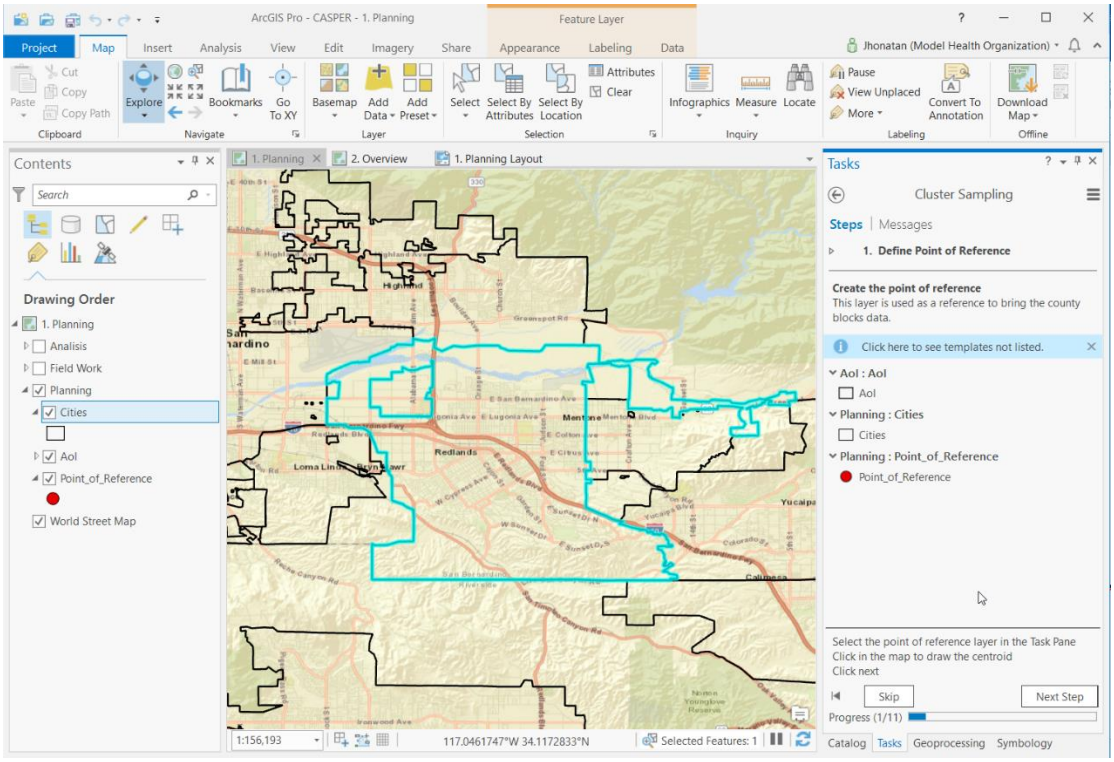


Figure 6-1 Planning Module

To start the planning process, the user first specifies a location of the reference. For example, the LA County Health Department needs to run CASPER to estimate the community health needs in the city of Northridge. Through any planning process, the GIS user will be able to integrate the census block information for the entire county of interest, a specified AOI, or a specified city. In this example, the user selects the choice of City Boundary, then the user selects Northridge city in the map view. As a result the census blocks intersecting Northridge are automatically selected.

Meanwhile, the CASPER solution also allows the integration of additional variables into the planning process (Figure 6-2). These variables greatly enrich the understanding of the situation of the study area. For example, the GIS specialist in Northridge can enrich the study area layer with the "2018 Total Housing Unit" Esri projection or the "ACS census information." In this way, the expert user can compare a variety of values with the official data of the 2010 Total Housing Unit and verify if any modification is necessary for the selection of clusters. Also, variables such as total population of children (Age < 14) or Senior (Age 65+) would be important for the planification improvement.

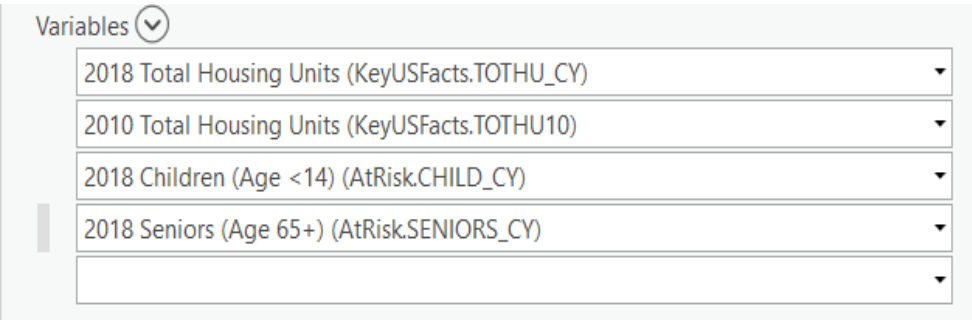


Figure 6-2 Planning Variables

After selecting the variable, the GIS specialist will follow the task flow to pseudo-randomly select 30 clusters (Figure 6-3 Cluster Selection).

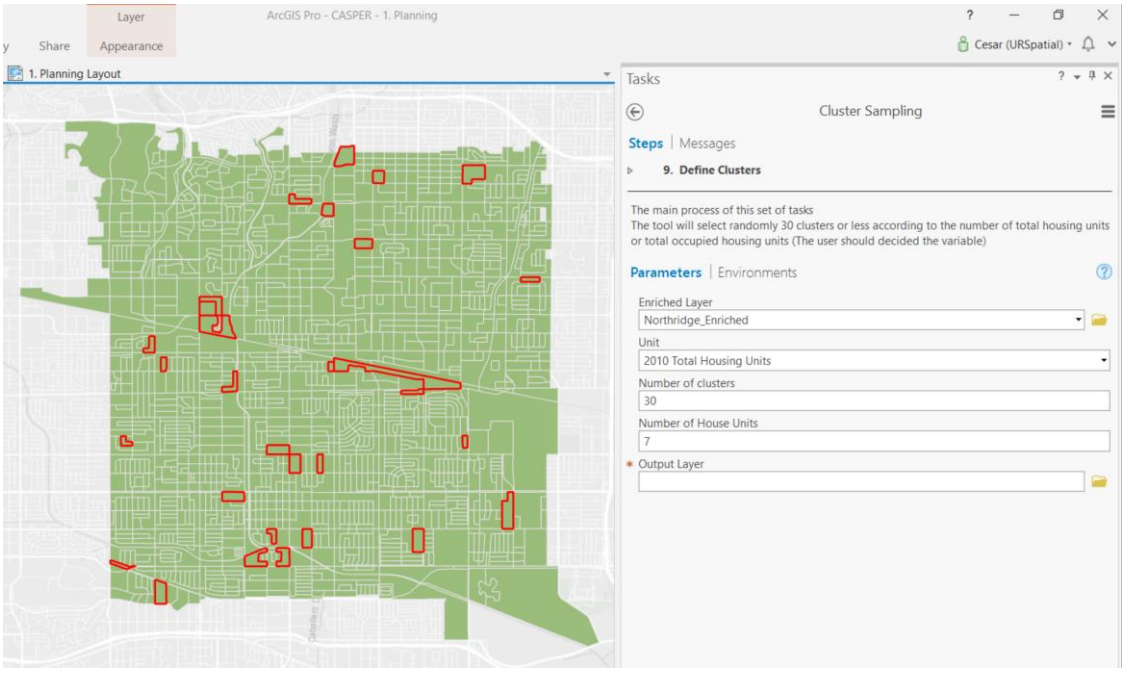


Figure 6-3 Cluster Selection

The Python tools are automatically executed to emulate the pseudo-random selection process recommended by the CDC.

Finally, the Planning module publishes the sampling result to the organizational portal of ArcGIS Online or Portal for ArcGIS. Any mobile or web client can then consume these layers. This significantly reduces the cost of printing hardcopy maps. However, the automatic printing of the maps corresponding to each selected block has been considered as an optional process (Figure 6-4).

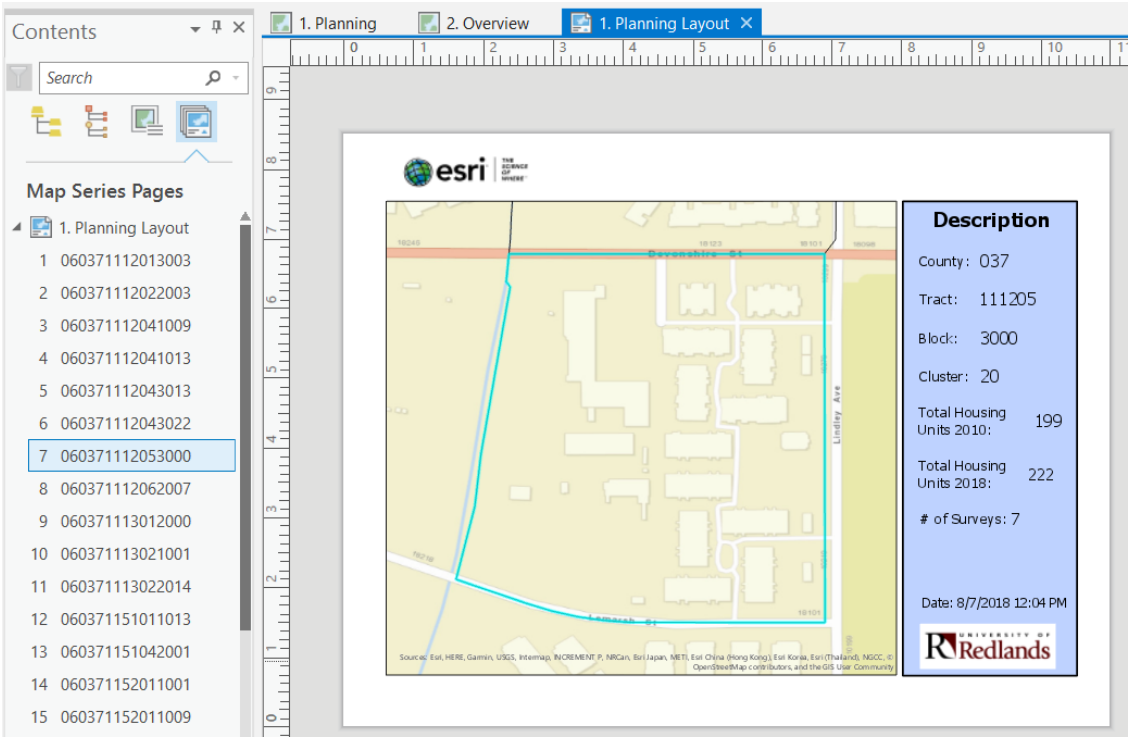


Figure 6-4 Data-Driven Printing

This process is done through the data-driven pages workflow. In this example, it is only necessary to use the ready-to-deploy template included in the solution for generating ready to print maps. It is possible that multiple selected clusters might be displayed when they are closed to each other. It is recommended that the user filter the cluster layer with the block ID of interest. In the future when ArcGIS Pro advances, this function can be integrated into the workflow automatically.

Compared to the original CASPER planning workflow, this automated planning process results in a great reduction in time. According to the tests carried out by Tracy Haywood, one of the critical reviewers of the current CDC CASPER guide, “the automated CASPER solution reduces the planning process time by three fourths." (e.g., from 2 days to half a day)

6.2 Field Work Module

Continuing with the earlier example; the user now needs to design the survey instruments before they can collect the data in the field. The Fieldwork module automates the digital survey creation process. The CASPER template has more than 180 standard questions that facilitate the survey design process. Figure 6-2 shows the sample questions included in this assessment.

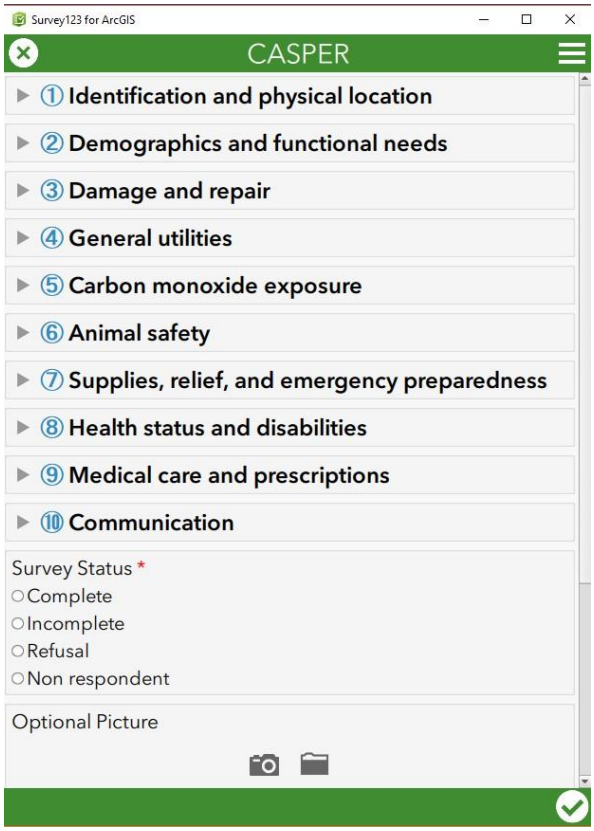


Figure 6-5 Field Work Module

The benefits of having a digital survey instrument are obvious. First, the printing cost of paper surveys is eliminated, which also leads to a reduction in time used.

Second, users can modify, delete, or add any questions easily. Editing the survey instrument can be done either during the design of the survey or even after the survey has been published and shared with all field users. This is a key advantage since the past practice of reprinting the modified surveys is time-consuming and costly. In addition, answers to the survey questions can be validated when they are entered. The CASPER template includes dozens of automatic validations as well as domain values that considerably enhance data quality.

Moreover, digital surveys can show or hide the conditioned questions based on the answers to the previous questions. In this way, the interviewer does not need to spend time validating whether a question applies or not in each case.

Finally, since these smart surveys are integrated with Integromat. The user in the field can report an emergency to the responsible parties by submitting the survey. In our example for the City of Northridge, a fieldworker had problems accessing the block 456 (Figure 6-6). He was able to report the problem by submitting the report in Survey123; the software triggered an email to the GIS Analyst in the office to request another block to survey.

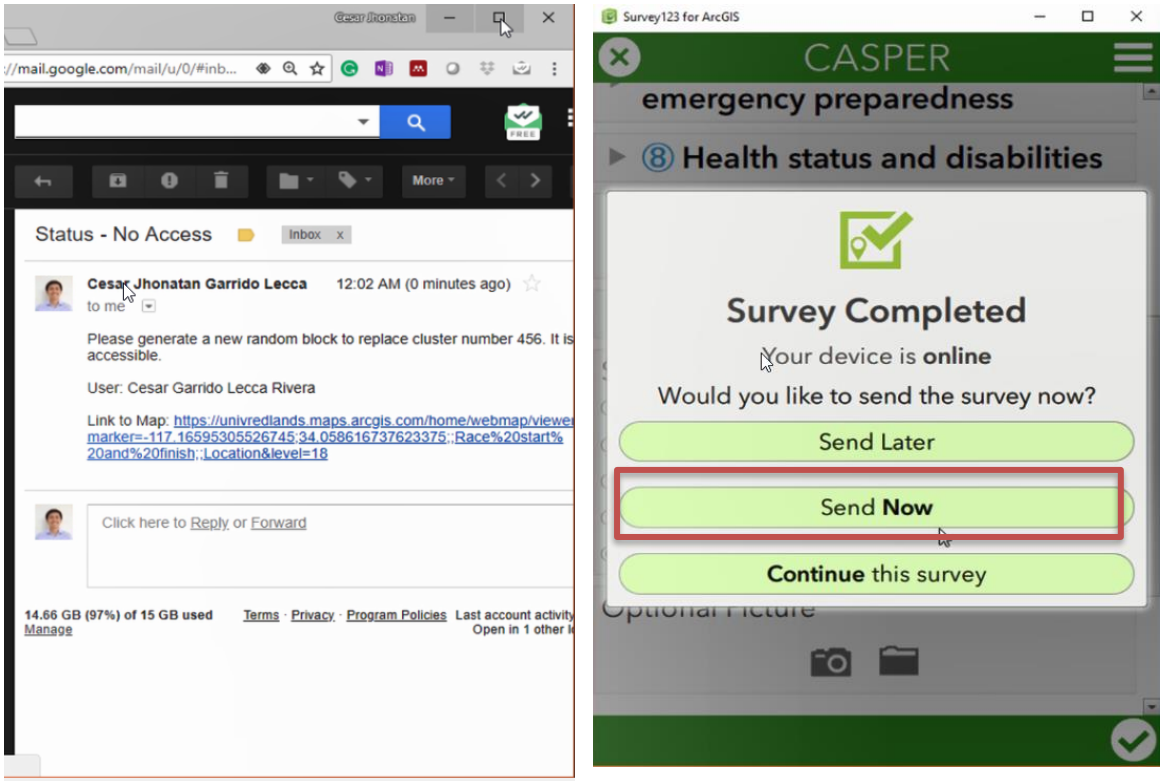


Figure 6-6 Integromat Integration

6.3 Reporting Module

The Reporting module allows officers to have an operational awareness of the status of the surveys under CDC standards. Figure 6-7 shows the Operational Status dashboards for the City of Northridge.

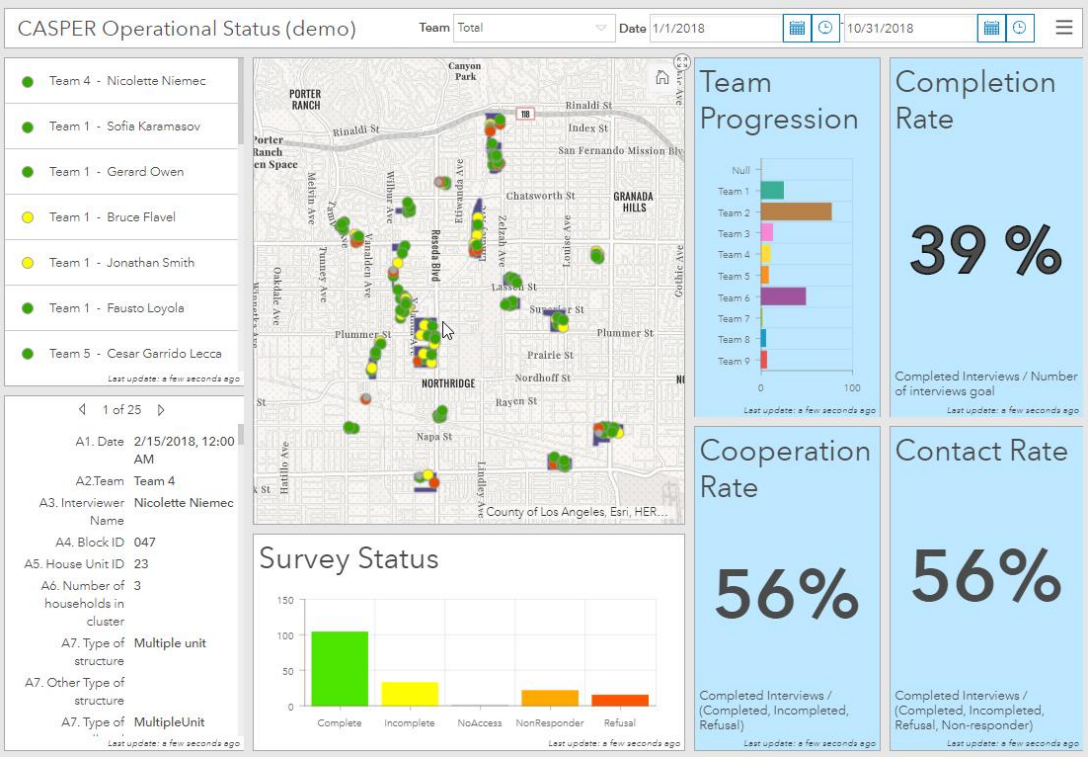


Figure 6-7 Operational Status Dashboard

The completion, cooperation, and contact rates give the office staff a clear idea not only of the progress of the surveys in the field but the behavior of the respondents. In this example, people at the L.A. Health Department can see that 39% of the total of 210 interviews in the study area have been completed in general. This also shows the distribution of the surveys according to their status (e.g., Completed, Incomplete, Refusal, Non-Responders and No Access). This allows the stakeholders and analysts to decide proactively on the need to change a block if some team has access or security problems to complete surveys.

Likewise, these tools allow the evaluation of the performance of each team in the field, allowing the data to be filtered by each team or the date of survey. (Figure 6-8)

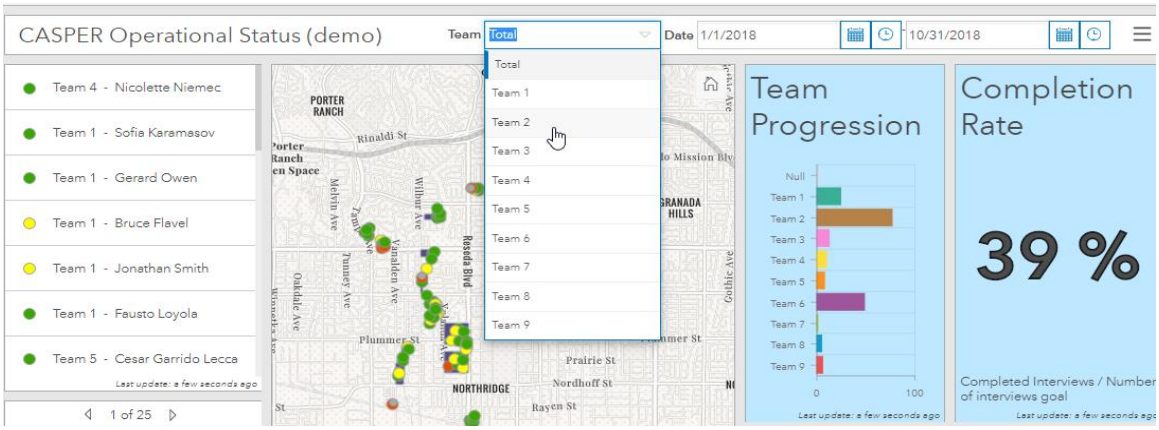


Figure 6-8 Filtering

The Demographic Breakdown dashboard allows the users in the office to understand better the sample of their data during its capture (Figure 6-9).

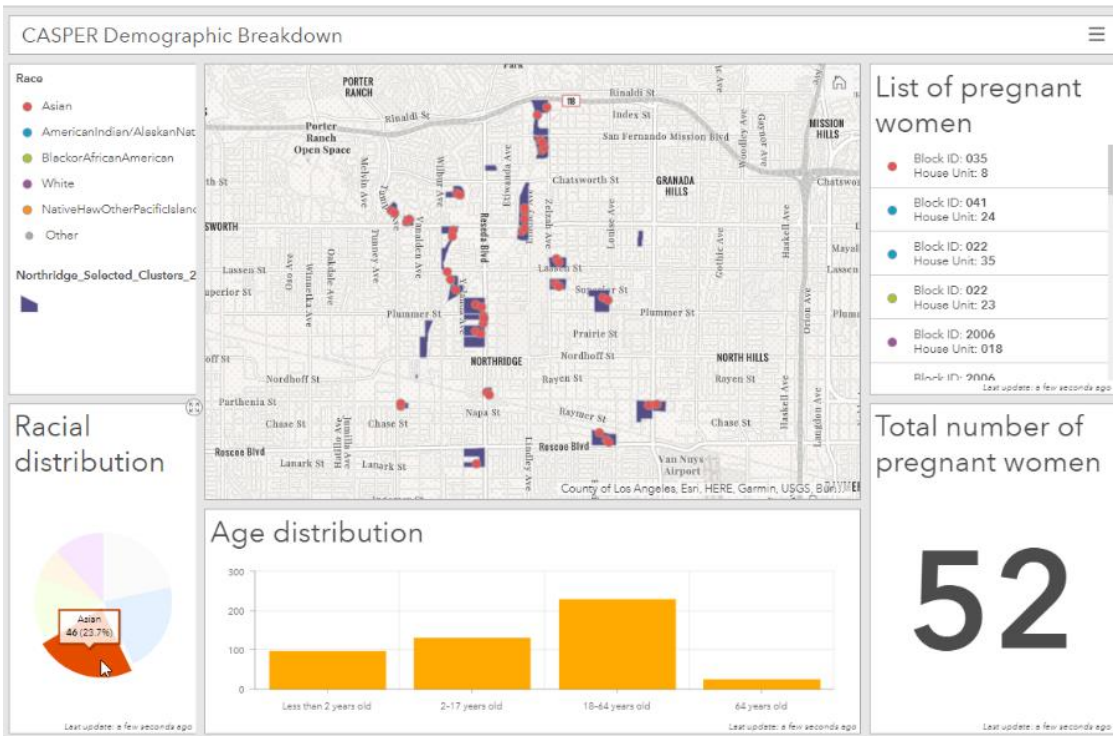


Figure 6-9 Demographic Breakdown Dashboard

In this case, the stakeholders of Northridge can see the age and racial distribution in the area of study. Moreover, the number of households that have pregnant women is recorded and spatially mapped in the dashboard.

This type of information is valuable to evaluate the effectiveness of the surveyors and propose strategies that reduce the percentage of incomplete or refusal surveys.

Finally, the CASPER Medical Needs dashboard provides the fieldwork teams with the ability to report health emergencies (Figure 6-10). This is undoubtedly an auxiliary process of value in post-emergency scenarios where the CASPER process can generate additional value for the community.

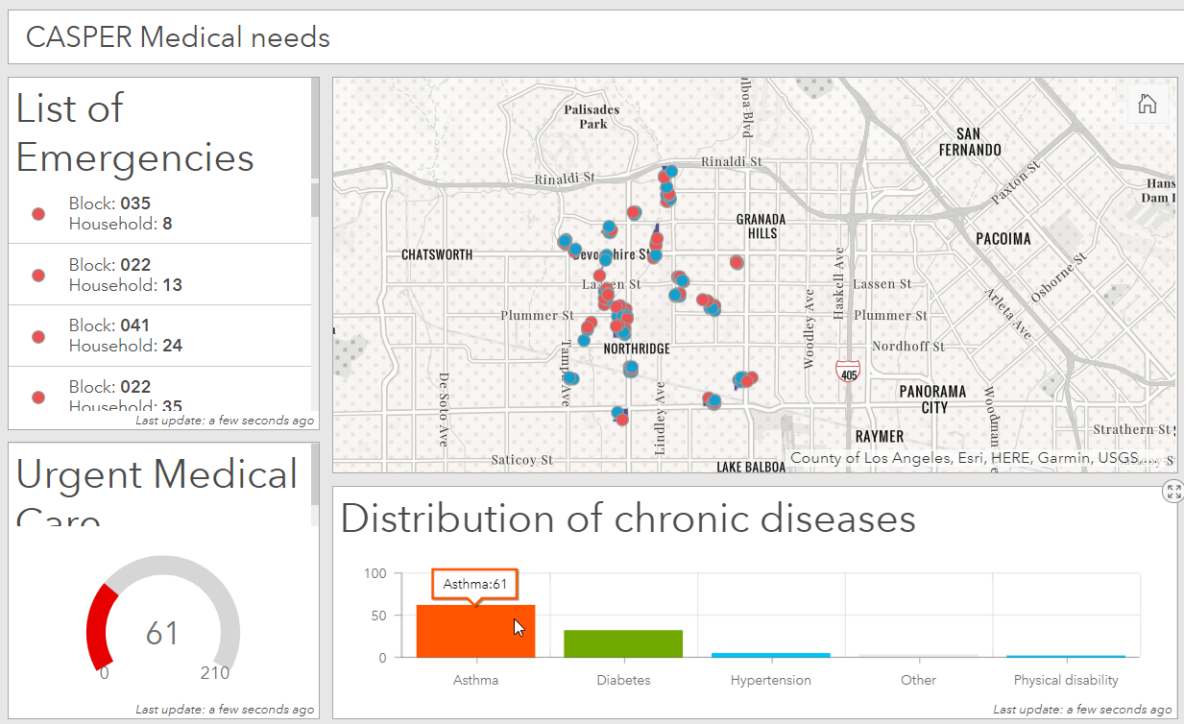


Figure 6-10 Medical Needs Dashboard

In the case of the Northridge, a chronical diseases distribution chart is shown. The chart shows that more than 60 surveys reported the signs of asthma in the region. This is probably caused by poor air quality in the area.

6.4 Analysis Module

Once the surveys were collected, the health needs of the community can be estimated. In this project, a few Python tools were developed to allow the users to prepare the data for statistical software like Epi Info or SAS or conduct the frequency analysis directly.

In the past, a manual calculation of the weight field was required for each cluster. By automating this step through Python, the data preparation process is much streamlined, and the non-technical experts can also carry out this task. With the tool Prepare and Export Data (Figure 6-11), an Excel file with an extra weight field is generated, which allows the expert user to use the software of their preference to conduct further analysis.

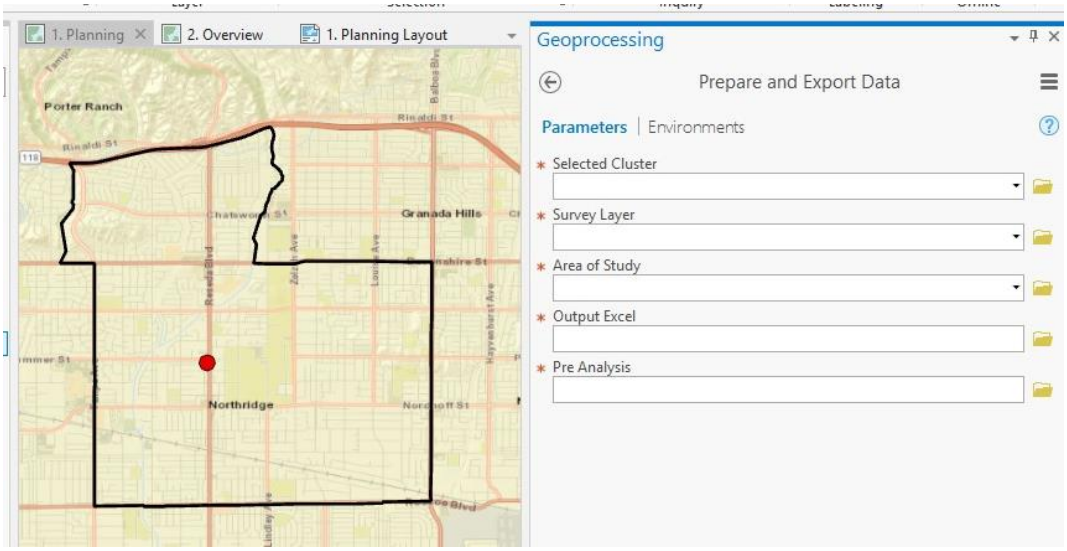


Figure 6-11 Prepare and Export Tool

For the Northridge use case, a file was generated from the collected surveys and exported into Epi Info. The results of the analysis show an estimate of 50% of the population in the community that could potentially have asthma symptoms (Figure 6-12).

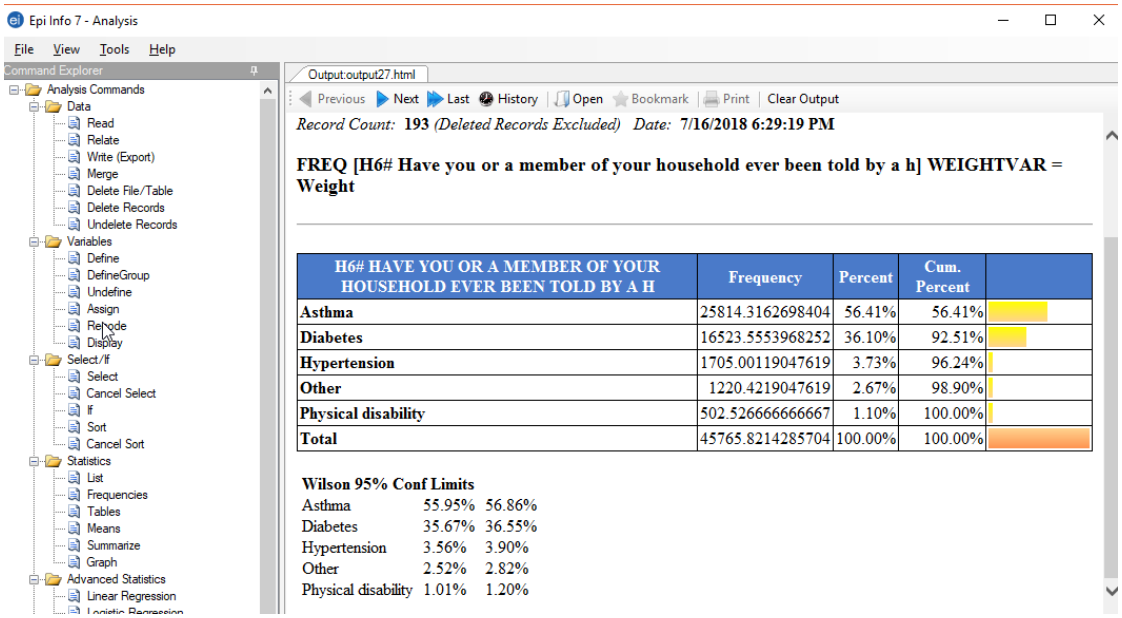


Figure 6-12 Epi Info Analysis Result

Alternatively, the GIS analyst could use the second analysis tool for a rapid estimation of the frequency of a selected variable (Figure 6-13). This tool allows users who do not have the statistical software installed on their computer to generate preliminary reports.

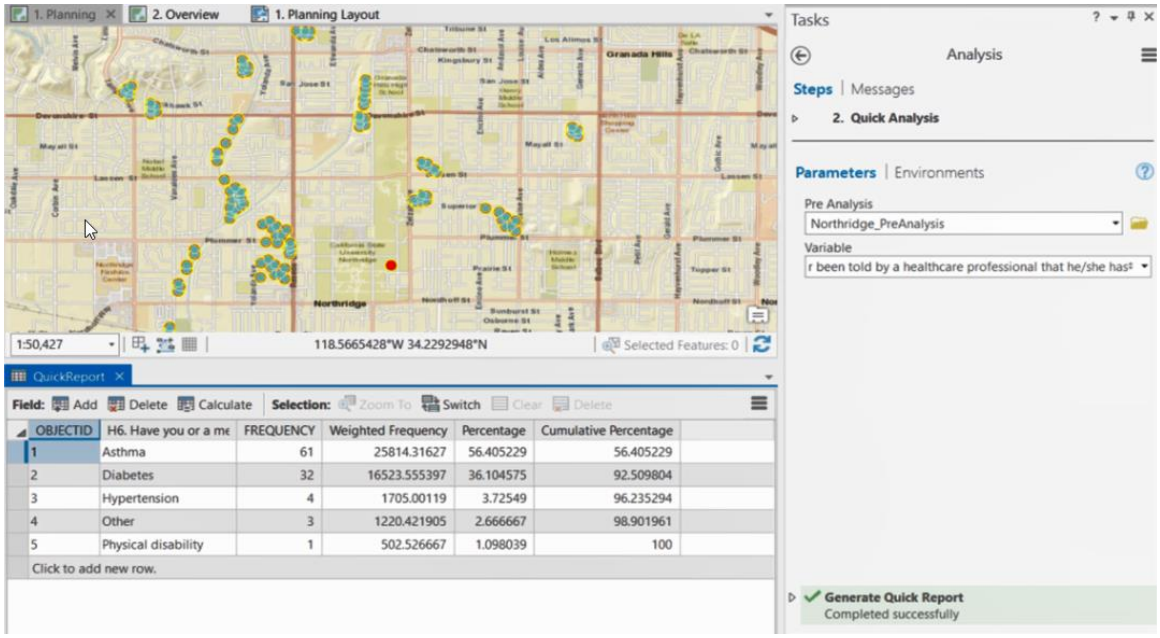


Figure 6-13 Quick Report Tool

6.5 Summary

In this chapter, the usability and results concerning benefits for users have been shown. In general, the user has a complete GIS-based framework to execute the main CASPER processes, including planning, fieldwork, operational awareness and data analysis. Each of these processes has considerable benefits regarding time, cost, and process improvement.

Chapter 7 – Conclusions and Future Work

This thesis has presented the complete development of a GIS-based framework for implementing the CASPER methodology more efficiently. This is a valuable tool for any health agency that needs to execute CASPER in post-emergency scenarios or in cases of executing a community health status analysis in a preventive manner. This tool is part of the set of initiatives of Esri for the Health sector. For the execution of the project, the agile SCRUM methodology was chosen, which facilitated frequent communication with the stakeholders. Regular communication was key to the process of gathering requirements and estimating the scope and execution schedule due to the size of the project and a large number of stakeholders involved. The iterative development of each of the components allowed different users to test the solution in different phases and request minor changes to improve the performance of the solution. This has allowed generating a final solution highly accepted by the end users. As Este Geraghty, Esri's Chief Health Officer, mentions, “this new solution has exceeded user expectations, allowing them to have a complete framework for the execution of CASPER.”

It is important to mention that despite having initial communication problems with the CDC, the management of this threat was carried out successfully, initiating an immediate collaboration with various health agencies in Texas, California, etc. On the other hand, initially, there was an underestimation of the complexity of the scripts to be developed. Fortunately, there was advice from the Esri Pattern and Practices team who provided an important guide to define an effective approach in the development of the scripts.

Dr. Eric Brenner, an epidemiologist and one of the main references for Rapid Health Assessment for CDC and WHO, mentioned that the new CASPER solution communicates the right message to high-level staff.

7.1 Conclusions

CASPER methodology is widely used by health agencies in the United States and the world. Being a key methodology for emergency response and the creation of public policies, the impact of the results is great. The GIS-based framework for the CASPER solution has gone through a comprehensive process of testing by various health professionals and government agencies trying to contribute successfully to the reduction of time, investment, and complexity. Finally, this solution was approved for use in the county of Williamson, Texas for the next execution of CASPER on September 14 and 15, 2018.

The tests carried out on the solution have yielded results such as a reduction of three-quarters of the average time invested in manual tasks within the planning process. There has also been a reduction of at least half of the time invested in the process of quality control and data loading of the fieldwork, such as reducing the cost of printing maps and surveys. Finally, the solution offers the ability to establish stable communication between the field workers and the interested parties.

7.2 Future Work

During the taking of requirements, development, and implementation of the project, a series of potential opportunities for improvement were identified. However, due to time

constraints, this project was limited to executing those that were selected in team agreement. Those requirements not selected were documented in a future list of enhancements for the development of the next realities of the solution. Next, the main functionalities discussed as future enhancements of the project will be described.

The first significant improvement in the project is the inclusion of a complete crew management system for the Fieldwork module. At present, the solution has a Survey123 template that allows the rapid deployment of smart forms based on CDC standards. However, the management of the crews, the assignment of tasks, and the integrated operations control would be a significant improvement for the Field Work process.

Second, this project did not consider systematizing the selection process of housing units for each block. It has been considered to apply machine learning tools for the automatic identification of housing units using the Esri's basemaps such as Esri satellite images, as well as the vector elements of Open Streetmap. However, this improvement requires a considerable investment of time or even the start of an alternative project.

A third important enhancement for this project would be the migration of analysis tools to a completely online environment through the use of Jupiter notebooks and the ArcGIS API for Python. This implementation would allow a more efficient analysis and directly connect to the Survey Service feature created by Survey123.

Finally, there are a variety of minor improvements proposed for the next version of the solution, such as the inclusion of webhooks within Survey123 that trigger alerts through emails and SMS in specific cases. Also, the creation of a support application for additional quality control of the selected data, and finally the creation of a final report application that summarizes the results of the CASPER estimates.

Works Cited

Bradt, D. A., & Drummond, C. M. (2002). Rapid epidemiological assessment of health status in displaced populations--an evolution toward standardized minimum, essential data sets. *Prehospital and Disaster Medicine*, *17*(4), 178–185. https://doi.org/10.1017/S1049023X00000479

Centers for Disease Control and Prevention. (2012). *Community Assessment for Public Health Emergency Response (CASPER) Toolkit*. *Atlanta (GA): CDC Centers for Disease Control and Prevention*. Retrieved from http://emergency.cdc.gov/disasters/surveillance/pdf/casper_toolkit_version_2_0_508_compliant.pdf

Das, K. , Lashkari, R. S. Biswas, N. (2013). Disaster Assessment and a Humanitarian Logistics Based Mitigation Planning. *The Asia Pacific Industrial Engineering & Management Systems Conference 2012*, (December), 278–289. https://doi.org/10.7232/iems.2013.12.4.336

de Ville de Goyet, C., & Lechat, M. F. (1976). Health Aspects in Natural Disasters. *Tropical Doctor*, *6*(4), 152–157. https://doi.org/10.1177/004947557600600404

Hazzan, O., & Dubinsky, Y. (2014). *The Agile Manifesto*. *SpringerBriefs in Computer Science*. https://doi.org/10.1007/978-3-319-10157-6_3

Henderson, R. H., Davis, H., Eddins, D. L., & Foege, W. H. (1973). Assessment of vaccination coverage, vaccination scar rates, and smallpox scarring in five areas of West Africa. *Bulletin of the World Health Organization*, *48*(2), 183–194.

Henderson, R. H., & Sundaresan, T. (1982). Cluster sampling to assess immunization

coverage: A review of experience with a simplified sampling method. *Bulletin of the World Health Organization*, *60*(2), 253–260.

Kaiser, R., Spiegel, P. B., Henderson, A. K., & Gerber, M. L. (2003). The Application of Geographic Information Systems and Global Positioning Systems in Humanitarian Emergencies: Lessons Learned, Programme Implications and Future Research Centers for Disease Control CDC and Prevention (CDC). *Disasters*, *27*(2), 127–140. https://doi.org/Doi 10.1111/1467-7717.00224

Kusiak, A., & Tang, C. Y. (2006). Innovation in a requirement life-cycle framework. *Proceedings of the 5th International Symposium on Intelligent Manufacturing Systems, IMS*, 61–67. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.74.8988&rep=rep1&type=pdf

MacEachren, A. M., Jaiswal, A., Robinson, A. C., Pezanowski, S., Savelyev, A., Mitra, P., … Blanford, J. (2011). SensePlace2: GeoTwitter analytics support for situational awareness. *VAST 2011 - IEEE Conference on Visual Analytics Science and Technology 2011, Proceedings*, (January 2015), 181–190. https://doi.org/10.1109/VAST.2011.6102456

Malilay, J., Flanders, W. D., & Brogan, D. (1996). A modified cluster-sampling method for post-disaster rapid assessment of needs. *Bulletin of the World Health Organization*, *74*(4), 399–405.

Morton, M., & Levy, J. L. (2011). Challenges in Disaster Data Collection during Recent Disasters. *Prehospital and Disaster Medicine*, *26*(03), 196–201.

https://doi.org/10.1017/S1049023X11006339

Navathe, S. B. (1992). Evolution of data modeling for databases. *Communications of the ACM*, *35*(9), 112–123. https://doi.org/10.1145/130994.131001

Nusser, S. M., Miller, L. L., Goodchild, M. F., & Clarke, K. (2001). Future views of field data collection in statistical surveys. *Proceedings of Dg.o 2001 National Conference on Digital Government Research*, (January 2001). Retrieved from https://www.researchgate.net/profile/Keith_Clarke2/publication/228975189_Future_views_of_field_data_collection_in_statistical_surveys/links/0912f5135647272e23000000.pdf

Platz, L., Huebner, E., Marlow, D., Moeller, R., Price, J., Richardson, M., … Nalen, C. Z. (2017). *Final Report : Zika Virus Community Assessment for Public Health Emergency Response (CASPER)* (Vol. 3660). Williamson County. Retrieved from http://www.wcchd.org/about_us/docs/WCCHD Zika CASPER Report_Final.pdf

Randell, A. A., Spellman, E., Ulrich, W., Wallk, J., Clark, R. M., Elliott, E., & Melaragno, W. (2014). Leveraging Business Architecture to Improve Business Requirements Analysis, (March). Retrieved from https://cdn.ymaws.com/www.businessarchitectureguild.org/resource/resmgr/public_resources/LeveragingBusinessArchitectu.pdf

Schwaber, K. (1997). SCRUM Development Process. *Business Object Design and Implementation*, (April 1997), 117–134. https://doi.org/10.1007/978-1-4471-0947-1_11

Schwaber, K. (2013). What Is Scrum? *Volaro*, 5. Retrieved from

http://www.volaroint.com/wp-content/uploads/dlm_uploads/2014/03/DC-VOLARO-Training-Scrum-What_Is_Scrum.pdf

Serfling, R. E., & Sherman, I. L. (1965). *Attribute sampling methods for local health departments (No. 1230).* Atlanta , Georgia: US Dept. of Health, Education, and Welfare, Public Health Service, Communicable Disease Center, Epidemiology Branch.

Tomlinson, R. F. (2007). *Thinking about GIS: geographic information system planning for managers*. Redlands, CA: Esri Press.

Watt, J. (1977). The role of the services in accident and disaster. *Injury*, *8*(4), 250–259. https://doi.org/10.1016/0020-1383(77)90097-3

World Health Organization. (1990). *Rapid Health Assessment in Epidemics: First Steps*. Geneva, Switzerland: World Health Organization. Retrieved from https://scholar.google.com/scholar?hl=en&as_sdt=0%2C5&q=World+Health+Organization%3A+Rapid+Health+Assessment+in+Epidemics%3A+First+Steps.+Geneva%2C+Switzerland&btnG=

World Health Organization. (2004). *Developing Health Management Information Systems*. *Developing health management information systems: a practical guide for Developing Countries*. Geneva, Switzerland: World Health Organization.

Young, S., Sanchez, C., & Malilay, J. (2005). Application of Geographic Information Systems and Global Positioning Systems for Public Health Rapid Needs Assessments in Disaster Settings.

Zane, D. F., Bayleyegn, T. M., Haywood, T. L., Wiltz-Beckham, D., Guidry, H.,

Sanchez, C., & Wolkin, A. F. (2010). Community assessment for public health emergency response following Hurricane Ike---Texas, 25-30 September 2008. *Prehospital & Disaster Medicine, 25*(6), 503–510. Retrieved from http://proxy.library.adelaide.edu.au/login?url=http://search.ebscohost.com/login.aspx?direct=true&db=c8h&AN=104982296&site=ehost-live&scope=site

Appendix A. Python Scripts

```
##---------------------------------------------------------------------------------------
```
CASPER Select County - Cesar Garrido Lecca - University of Redlands

```
import arcpy
import os

#Parameters
layer = arcpy.GetParameterAsText(0)
outputfile = arcpy.GetParameterAsText(2)

#Finding the Workspace
d = arcpy.Describe(layer)
work = d.path

#Set Enviroment from the Parameter
text2 = work.split('.gdb')
text3 = text2[0] + '.gdb'
arcpy.env.workspace = text3
arcpy.env.overwriteOutput = True

arcpy.AddMessage(arcpy.env.workspace)

#Intersecting Point of reference with county layer
output = arcpy.Intersect_analysis ([ layer , "Base/Counties"], 'in_memory\\output1')
fields = ['STATE_NAME', 'CNTY_FIPS','NAME']
State = ''
County = ''
Name = ''

with arcpy.da.SearchCursor(output, fields) as cursor:
    for row in cursor:
        State = row[0]
        County = row[1]
        Name = row[2]
arcpy.AddMessage([State,County,Name])

#Creating a new layer of blocks layer for the specific county selected.
featureclasses = arcpy.ListFeatureClasses(feature_dataset='Base')
for fc in featureclasses:
    if fc == State:
        path = os.path.join(arcpy.env.workspace,fc)
        arcpy.AddMessage(path)
        arcpy.AddMessage("COUNTYFP10 = '" + County +"'")
```

```
 county = arcpy.MakeFeatureLayer_management (path, "County", "COUNTYFP10
          = '"+ str(County)+"'")
 copy = arcpy.management.CopyFeatures(county, outputfile)

arcpy.SetParameterAsText(1,copy)
```

```
##---------------------------------------------------------------------------
```

CASPER Cluster Intersection - Cesar Garrido Lecca - University of Redlands

```
import arcpy

#Parameters
clusters = arcpy.GetParameterAsText(0)
AOI = arcpy.GetParameterAsText(1)
output = arcpy.GetParameterAsText(3)
method = arcpy.GetParameterAsText(4)

#Getting the name of the feature from the output parameter
name = output.rsplit('\\', 1)[1]

#Set Enviroment from the Parameter
arcpy.env.overwriteOutput = True
arcpy.AddMessage(method)
arcpy.AddMessage(output)
arcpy.AddMessage(name)

#Creating an intermedium Feature Class
arcpy.AddMessage("Intersecting...")

if method == 'true':
   #Selecting
   arcpy.AddMessage("true...")
   inter = arcpy.SelectLayerByLocation_management (clusters, "INTERSECT",
   AOI)
   #BlocksAOI =arcpy.CopyFeatures_management(inter, 'BlocksAOI')
   BlocksAOI =arcpy.CopyFeatures_management(inter, output)
   arcpy.SelectLayerByAttribute_management(clusters, "CLEAR_SELECTION")
else:
   #Clipping
   arcpy.AddMessage("Clipping...")
   BlocksAOI = arcpy.Intersect_analysis ([clusters, AOI], output)
    BlocksAOILayer= arcpy.MakeFeatureLayer_management (BlocksAOI,
    'BlocksAOILayer')
   ClusterLayer= arcpy.MakeFeatureLayer_management (clusters,  'ClusterLayer')
   #Creating a Join
```

```
    arcpy.AddMessage(clusters)
    #If the name of the layer is based in compose  ie : Geodatabase//San_bernardino

    if len(clusters.split("\\")) > 1:
        string = clusters.split("\\")[1]
    else:
        string = clusters
    arcpy.AddMessage(string)

     BlockUnited = arcpy.AddJoin_management(BlocksAOILayer, 'FID_'+ str(string),
     ClusterLayer, 'OBJECTID')
    mylist = [ name +'.' + "FID_" + str(string)]

    #Identifing both area fields
    fields = arcpy.ListFields(BlockUnited, '*.Shape_Area')
    for field in fields:
        mylist.append(field.name)

    fc = BlockUnited
    fields = mylist
    mylist2 = []

    #If there is a difference bewteen the 2 areas , then get the FID , adding to a list
    with arcpy.da.SearchCursor(fc, fields) as cursor:
        for row in cursor:
            if float(row[2]) - float(row[1]) > 1:
                mylist2.append(row[0])

    #Adding a field
    arcpy.AddMessage("Adding Fields...")
    arcpy.AddField_management(BlockUnited, "Clipped", "TEXT")

    # Update the Field Clipped using the FID in the list
    with arcpy.da.UpdateCursor(BlocksAOI, ['FID_'+ str(string),'Clipped']) as cursor:
        for row in cursor:
            row[1] = 'non clipped'
            for i in mylist2:
                if row[0] == i:
                    row[1] = 'clipped'
                cursor.updateRow(row)

arcpy.SetParameterAsText(3,BlocksAOI)
```

```
##-------------------------------------------------------------------------
```
CASPER Occupied HU - Cesar Garrido Lecca - University of Redlands

```
import arcpy
from random import randint
import numpy as np
import math

#Parameters
layer = arcpy.GetParameterAsText(0)

#Finding the Workspace
d = arcpy.Describe(layer)
work = d.path

#Set Enviroment from the Parameter
arcpy.env.workspace = work
arcpy.env.overwriteOutput = True

#Adding fields
arcpy.AddMessage("Adding Fields...")
arcpy.AddField_management(layer, "TotalOccupied", "LONG")
#arcpy.AddField_management(copy, "RANDOM", "LONG")
#Calculating the new Fields

arcpy.AddMessage("Update Fields...")
fields = ['TotalOccupied' , 'RENTER10','OWNER10']
# Create update cursor for feature class
suma = 0
with arcpy.da.UpdateCursor(layer, fields) as cursor:
   for row in cursor:
      row[0] = row[1] + row[2]
      # Update the cursor with the updated list
      cursor.updateRow(row)
```

```
#---------------------------------------------------------------------------
```
CASPER Random Sampling - Cesar Garrido Lecca - University of Redlands

```
import arcpy
from random import randint
import numpy as np
import math
from collections import Counter

#Parameters
layer = arcpy.GetParameterAsText(0)
unit = arcpy.GetParameterAsText(1)
Ncluster = arcpy.GetParameterAsText(3)
Ncluster = int(Ncluster)
HouseUnits = arcpy.GetParameterAsText(4)
output = arcpy.GetParameterAsText(5)
name = output.rsplit('\\', 1)[1]
arcpy.env.overwriteOutput = True

#Creating an intermedium Feature Class
arcpy.AddMessage("Copy Feature...")
filterblocks = arcpy.MakeFeatureLayer_management(layer, "filterblocks" , str(unit)
+' <> 0')
copy = arcpy.management.CopyFeatures(filterblocks,
"in_memory\\CopyEnrichedRamdom10")

#Adding fields
arcpy.AddMessage("Adding Fields...")
arcpy.AddField_management(copy, "SUM", "LONG")
arcpy.AddField_management(copy, "CLUSTER_ID", "LONG")
arcpy.AddField_management(copy, "NTimes", "LONG")
arcpy.AddField_management(copy, "NSurveys", "LONG")

#---------------------------------------------------------
#Calculating the new Fields
arcpy.AddMessage("Update Fields...")
fields = [unit , 'SUM']
# Create update cursor for feature class
suma = 0
with arcpy.da.UpdateCursor(copy, fields) as cursor:
   for row in cursor:
      suma = row[0] + suma
      row[1] = suma
      # Update the cursor with the updated list
      cursor.updateRow(row)
```

```
##--------------------------------------------------------------------------

arcpy.AddMessage("Processing...")

#GETTING THE MAX VALUES FROM THE SUM COLUMN
maxValue = arcpy.SearchCursor(copy, "", "", "", 'SUM' + "
D").next().getValue('SUM') #Get 1st row in descending cursor sort

#CREATING A LIST WITH FROM THE SUM COLUMN
listasuma = []
cursor = arcpy.SearchCursor(copy, "", "", "", 'SUM')
for row in cursor:
   listasuma.append(row.getValue('SUM'))
#arcpy.AddMessage(listasuma)

#GETTING THE CLOSEST HIGHER NUMBER TO EACH RAMDOM VALUE
newvalues = []
ramdoml= []
indexe = []

while len(newvalues) < Ncluster:
   random = randint(0,maxValue)
   for i in listasuma:
      #arcpy.AddMessage(i)
      if i == random:
         newvalues.append(i)
         ramdoml.append(random)
         #Incressing the ramdom to do not match other numbers.
         random = random + math.inf
         #Adding the index of the number
         indexe.append(listasuma.index(i))
      elif i > random:
         # If this value is in already in the list (we skip rerun the proces, because the
         block is aready selected.
         newvalues.append(i)
         ramdoml.append(random)
         #Incressing the ramdom to do not match other numbers.
         random = random + math.inf
         #Adding the index of the number
         indexe.append(listasuma.index(i))

#Printing a message with values RAMDOM
arcpy.AddMessage("Random Values")
arcpy.AddMessage(sorted(ramdoml))
arcpy.AddMessage(len(ramdoml))
```

```
#Printing a message with values
arcpy.AddMessage("Captured Values")
arcpy.AddMessage(sorted(newvalues))
arcpy.AddMessage(len(newvalues))

##--------------------------------------------------------------------------

#Converting new values list into a dictionaty
a = Counter(newvalues)
arcpy.AddMessage(a)

#UPDATING RESULTS
suma = 1
fields = ['SUM', 'CLUSTER_ID','NTimes']
with arcpy.da.UpdateCursor(copy, fields) as cursor:
   for row in cursor:
      # For each value in the dictionaty
      for x in list(a.keys()):
         #If the value in the table is the same than in the dictionaty
         if row[0] == x:
            #arcpy.AddMessage(x)
            row[1] = suma
            row[2] = a.get(x)
            #arcpy.AddMessage(suma)
            suma = suma + 1

      # Update the cursor with the updated list
      cursor.updateRow(row)

#CREATING CLUSTER LAYER
arcpy.AddMessage("Creating Clusters...")
arcpy.CalculateField_management(copy, "NSurveys",'!NTimes! *' + HouseUnits,
"PYTHON3")
cluster = arcpy.MakeFeatureLayer_management(copy, "cluster", "CLUSTER_ID > 0
and CLUSTER_ID <= " + str(Ncluster))
final = arcpy.management.CopyFeatures(cluster, output)
arcpy.SetParameterAsText(2,final)
```

```
#---------------------------------------------------------------------------------------
```
CASPER Prepare and Export data - Cesar Garrido Lecca - University of Redlands

```
import arcpy
import collections
import numpy as np

#Parameters
clusters = arcpy.GetParameterAsText(0)
survey = arcpy.GetParameterAsText(1)
Totpop = arcpy.GetParameterAsText(2)
excel = arcpy.GetParameterAsText(3)
PreAnalysis = arcpy.GetParameterAsText(5)

#Finding the Workspace
d = arcpy.Describe(Totpop)
work = d.path

new = work.split(".gdb")[0]
new2 = new +'.gdb'

#Set Enviroment from the Parameter
arcpy.env.workspace = work
arcpy.env.overwriteOutput = True

#Creating an intermedium Feature Class
arcpy.AddMessage("Processing...")

arcpy.AddMessage("Intersect...")
SurveyAnalysis = arcpy.Intersect_analysis ([clusters, survey], PreAnalysis)

arcpy.AddMessage("Adding Fields...")
arcpy.AddField_management(SurveyAnalysis, "Weight", "DOUBLE")
arcpy.AddField_management(SurveyAnalysis, "NumInterview", "SHORT")

arcpy.DeleteField_management(SurveyAnalysis,
                        ["STATEFP10", "COUNTYFP10", "TRACTCE10","GEOID10",
"NAME10", "MTFCC10","UR10", "UACE10", "UATYPE","FUNCSTAT10",
"ALAND10", "AWATER10","INTPTLAT10", "INTPTLON10",
"GEOIDNum","FInal__GEO_id", "FInal__GEO_id2",
"FInal__GEO_display_label","FInal__Population","FInal__House_Units",
"FInal__Occupied", "FInal__Vacant","HasData", "TOTHU10",
"HHPOP10","EnrichOccupied", "PopulationOccupied2010",
"SUM_","RANDOM","Shape__Area", "Shape__Length"])
```

```
    # Calculating the total housing units
    #-------------------------------------------
    TotalHousingUnits = arcpy.analysis.Statistics( Totpop ,
r"in_memory\\TotalHousingUnits", "TOTHU10 SUM", None)
    fc = TotalHousingUnits
    fields = ['SUM_TOTHU10']
    mylist3 = []

    with arcpy.da.SearchCursor(fc, fields) as cursor:
        for row in cursor:
            mylist3.append((row[0]))
    arcpy.AddMessage(mylist3)

    # Calculating the number of surveys per Block
    #-------------------------------------------
    Stat = arcpy.analysis.Statistics(SurveyAnalysis, r"in_memory\\statSurvey",
"BLOCKID10 COUNT", "BLOCKID10")
    mylist = ['BLOCKID10','FREQUENCY']
    fc = Stat
    fields = mylist
    mylist2 = []

    # Cursor in the summary statistics table
    with arcpy.da.SearchCursor(fc, fields) as cursor:
        for row in cursor:
            mylist2.append((row[0],row[1]))

    #Updating the Fields
    #-----------------------
    with arcpy.da.UpdateCursor(SurveyAnalysis, ['BLOCKID10','NumInterview']) as
    cursor:
        for row in cursor:
            for i in mylist2:
                if row[0] == i[0]:
                    row[1] = i[1]
            cursor.updateRow(row)

    ########################################################################

    arcpy.AddMessage("Adding Fields...")
    arcpy.AddField_management(SurveyAnalysis, "BLOCKID_Mod", "TEXT")
    arcpy.AddField_management(SurveyAnalysis, "NumInter_Mod", "short")
    arcpy.CalculateField_management (SurveyAnalysis, 'BLOCKID_Mod',
    "!BLOCKID10!","PYTHON3")
    arcpy.CalculateField_management (SurveyAnalysis, 'NumInter_Mod',
    "!NumInterview!","PYTHON3")
```

```
svet= []
svetsort = []
with arcpy.da.SearchCursor(SurveyAnalysis, ['BLOCKID_Mod','NumInter_Mod',
'OBJECTID']) as cursor:
    for row in cursor:
        if row[1] > 7:
            svet.append([row[0],row[1],row[2]])

lista = sorted(svet)
#arcpy.AddMessage(lista)

# Use Dictionaties
# General counting
idDic={};
groupby=7
for i in range(len(lista)):
    _id=lista[i][0]
    if(_id not in idDic):
        idDic[_id]=[]
    idDic[_id].append(i)

# if block has more than 7 surveys then split the block in
# two virtual blocks and change the coding
for _id in idDic:
    _array=idDic[_id]
    _total=len(_array)
    for i in range(_total):
        _index=_array[i]
        if _total<groupby:
            lista[_index][1]=_total
        elif _total>groupby:
            group_num=int(i/groupby)
            if( group_num > 0):
                exceed=_total-(groupby*group_num)
                if(exceed>groupby):
                    exceed=groupby
                lista[_index][0]+="_"+str(group_num+1)
                lista[_index][1]=exceed
            else:
                lista[_index][1]=groupby

#update the new values for the block ids
with arcpy.da.UpdateCursor(SurveyAnalysis, ['BLOCKID_Mod','NumInter_Mod',
'OBJECTID']) as cursor:
    for row in cursor:
```

```
        for i in lista:
            if row[2] == i[2]:
                row[0] = i[0]
                row[1] = i[1]
        cursor.updateRow(row)

arcpy.AddMessage('Finish Spliting')

#########################################################

#Export Excel
with arcpy.da.UpdateCursor(SurveyAnalysis, ['NumInter_Mod' , 'Weight']) as
cursor:
    for row in cursor:
        row[1] = (int(mylist3[0])/(int(row[0]) * 30))
        cursor.updateRow(row)

arcpy.SetParameterAsText(4,SurveyAnalysis)
arcpy.conversion.TableToExcel(SurveyAnalysis, excel + '.xls', "ALIAS",
"DESCRIPTION")


#---------------------------------------------------------------------------
```

CASPER Quick report - Cesar Garrido Lecca - University of Redlands

```
import arcpy
import collections
import numpy as np

#Parameters
PreAnalysis = arcpy.GetParameterAsText(0)
Variable = arcpy.GetParameterAsText(1)
arcpy.AddMessage(Variable)

#Finding the Workspace
d = arcpy.Describe(PreAnalysis)
work = d.path
new = work.split(".gdb")[0]
new2 = new +'.gdb'

#Set Enviroment from the Parameter
arcpy.env.workspace = work
arcpy.env.overwriteOutput = True

#Creating the Report
```

```
    notnull = arcpy.MakeFeatureLayer_management (PreAnalysis, 'in_memory\\notnull',
Variable + ' IS NOT NULL')
    table = arcpy.analysis.Statistics(notnull, new2 + "\\QuickReport", 'Weight' + "
SUM", Variable)

    #Adding fields
    arcpy.AddMessage("Adding Fields...")
    arcpy.AddField_management(table, "Percent", "DOUBLE")
    arcpy.AddField_management(table, "Cum_Percent", "DOUBLE")
    a = 0
    with arcpy.da.SearchCursor(table, ['SUM_Weight']) as cursor:
       for row in cursor:
          a = a + row[0]

    #Total sum of Weight
    arcpy.AddMessage(a)
    comulative = []
    with arcpy.da.UpdateCursor(table, ['SUM_Weight', 'Percent']) as cursor:
       for row in cursor:
          row[1] = (row[0] * 100) / a
          comulative.append(row[1])
          cursor.updateRow(row)

    #Comulative percentage Field
    comulative2 = np.cumsum(comulative)
    comulative3 = map(lambda x,y:(x,y),comulative,comulative2)

    for i in comulative3:
       arcpy.AddMessage(i[0])
       with arcpy.da.UpdateCursor(table, ['Percent','Cum_Percent']) as cursor:
          for row in cursor:
             if round(row[0],4) == round(i[0],4):
                row[1] = i[1]
             cursor.updateRow(row)

    arcpy.AlterField_management(table, 'SUM_Weight', 'weighted_frequency',
'Weighted Frequency')
    arcpy.AlterField_management(table, 'FREQUENCY', 'frequency', 'Frequency')
    arcpy.AlterField_management(table, 'Cum_Percent', 'cumulative_percentage',
'Cumulative Percentage')
    arcpy.AlterField_management(table, 'Percent', 'Percentage', 'Percentage')

    arcpy.SetParameterAsText(2,table)
```

Appendix B. Survey Fields

type	name	label
begin group	Identification	① Identification and physical location
date	Date	A1. Date
text	TeamID	A2.Team
text	InterName	A3. Interviewer Name
text	BlockId	A4. Block ID
text	HU	A5. House Unit ID
integer	NumHousehold	A6. Number of households in cluster
select_one Typeofstructure	TypeStruc	A7. Type of structure
text	TypeStrucOther	A7. Other Type of structure
text	TypeStrucCollated	A7. Type of structure collated
geopoint	geo	A8. Location
end group	Identificationfin	Final
begin group	Demographics	② Demographics and functional needs
select_one yes_no	PrimRes	D1. Is this your primary residence?
integer	ResBefore	D2. How many people lived in your household before the disaster?
integer	ResAfter	D3. How many people slept in your home last night?
select_one StayCh	Stay	D3a. If NONE, where did you and members of your household sleep?
text	Stay_txt	D3b. Other
text	Stay_Collated	D3c. If NONE, where did you and members of your household sleep? (Collated)
integer	NumSlept	D4. How many people from other households slept in your home last night?
begin group	D5	D5. Currently, how many people living in your household are:
integer	P_LT2	D5a. How many people living in your household are less than 2 years old?
integer	P_2to17	D5b. How many people living in your household are 2–17 years old?

integer	P_16to64	D5c. How many people living in your household are 18–64 years old?
integer	P_65	D5d. How many people living in your household aremore than 64 years old?
note	Notification	 The total sum of people living in your house does not coincide with D3, please check the data entered
end group	D5fin	
begin group	D6	D6. How many people living in your household are:
integer	D6_Male	D6a. How many people living in your household are male?
integer	D6_Female	D6b. How many people living in your household are female?
end group	D6fin	Final
select_one yes_no	Preg	D7. Is anyone in your household pregnant?
integer	PregNum	D7a.If Yes, how many?
select_multiple Race	Race	D8. What is your race? (check all that apply)
select_one yes_no	Hispanic	D9. Are you Hispanic or Latino?
select_one yes_no	NonEnglish	D10. Is there an adult in your household who does not speak English?
select_one Marital	Marital	D11. What is your current marital status
select_one yes_no	Deaf	D12. Do you or does a member of your household have serious difficulty hearing?
select_one yes_no	Blind	D13. Are you or a member of your household blind or have serious difficulty seeing, even when wearing glasses?
select_one yes_no	Concentrate	D14. Because of a physical, mental, or emotional condition, do you or does a member of your household have serious difficulty concentrating, remembering, or making decisions? (5 years old or older)
select_one yes_no	Climbing	D15. Do you or does a member of your household have difficulty walking or climbing stairs? (5 years or older)
select_one yes_no	DifficultBathing	D16. Do you or does a member of your household have difficulty dressing or bathing? (5 years old or older)
select_one yes_no	DifficultErrands	D17. Because of a physical, mental, or emotional condition, do you or does a member of your household have difficulty doing errands alone such as visiting a doctor’s office or shopping? (15 years old or older)

select_one Temporal	Worried_Rent	D18. How often in the past 12 months would your household say they were worried or stressed about having enough money to pay your rent/mortgage? Would you say you were worried or stressed –
select_one Temporal	Worried_Meal	D19. How often in the past 12 months would your household say they were worried or stressed about having enough to buy nutritious meals? Would you say you were worried or stressed –
select_one Own	Own	D20. Do you own or rent this residence?
text	Own_txt	D20.Other
text	OwnCollated	D20. Do you own or rent this residence? (Collated)
select_one yes_no	Evacuate	D21. Did your household evacuate your home any time during or before the [disaster]?
select_one EvacNo	EvacNo	D21b. If NO, what prevented your household from evacuating?
text	EvacNo_txt	D21b. Other
text	EvacNoCollated	D21b. If NO, what prevented your household from evacuating? (Collated)
select_one EvacLoc	EvacLoc	D21c. If YES, where did your household evacuate to?
text	EvacLoc_txt	D21c. Other
text	EvacLocCollated	D21c. If YES, where did your household evacuate to? (Collated)
date	EvacDate	D21d. If YES, when date did your household evacuate?
date	EvacRet	D21e. If YES, on what date did your household return home to sleep?
select_one yes_no	TempRes	D22. Are you and members of your household currently in temporary housing?
end group	Demographicsfin	Final
begin group	DamageRepair	③ Damage and repair
select_one Damage	Damage	DR1. How would you describe the damage to your home?
select_one yes_no	Tarp	DR1b. If DAMAGED, do you need a tarp?
select_one yes_no	Safe	DR2. Do you feel your home is currently physically safe to live in?
select_multiple NoSafe	NoSafe	DR2b. If NO, why not? (check all that apply):
text	NoSafe_txt	DR2b. Other
text	NoSafeCollated	DR2b. If NO, why not? (check all that apply): (Collated)
select_one WhnSafe	_WhnSafe	DR2c. If NO, when do you expect this home to be physically safe to live in?

integer	FloodHgt	DR3. How high did the flood waters reach inside your home?
select_one Condition	Condition	DR4. What is the condition of your home now?
text	Condition_txt	Other
text	ConditionCollated	DR4. What is the condition of your home now? (Collated)
select_one Live1Mo	Live1Mo	DR6. Where does your household expect to be living in one month?
text	Live1Mo_txt	DR6. Other
text	Live1MoCollated	DR6. Where does your household expect to be living in one month? (Collated)
select_one yes_no	Mold	DR7. Do you see mold or smell a moldy/musty odor in your home?
select_one yes_no	MoldDam	DR8. Is there 10 sq ft or more of water damage and/or mold anywhere inside your home?‡
select_one CropLoss	CropLoss	DR9. Since the [disaster], have you had any significant loss to your crops?
select_one yes_no	InsClean	DR10. Does your household have insurance to cover cleanup?
select_one yes_no	InsRepair	DR11. Does your household have insurance to cover repair?
select_one yes_no	FinClean	DR12. Does your household need financial assistance for cleanup of your home?†
select_one yes_no	FinRepair	DR13. Does your household need financial assistance to repair your home?†
select_one yes_no	MeanCln	DR14. Does your household have the means (e.g., financial, physical) to cleanup your home?
select_one yes_no	MeanRep	DR15. Does your household have the means (e.g., financial, physical) to repair your home?
end group	DamageRepairFin	Final
begin group	GeneralUtilities	④ General utilities
begin group	U1	U1. Do you currently have the following services in your home?† (check all that apply):
select_one yes_no_2	WaterServ	Running water
select_one yes_no_2	Electric	Electricity
select_one yes_no_2	Garbage	Garbage pickup
select_one yes_no_2	Gas	Natural gas
select_one yes_no_2	Sewage	Sewage service
select_one yes_no_2	Phone	Telephone (landline)
select_one yes_no_2	Cell	Cell phone

end group	U1Fin	
select_one Electric	ElectricSource	U2. What is your current source of electricity?
select_one Heat	Heat	U3. What is your current source of heat?
text	Heat_txt	U3. Other
text	HeatCollated	U3. What is your current source of heat? (Collated)
select_one yes_no_2	Toilet	U4. Do you have a working indoor toilet?
select_one yes_no_2	_access	U4b. If NO, do you have access to a working toilet?
end group	GeneralUtilitiesFin	
begin group	CarbonMonoxide	⑤ Carbon monoxide exposure
select_one yes_no	GenUse	E1. Since the [disaster], have you used a generator?
select_one GenLoc	GenLoc	E1b. If YES, where is/was the generator located?
select_one yes_no	GenWin	E1c. If OUTSIDE, is/was the generator near an open or broken window?
select_one Grill	Grill	E2. Since the [disaster], have you used a charcoal or gas grill/camp stove?
select_one yes_no	GrillWin	E2b. If OUTSIDE, is/was the grill/camp stove near an open or broken window?
select_one yes_no	PresWash	E3. Since the [disaster], have you used a pressure washer with the actual engine in the house/garage?
select_one yes_no	CODetect	E4. Do you have a carbon monoxide detector?
select_one yes_no	COWork	E4b. If YES, is it working?
end group	CarbonMonoxideFin	
begin group	AnimalSafety	⑥ Animal safety
select_one yes_no	Mosquito	A1. Since the [disaster], have you noticed an increase in mosquitoes around your home or neighborhood?
select_one yes_no	Protect	A2. Are you or your household members doing anything to protect yourselves from mosquitoes?
select_multiple TypeProt	TypeProt	A2b. If YES, what type of protective measures are you using? (check all that apply):
text	TypeProt_txt	A2b. Other
text	TypeProtCollated	A2b. If YES, what type of protective measures are you using? (check all that apply): (Collated)
select_one yes_no	Snakes	A3. Since the [disaster], have you or anyone in your household noticed an increase in snakes around your home or neighborhood?
select_one yes_no	Alligators	A4. Since the [disaster], have you or anyone in your household noticed an increase in alligators around your home or neighborhood?

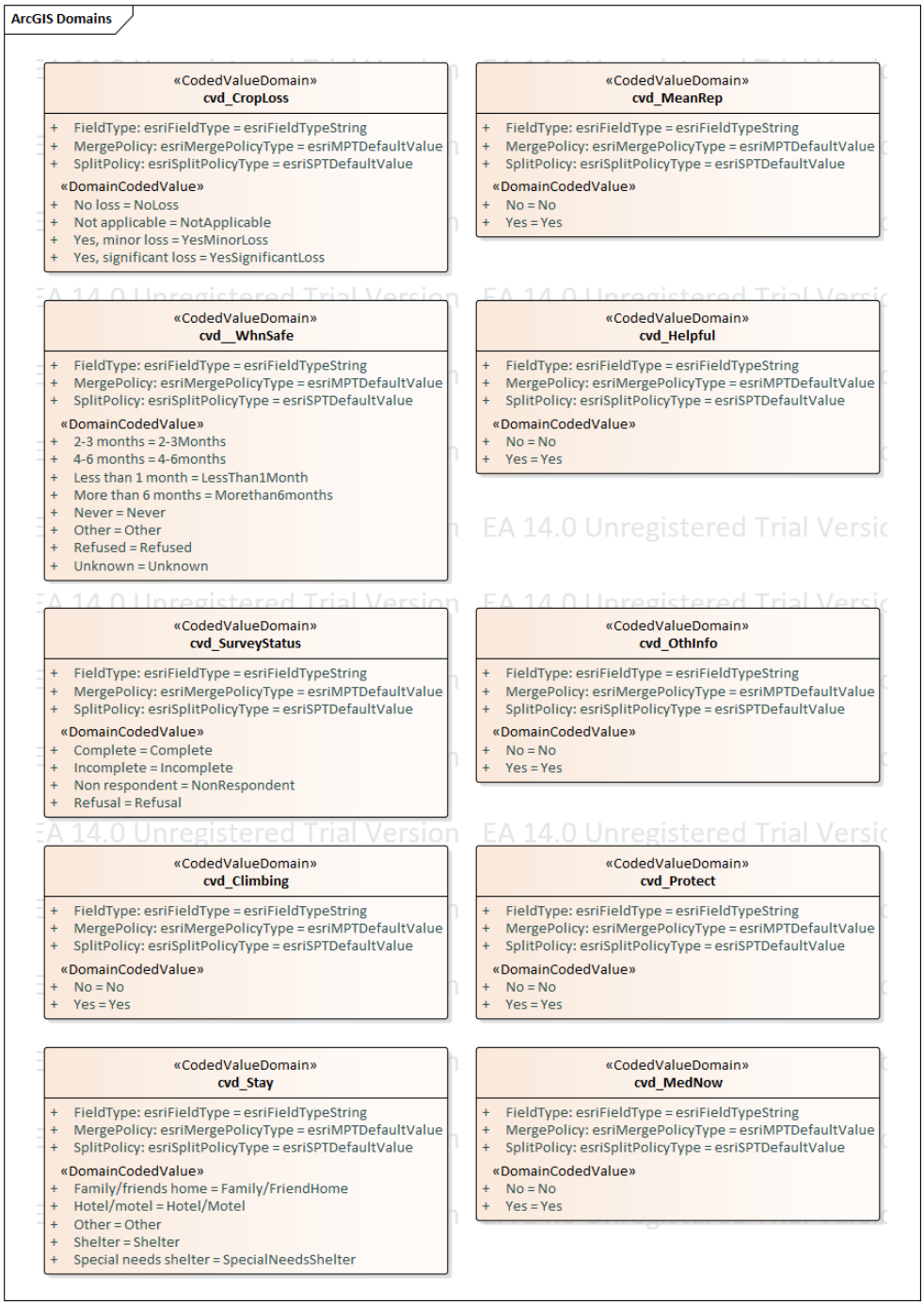
select_one yes_no	Bitten	A5. Since the [disaster], have you or anyone in your household been bitten by an animal or insect other than mosquitoes?
text	_Animal	A5b. If YES, what animal?
end group	AnimalSafetyFin	
begin group	SuppliesRelief	⑦ Supplies, relief, and emergency preparedness
select_one yes_no	Water	S1. Does your household have adequate drinking water (besides tap) for the next 3 days? (1 gallon/person/day)
select_one WBefore	WBefore	S2. What was your households primary source of drinking water before the [disaster]?
select_one WNow	WNow	S2b. What is your households primary source of drinking water right now?
select_one WTreat	WTreat	S2c. If using WELL or MUNICPAL water, is your household treating the water?
select_one yes_no	WatAid	S3. Has your household received bottled water as part of the disaster relief effort?
select_one yes_no	Food	S4. Do you have access to adequate food for everyone in your household for the next 3 days?
select_one yes_no	SupplyKit	S5. Has your household prepared an Emergency Supply Kit with supplies like water, food, flashlights, and extra batteries that is kept in a designated place in your home?
select_one yes_no	FirstAidKit	S6. Has your HH prepared a first aid kit with emergency supplies to take with you if you had to leave quickly?
select_one yes_no_Med	SupplyMeds	S7. Does your household currently have a 7-day supply of medication for each person who takes prescribed meds?
begin group	S8	S8. Does your household have any of the following emergency plans?
select_one yes_no	_commPlan	S8a.Emergency communication plan such as a list of numbers and designated out-of-town contact
select_one yes_no_NA	_meetHome	S8b.Designated meeting place immediately outside your home or close by in your neighborhood
select_one yes_no_NA	_meetOut	S8c.Designated meeting place outside of your neighborhood in case you cannot return home
select_one yes_no	_documents	S8d.Copies of important documents in a safe location (e.g., water proof container)
select_one yes_no	_routes	S8e.Multiple routes away from your home in case evacuation is necessary
end group	U8Fin	

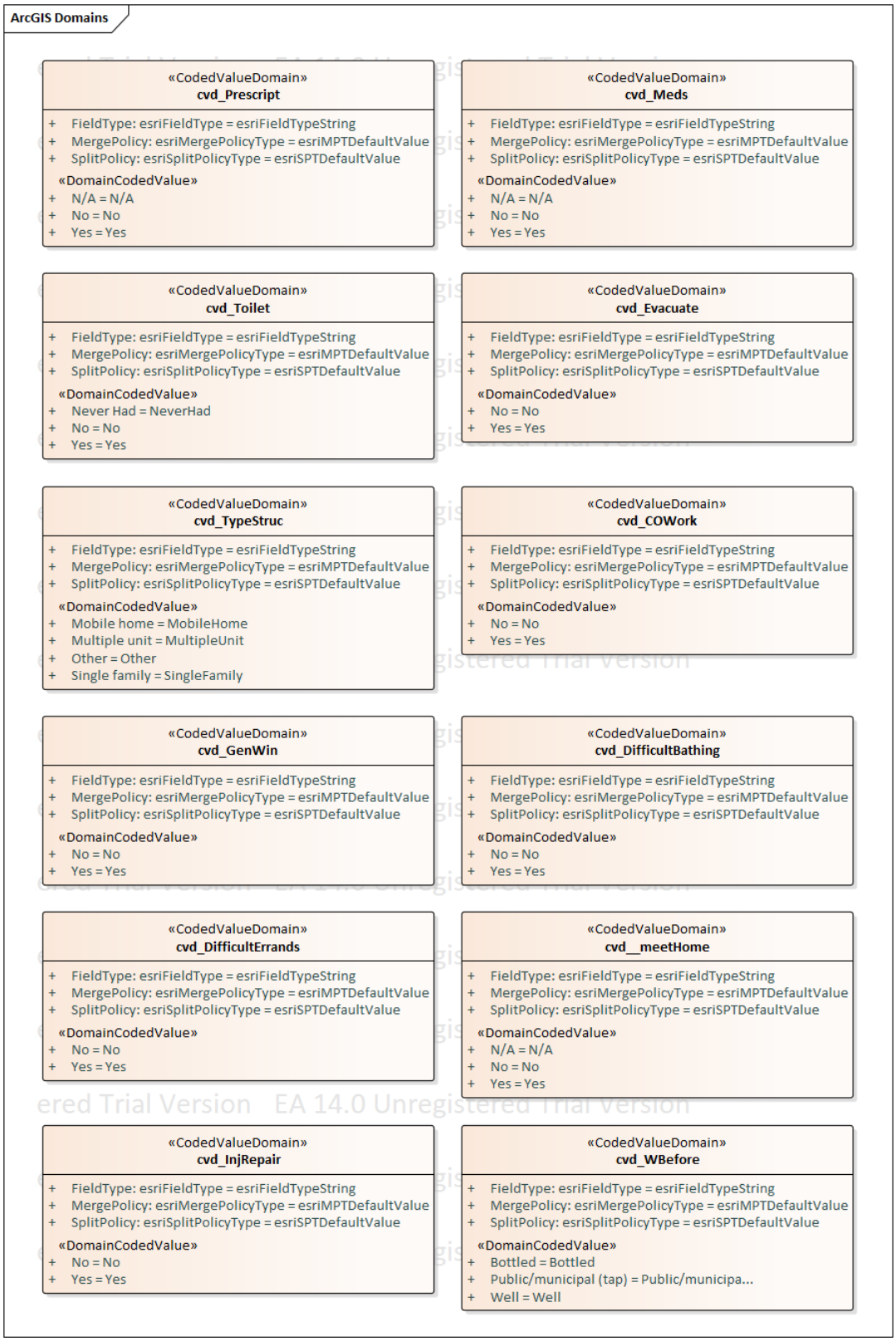
select_one yes_no	FoodAid	S9. Has your household received food as part of the disaster relief effort?
select_one yes_no	Aid	S10. Have you or a member of your household received any type of aid as part of the relief effort?
select_multiple NoAid	NoAid	S10b. If NO, why not?
text	NoAid_txt	S10b. Other
text	NoAidCollated	S10b. If NO, why not? (Collated)
select_multiple YesAid	YesAid	S10c. If YES, what type of aid?
select_one yes_no_2	Transport	S11. Does your household currently have access to transportation if needed?
select_one yes_no_NA	Fuel	S12. Does your household have access to fuel?†
select_one yes_no	Clothes	S13. Do you, or does anyone in your household, need clothes?
text	GreatNeed	S14. What is your greatest need at this time?‡
end group	SuppliesReliefFin	
begin group	HealthStatus	⑧ Health status and disabilities
select_one yes_no	Injury	H1. Since the [disaster], have you or a member of your household been injured?†
select_multiple injury	InjuryOpt	H1b. If YES, what was/were the injury(s)?‡
text	InjuryOpt_txt	H1b.Other
text	InjuryOptCollated	H1b. If YES, what was/were the injury(s)?‡ (Collated)
select_multiple InjuryOpt2	InjuryOpt2	H1c. If YES, what part of the body was injured?
select_one yes_no	InjRepair	H1d. If YES, were you or the member of your household injured while repairing the residence or retrieving items?
text	InjCause	H1e. If YES, what caused the injury?‡
select_one yes_no_NA	Tetanus	H2. Has everybody in your household had a tetanus (DTaP/Tdap/Td) shot in the past 10 years?
select_one yes_no	Died	H3. Has any member of your household died as a result of the [disaster]?
integer	DiedNum	H3b. If YES, how many?
select_multiple Died	DiedHow	H3c. If YES, how did they die?
text	DiedHow_txt	H3c.Other
text	DiedHowCollated	H3c. If YES, how did they die? (Collated)
select_multiple Health	Health	H4. Since the [disaster], has anybody in your household experienced‡
text	Chr_txt	H4. What type(s) of chronic illness(es)
text	Health_txt	H4. Other
text	HealthCollated	H4. Since the [disaster], has anybody in your household experienced‡ (Collated)

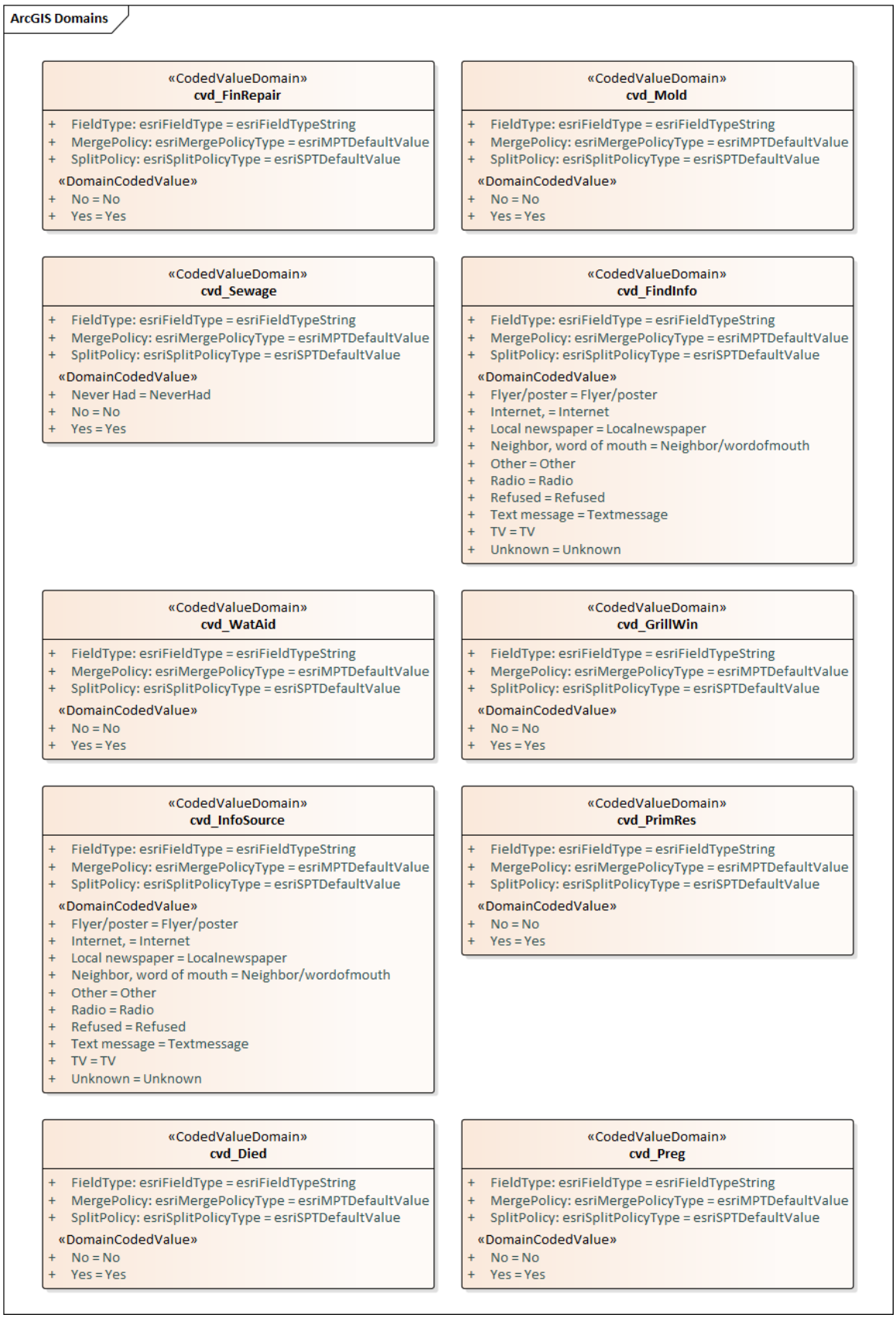
select_one yes_no	Illness	H5. Has anybody in your household become ill since the [disaster]?
select_multiple PrevHealth	PrevHealth	H6. Have you or a member of your household ever been told by a healthcare professional that he/she has‡
text	PrevHealth_txt	H6. Other
text	PrevHealthCollated	H6. Have you or a member of your household ever been told by a healthcare professional that he/she has‡ (Collated)
select_multiple MH	MentalHealth	H7. Since the [disaster], has anybody in your household experienced an increase in‡
text	MentalHealth_txt	H7. Other
text	MentalHealthCollated	H7. Since the [disaster], has anybody in your household experienced an increase in‡ (Collated)
select_one yes_no	DailyTask	H8. Since the [disaster], have you or any member of your household been unable to perform your/their daily tasks?
select_one yes_no	MH	H9. Since the [disaster], are emotional concerns preventing you or any member of your household from taking care of yourself/themselves or others?
end group	HealthStatusFin	
begin group	MedicalCare	⑨ Medical care and prescriptions
select_one yes_no	MedCare	M1. Since the [disaster], have you or has anybody in your household required medical care?
select_one yes_no	CareNeed	M2. Since the [disaster], are you able to get the care you need for everyone in your household?
select_one NoCare	NoCare	M2b. If NO, why not?
text	NoCare_txt	M2b. Other
text	NoCareCollated	M2b. If NO, why not? (Collated)
select_one yes_no	MedNow	M3. Is there anyone in your household who currently requires urgent medical care?
select_one yes_no_NA	Prescript	M4. Since the [disaster], is everybody in your household getting the prescription medications they need?
select_multiple NoPrescript	NoPrescript	M4b. If NO, why not?
text	NoPrescript_txt	M4b. Other
text	NoPrescriptCollated	M4b. If NO, why not? (Collated)
select_one yes_no_NA	Meds	M5. Do the people in your household who need prescribed medicine have enough for the next 3–7 days?
select_multiple Med	Med	M6. Do you or does any member of your household need

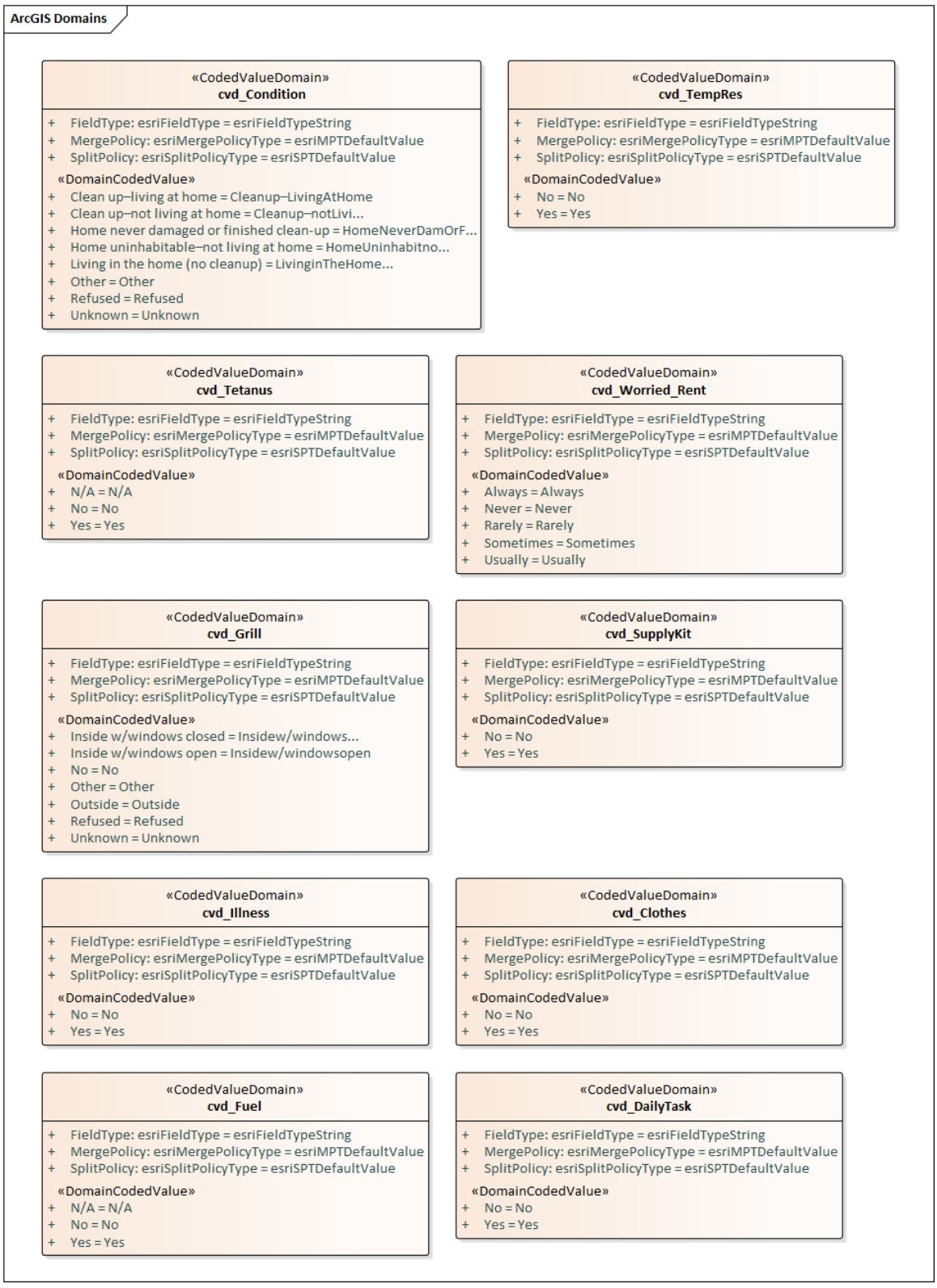
text	Med_txt	M6.Other
text	MedCollated	M6. Do you or does any member of your household need (Collated)
end group	MedicalCareFin	
begin group	Communication	⑩ Communication
select_one yes_no	Boil	C1. Since the [disaster], have you or members of your household received any information about boil water advisories in your area?
select_one InfoSource	InfoSource	C1b. If YES, what was the PRIMARY source of information?†
text	Web_txt	WebPage
text	InfoSource_txt	C1b. Other
text	InfoSourceCollated	C1b. If YES, what was the PRIMARY source of information?† (Collated)
select_one yes_no	Floods	C2. Have you or members of your household received warnings about Floods [specific concern, i.e., carbon monoxide]?‡
select_one yes_no	SearchInfo	C3. Have you or members of your household looked for information or answers to questions regarding the [disaster] and its effects?‡
select_one InfoSource	FindInfo	C4. Where would you first look for reliable information regarding the [disaster] and cleaning up after the [disaster]?
text	FWeb_txt	C4. WebPage
text	FindInfo_txt	C4. Other
text	FindInfoCollated	C4. Where would you first look for reliable information regarding the [disaster] and cleaning up after the [disaster]? (Collated)
select_one yes_no	OthInfo	C5. Have you or members of your household received any other health advice, cleanup tips, or other information related to the [disaster]?‡
select_one yes_no	Helpful	C5b. If YES, was the information received helpful?
select_one yes_no	Campaign	C6. Are you aware of the following materials [show materials, mention website/campaign, etc] to better prepare you and your family for a natural disaster or other significant event?
select_one yes_no	Chelpful	C6b. If YES, was the information received helpful
end group	CommunicationFin	
select_one SurveyStatus	SurveyStatus	Survey Status
image	Picture	Optional Picture

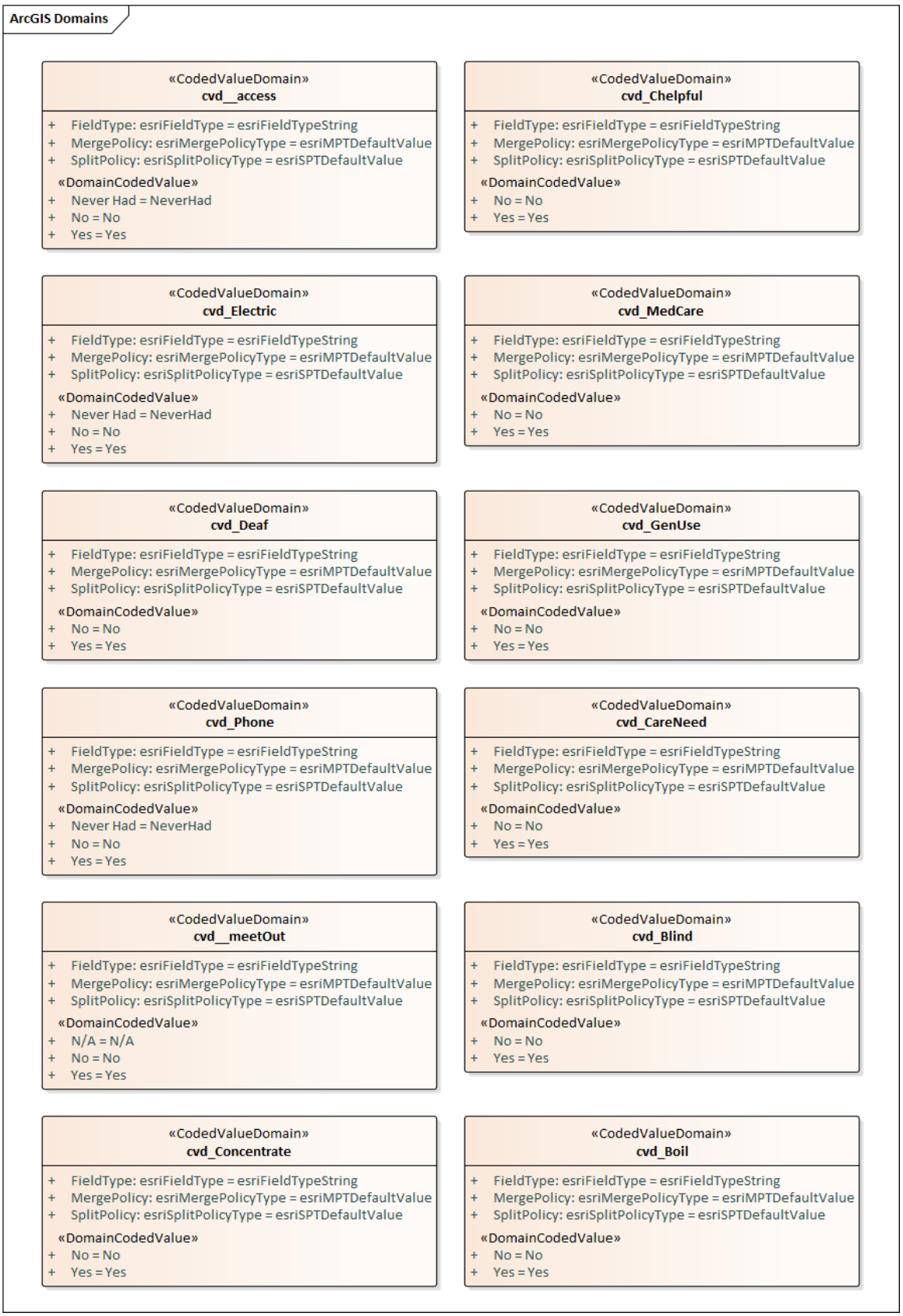
Appendix C. Domains

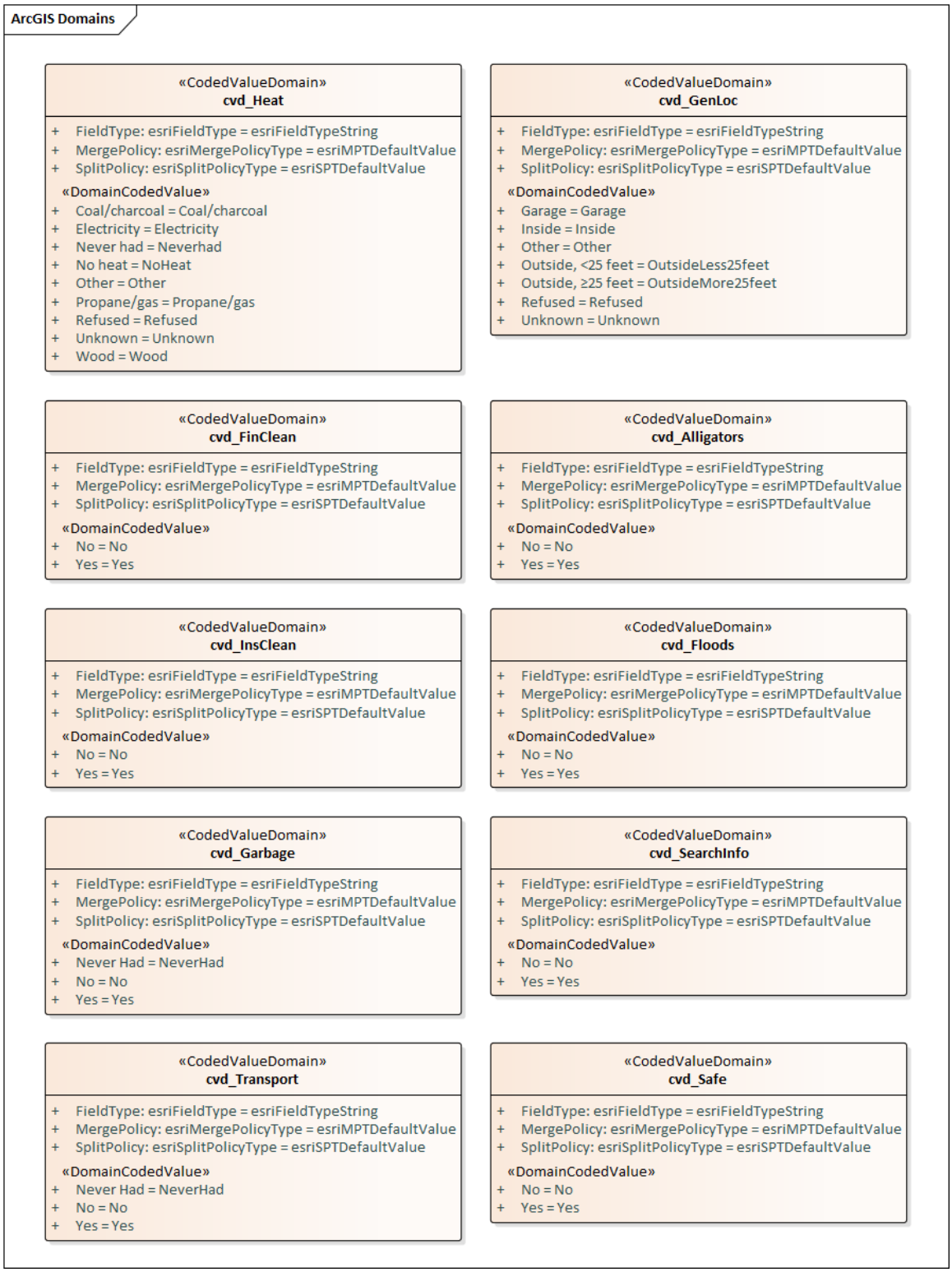


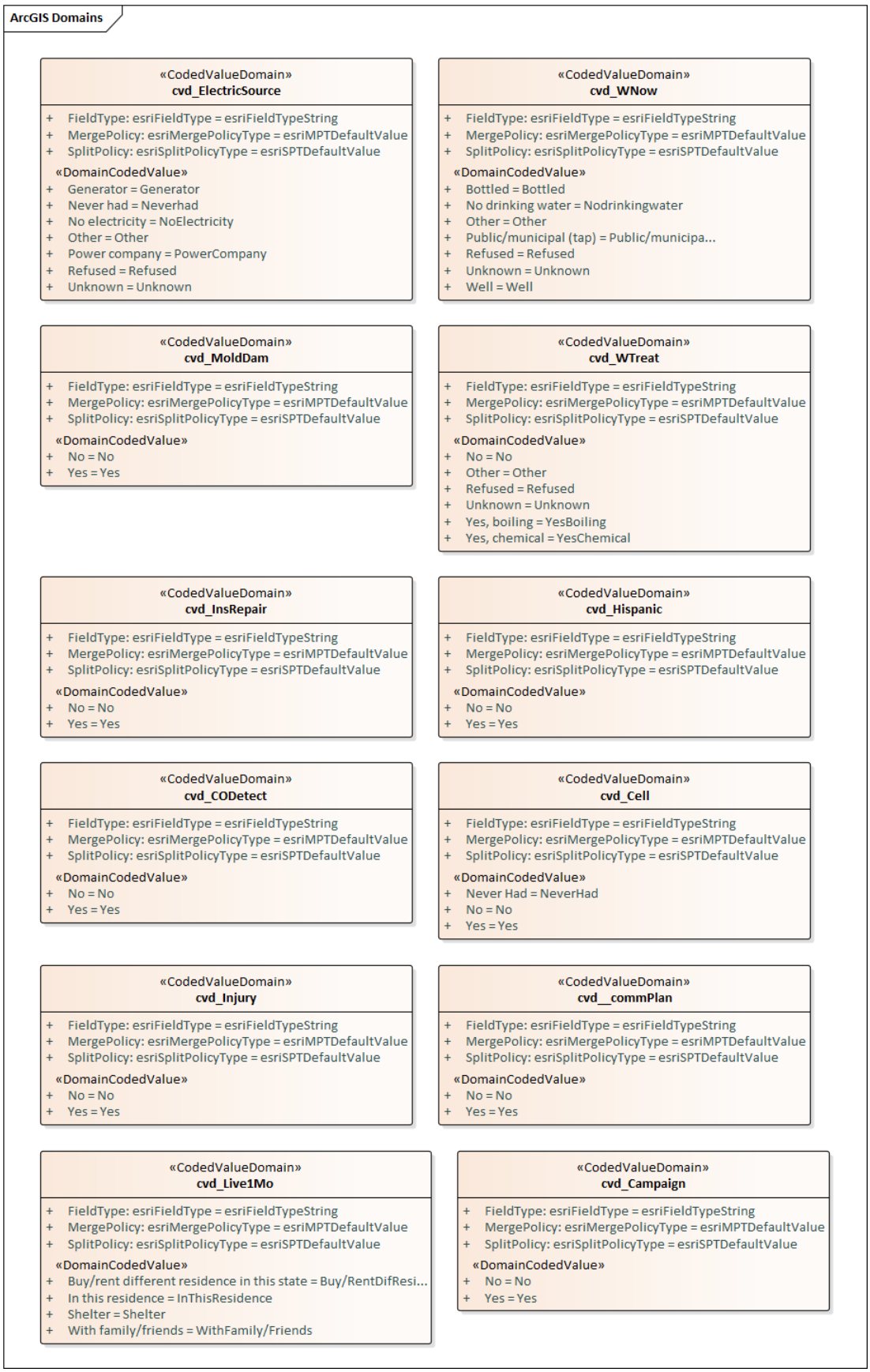

ArcGIS Domains
«CodedValueDomain»
cvd_Prescript
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Meds
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Toilet
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Evacuate
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_TypeStruc
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Mobile home = MobileHome
+ Multiple unit = MultipleUnit
+ Other = Other
+ Single family = SingleFamily
«CodedValueDomain»
cvd_COWork
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_GenWin
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_DifficultBathing
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_DifficultErrands
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd__meetHome
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_InjRepair
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_WBefore
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Bottled = Bottled
+ Public/municipal (tap) = Public/municipa...
+ Well = Well

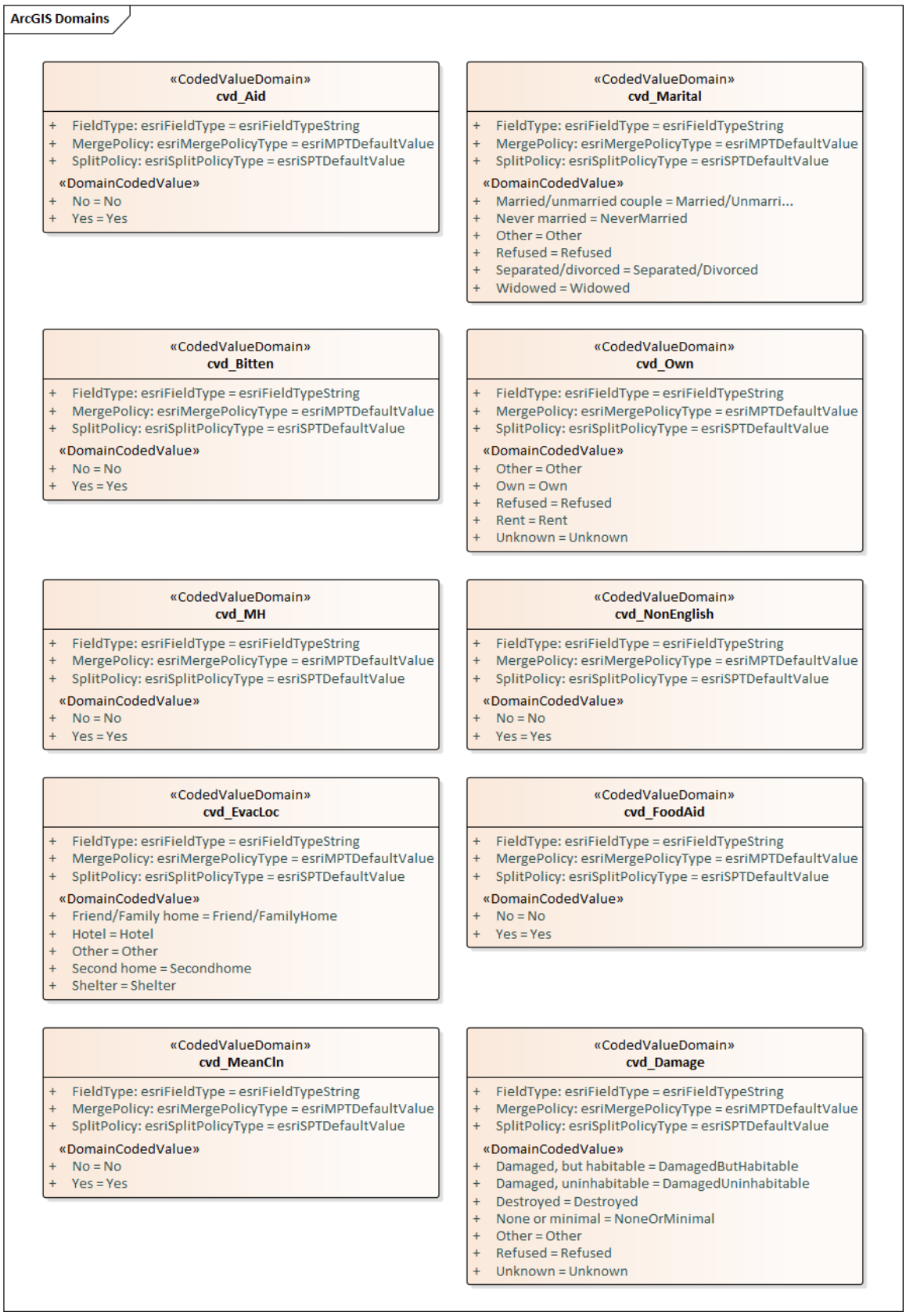

ArcGIS Domains
«CodedValueDomain»
cvd_FinRepair
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Mold
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Sewage
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_FindInfo
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Flyer/poster = Flyer/poster
+ Internet, = Internet
+ Local newspaper = Localnewspaper
+ Neighbor, word of mouth = Neighbor/wordofmouth
+ Other = Other
+ Radio = Radio
+ Refused = Refused
+ Text message = Textmessage
+ TV = TV
+ Unknown = Unknown
«CodedValueDomain»
cvd_WatAid
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_GrillWin
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_InfoSource
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Flyer/poster = Flyer/poster
+ Internet, = Internet
+ Local newspaper = Localnewspaper
+ Neighbor, word of mouth = Neighbor/wordofmouth
+ Other = Other
+ Radio = Radio
+ Refused = Refused
+ Text message = Textmessage
+ TV = TV
+ Unknown = Unknown
«CodedValueDomain»
cvd_PrimRes
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Died
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Preg
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes

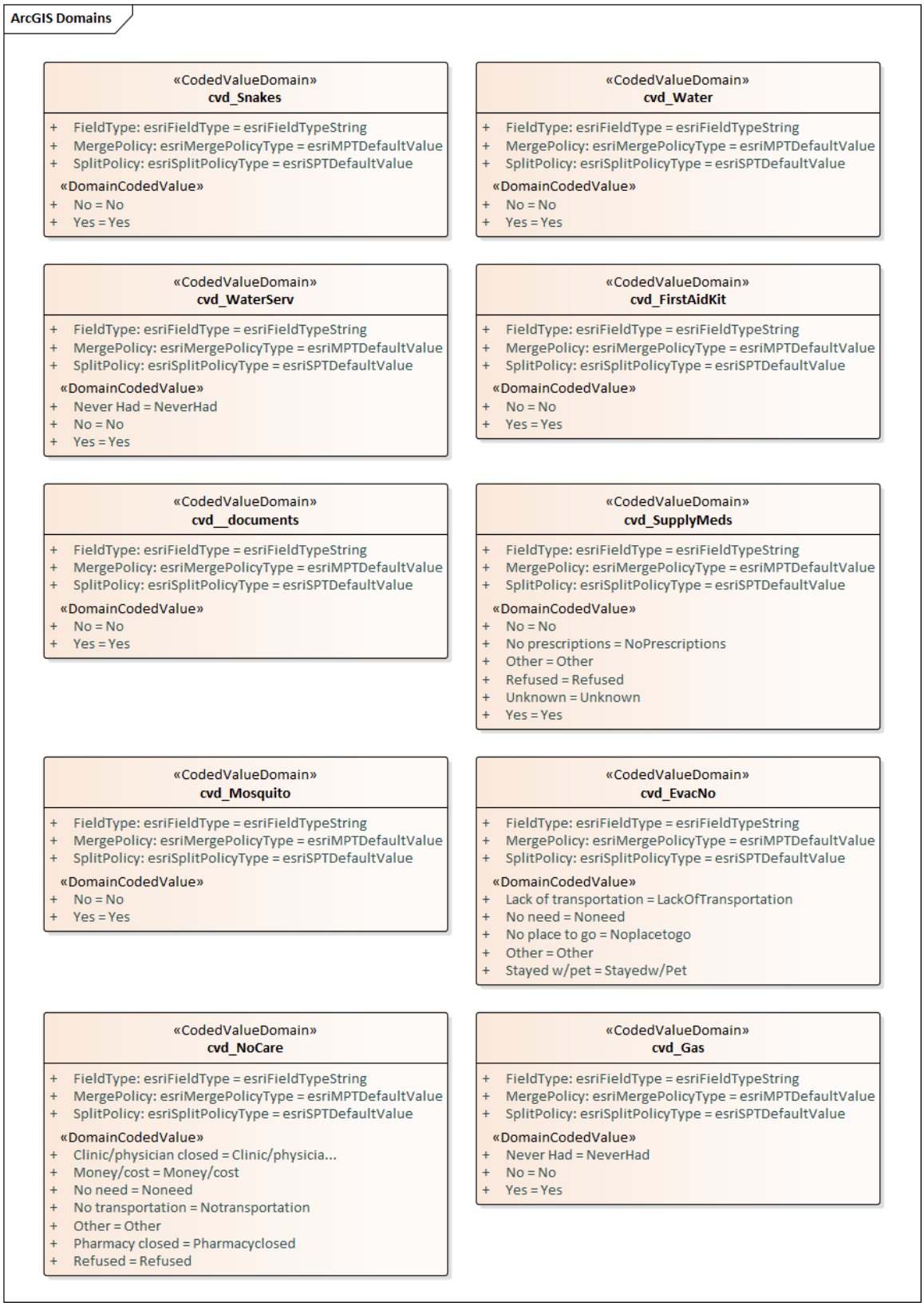

ArcGIS Domains
«CodedValueDomain»
cvd_Condition
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Clean up–living at home = Cleanup–LivingAtHome
+ Clean up–not living at home = Cleanup–notLivi...
+ Home never damaged or finished clean-up = HomeNeverDamOrF...
+ Home uninhabitable–not living at home = HomeUninhabitno...
+ Living in the home (no cleanup) = LivinginTheHome...
+ Other = Other
+ Refused = Refused
+ Unknown = Unknown
«CodedValueDomain»
cvd_TempRes
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Tetanus
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Worried_Rent
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Always = Always
+ Never = Never
+ Rarely = Rarely
+ Sometimes = Sometimes
+ Usually = Usually
«CodedValueDomain»
cvd_Grill
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Inside w/windows closed = Insidew/windows...
+ Inside w/windows open = Insidew/windowsopen
+ No = No
+ Other = Other
+ Outside = Outside
+ Refused = Refused
+ Unknown = Unknown
«CodedValueDomain»
cvd_SupplyKit
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Illness
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Clothes
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Fuel
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_DailyTask
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes

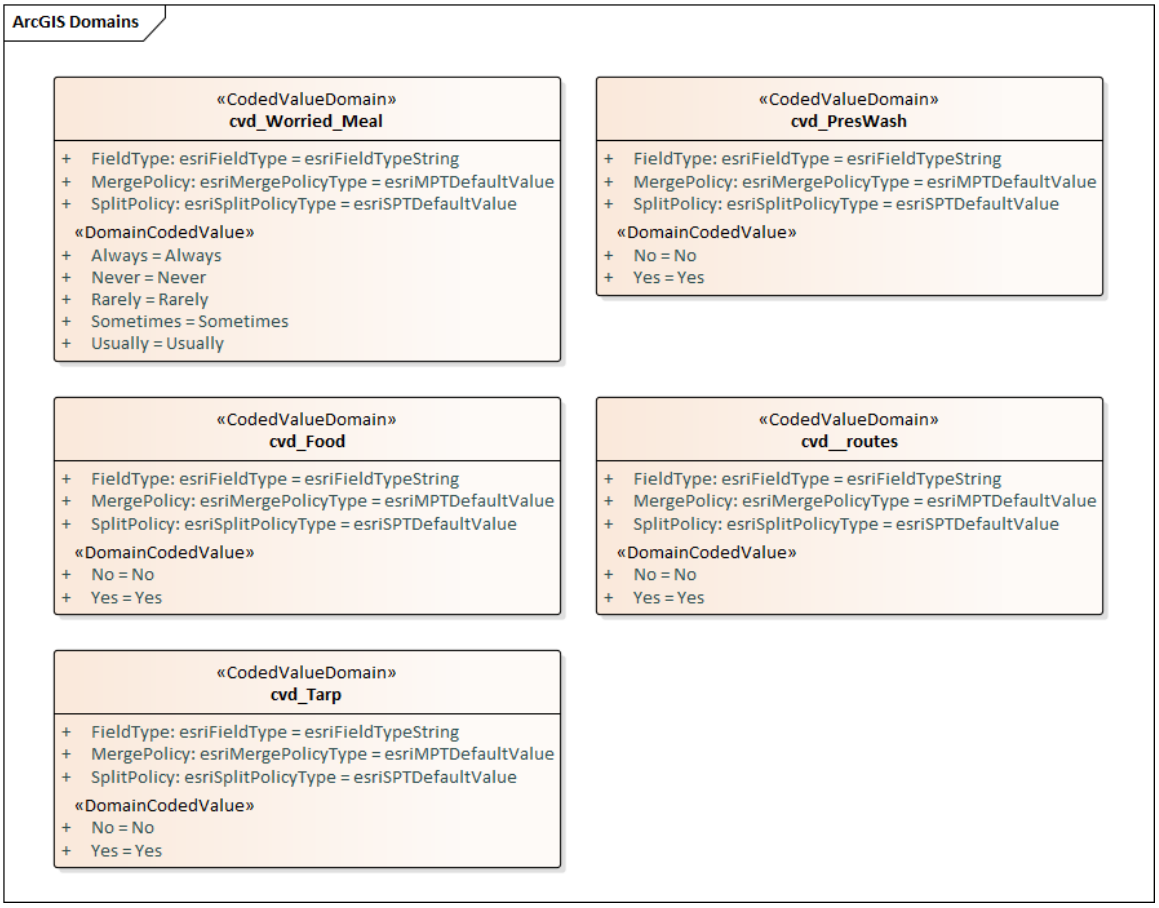

ArcGIS Domains
«CodedValueDomain»
cvd__access
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Chelpful
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Electric
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_MedCare
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Deaf
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_GenUse
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Phone
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_CareNeed
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd__meetOut
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ N/A = N/A
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Blind
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Concentrate
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Boil
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes


ArcGIS Domains
«CodedValueDomain»
cvd_Heat
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Coal/charcoal = Coal/charcoal
+ Electricity = Electricity
+ Never had = Neverhad
+ No heat = NoHeat
+ Other = Other
+ Propane/gas = Propane/gas
+ Refused = Refused
+ Unknown = Unknown
+ Wood = Wood
«CodedValueDomain»
cvd_GenLoc
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Garage = Garage
+ Inside = Inside
+ Other = Other
+ Outside, <25 feet = OutsideLess25feet
+ Outside, ≥25 feet = OutsideMore25feet
+ Refused = Refused
+ Unknown = Unknown
«CodedValueDomain»
cvd_FinClean
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Alligators
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_InsClean
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Floods
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Garbage
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_SearchInfo
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Transport
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Safe
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes


ArcGIS Domains
«CodedValueDomain»
cvd_ElectricSource
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Generator = Generator
+ Never had = Neverhad
+ No electricity = NoElectricity
+ Other = Other
+ Power company = PowerCompany
+ Refused = Refused
+ Unknown = Unknown
«CodedValueDomain»
cvd_WNow
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Bottled = Bottled
+ No drinking water = Nodrinkingwater
+ Other = Other
+ Public/municipal (tap) = Public/municipa...
+ Refused = Refused
+ Unknown = Unknown
+ Well = Well
«CodedValueDomain»
cvd_MoldDam
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_WTreat
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Other = Other
+ Refused = Refused
+ Unknown = Unknown
+ Yes, boiling = YesBoiling
+ Yes, chemical = YesChemical
«CodedValueDomain»
cvd_InsRepair
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Hispanic
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_CODetect
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Cell
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Injury
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd__commPlan
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Live1Mo
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Buy/rent different residence in this state = Buy/RentDifResi...
+ In this residence = InThisResidence
+ Shelter = Shelter
+ With family/friends = WithFamily/Friends
«CodedValueDomain»
cvd_Campaign
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes

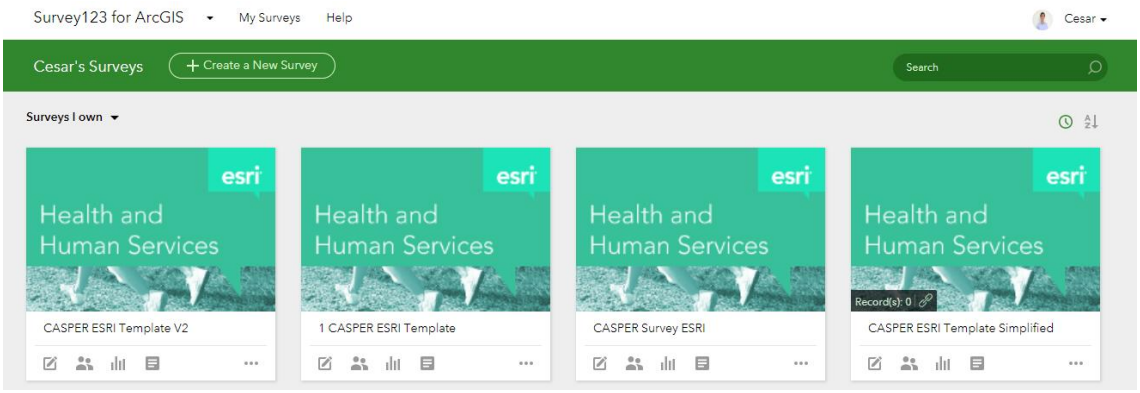

ArcGIS Domains
«CodedValueDomain»
cvd_Aid
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Marital
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Married/unmarried couple = Married/Unmarri...
+ Never married = NeverMarried
+ Other = Other
+ Refused = Refused
+ Separated/divorced = Separated/Divorced
+ Widowed = Widowed
«CodedValueDomain»
cvd_Bitten
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Own
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Other = Other
+ Own = Own
+ Refused = Refused
+ Rent = Rent
+ Unknown = Unknown
«CodedValueDomain»
cvd_MH
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_NonEnglish
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_EvacLoc
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Friend/Family home = Friend/FamilyHome
+ Hotel = Hotel
+ Other = Other
+ Second home = Secondhome
+ Shelter = Shelter
«CodedValueDomain»
cvd_FoodAid
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_MeanCln
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Damage
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Damaged, but habitable = DamagedButHabitable
+ Damaged, uninhabitable = DamagedUninhabitable
+ Destroyed = Destroyed
+ None or minimal = NoneOrMinimal
+ Other = Other
+ Refused = Refused
+ Unknown = Unknown


ArcGIS Domains
«CodedValueDomain»
cvd_Snakes
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Water
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_WaterServ
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_FirstAidKit
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd__documents
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_SupplyMeds
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ No prescriptions = NoPrescriptions
+ Other = Other
+ Refused = Refused
+ Unknown = Unknown
+ Yes = Yes
«CodedValueDomain»
cvd_Mosquito
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_EvacNo
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Lack of transportation = LackOfTransportation
+ No need = Noneed
+ No place to go = Noplacetogo
+ Other = Other
+ Stayed w/pet = Stayedw/Pet
«CodedValueDomain»
cvd_NoCare
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Clinic/physician closed = Clinic/physicia...
+ Money/cost = Money/cost
+ No need = Noneed
+ No transportation = Notransportation
+ Other = Other
+ Pharmacy closed = Pharmacyclosed
+ Refused = Refused
«CodedValueDomain»
cvd_Gas
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Never Had = NeverHad
+ No = No
+ Yes = Yes

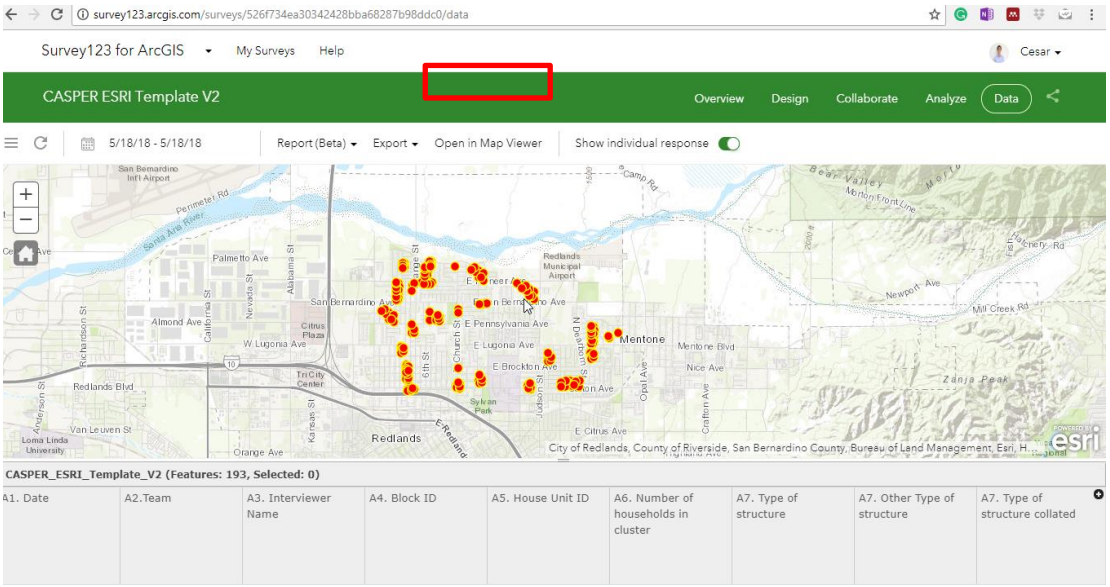

ArcGIS Domains
«CodedValueDomain»
cvd_Worried_Meal
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ Always = Always
+ Never = Never
+ Rarely = Rarely
+ Sometimes = Sometimes
+ Usually = Usually
«CodedValueDomain»
cvd_PresWash
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Food
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd__routes
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes
«CodedValueDomain»
cvd_Tarp
+ FieldType: esriFieldType = esriFieldTypeString
+ MergePolicy: esriMergePolicyType = esriMPTDefaultValue
+ SplitPolicy: esriSplitPolicyType = esriSPTDefaultValue
«DomainCodedValue»
+ No = No
+ Yes = Yes

Appendix D. Manual for Dashboard Configuration

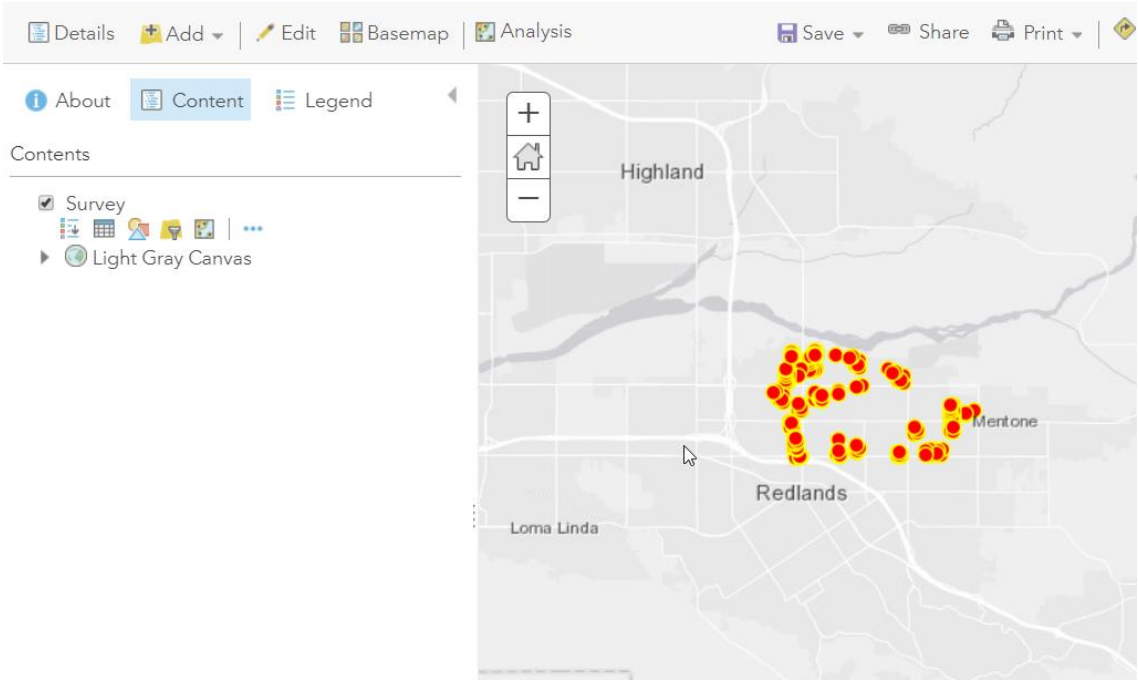
Once the people in the field is conducting the survey, thought ArcGIS Online you can monitor the number of surveys collected. Accessing to survey123.arcgis.com with your credentials and going to my surveys.



Click in the published survey and then click in the DATA tab. Fin,ally click in the option Open in a Map Viewer.

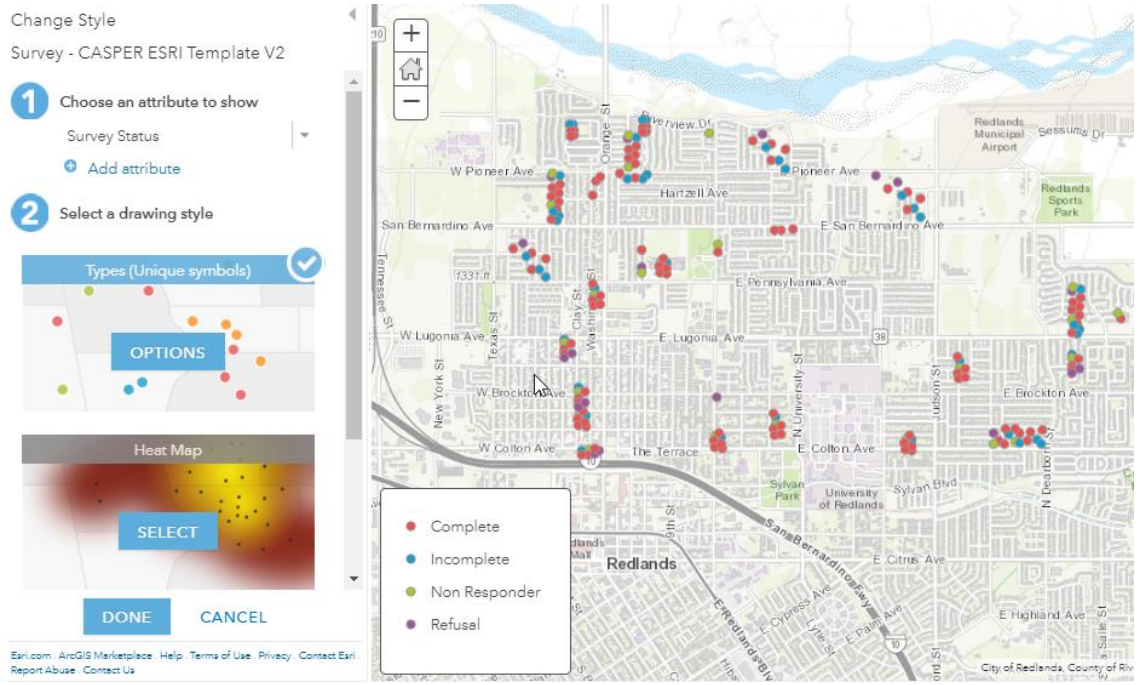


Once the data is open in the data viewer, change the name of the map and save it as

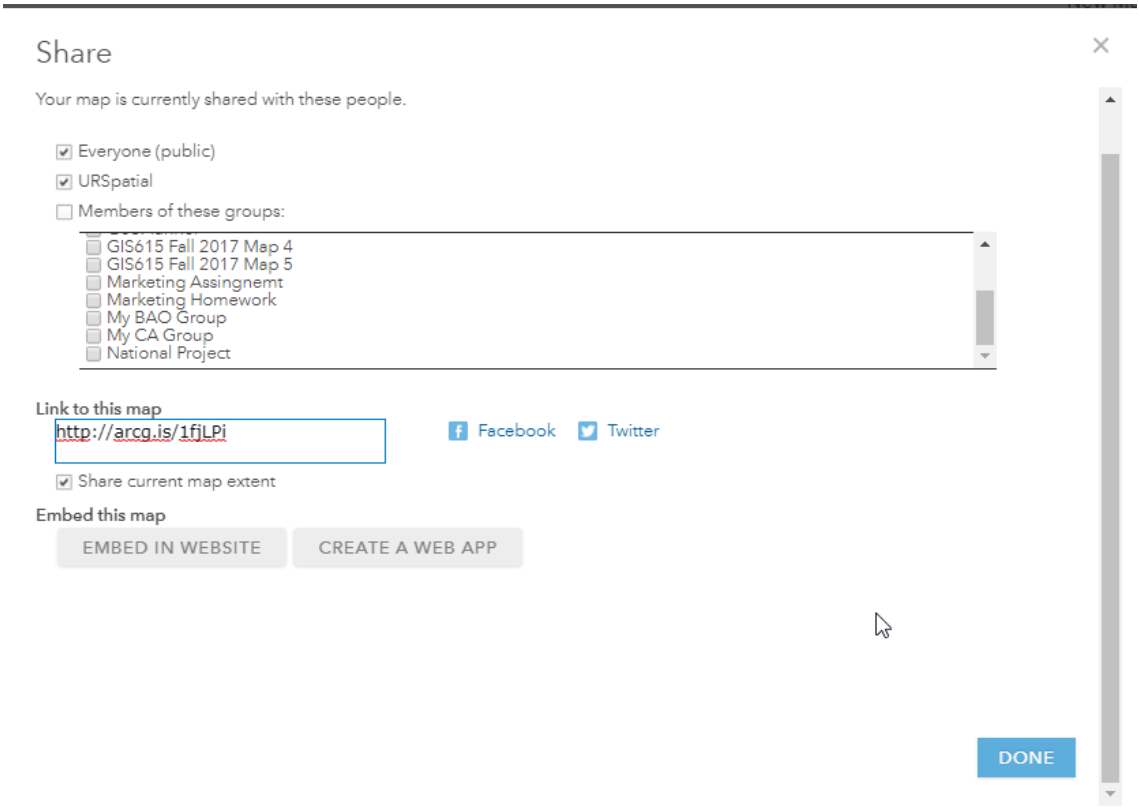


CASPER Status Monitoring

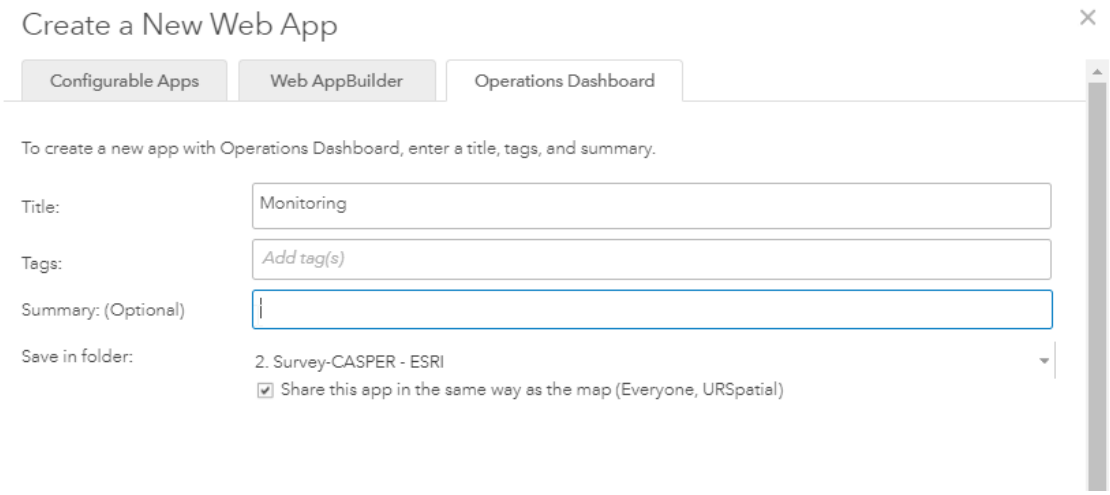
Change the symbology of the survey using the Survey Status field and Save.



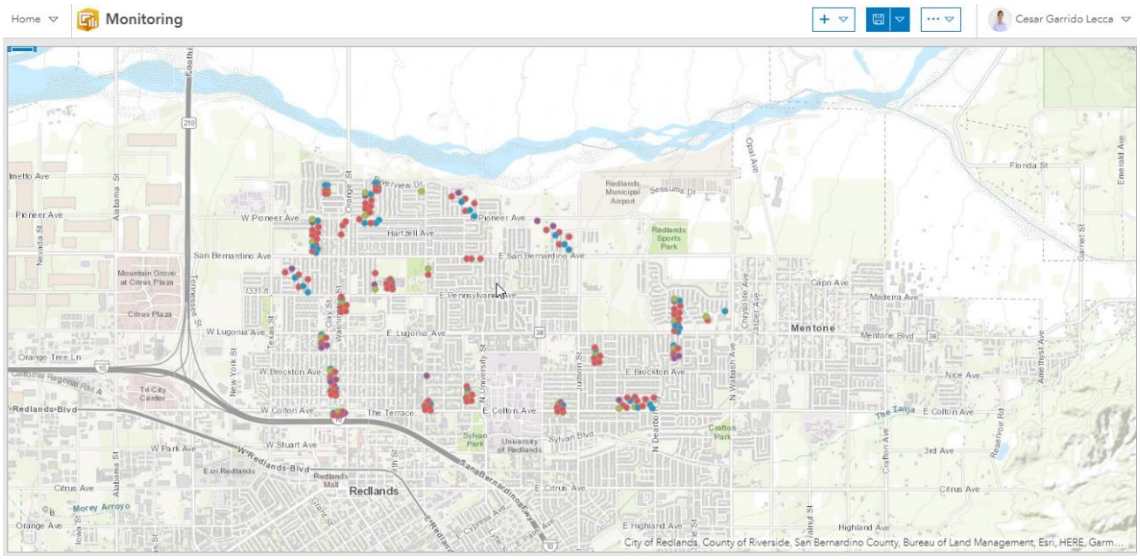
Click in the button share the Map and choose the option Create Web App



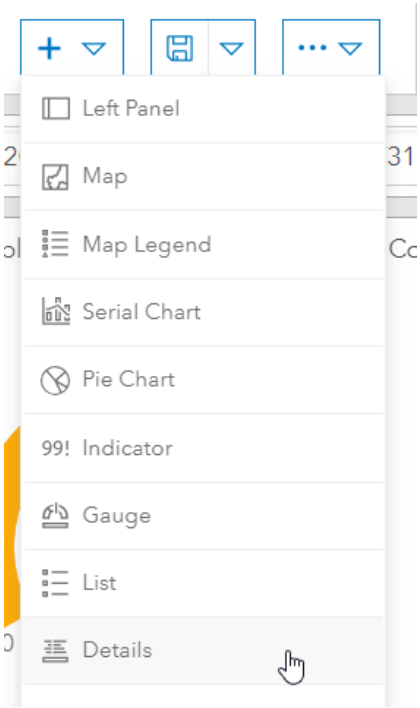
In the next window, click in the Operation Dashboard tab, add tags and the summary and create the dashboard.



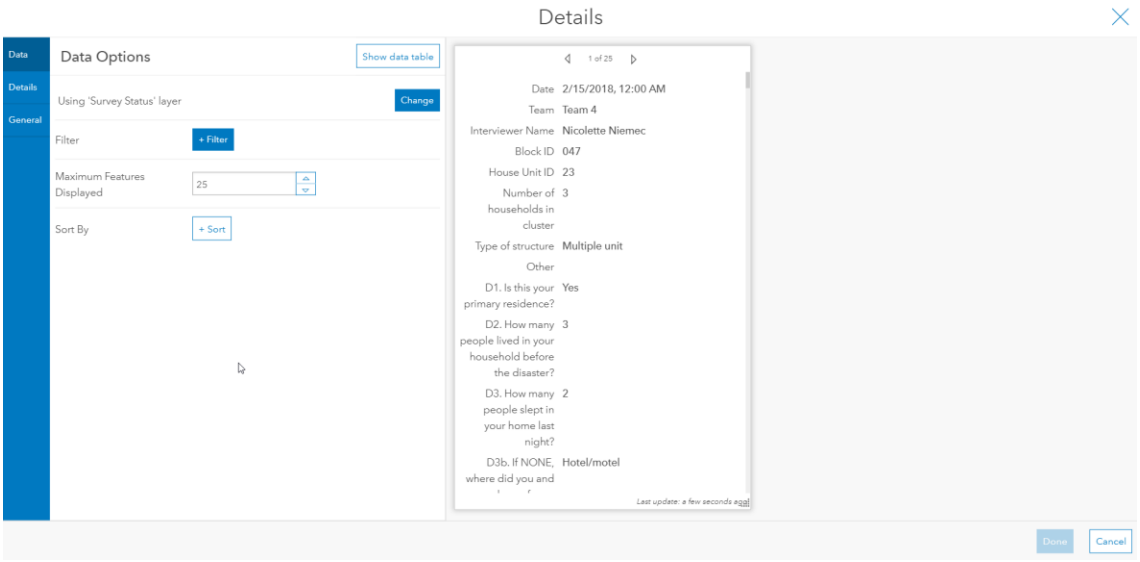
Finally, you will have a Dashboard ready to be configure.



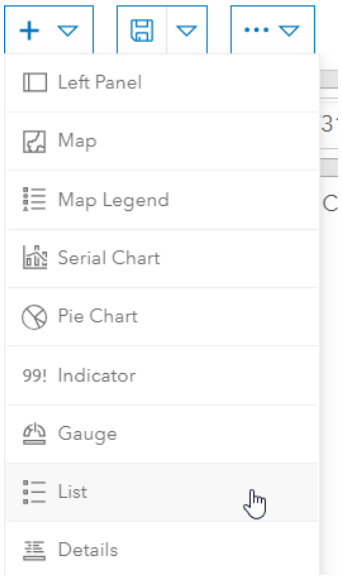
Click in the + symbol and add a Details



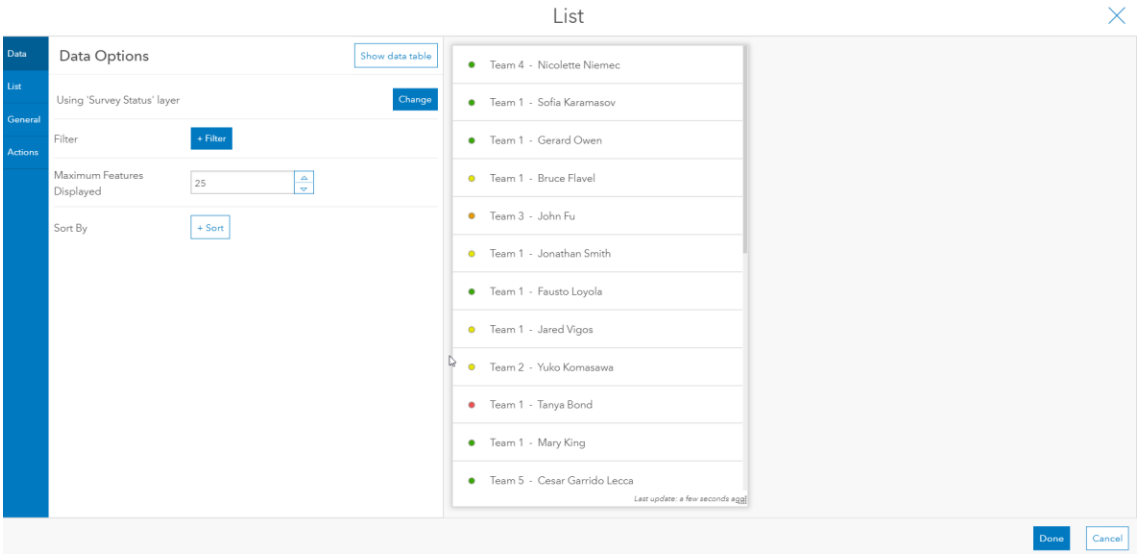
Select your survey as the source of information.



Click in the + symbol and add a List

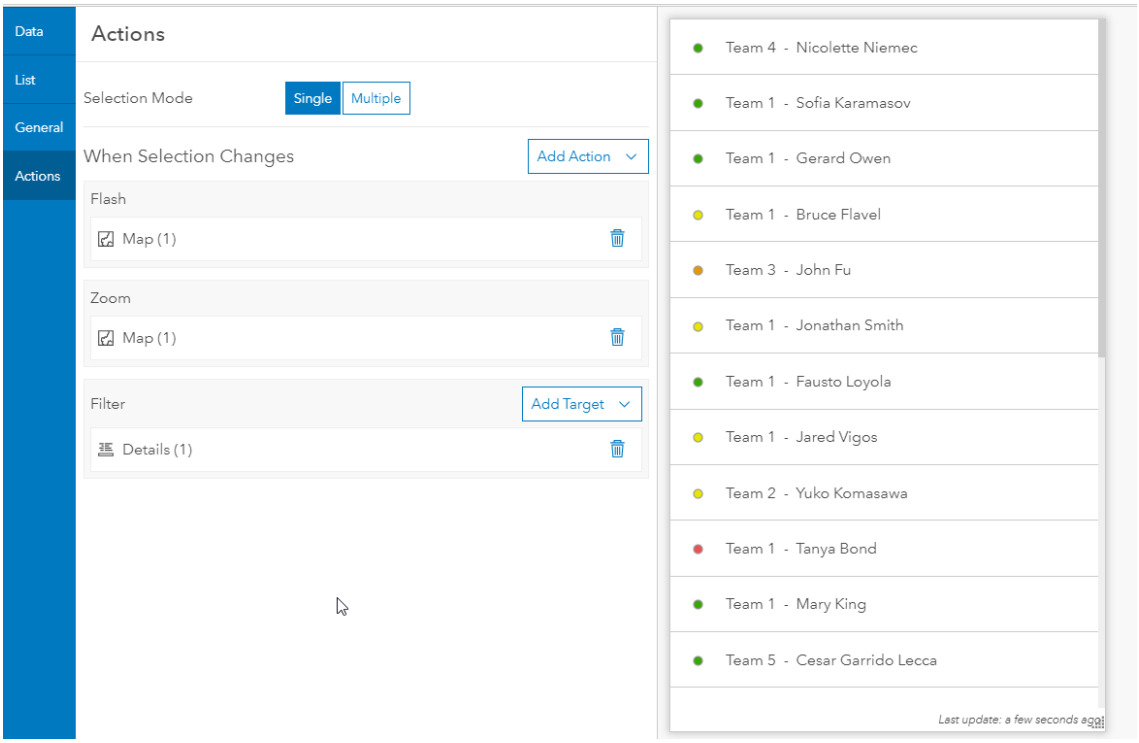


Select the survey as the source of information.

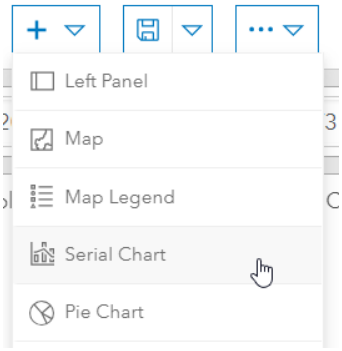


Select the actions tab and configure the following.

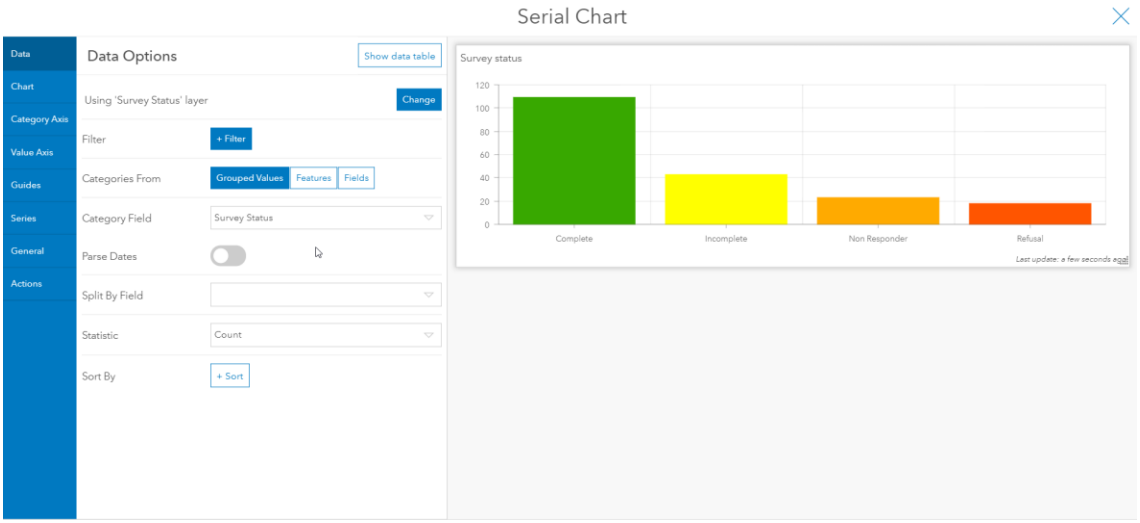
- Flash to the Map
- Zoom to the Map
- Filter to the Details Element.



Click in the + symbol and add a Serial chart



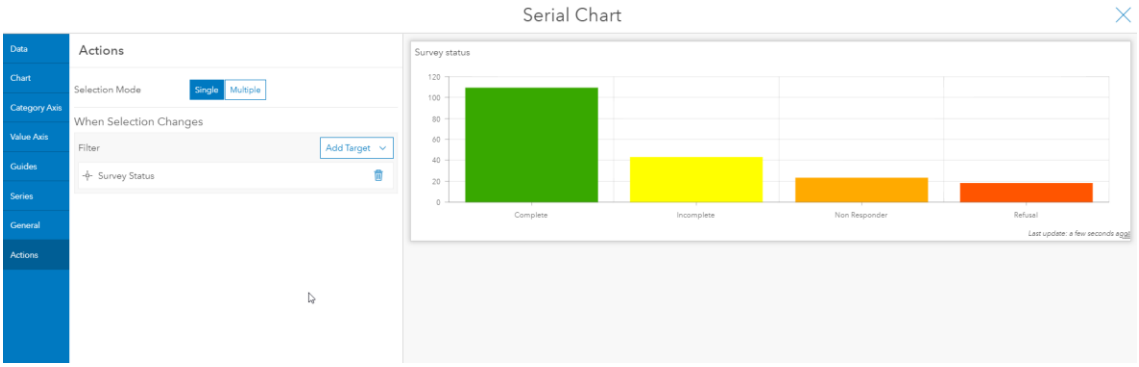
Select the survey as the source of information. Select Survey Status as Category Field.

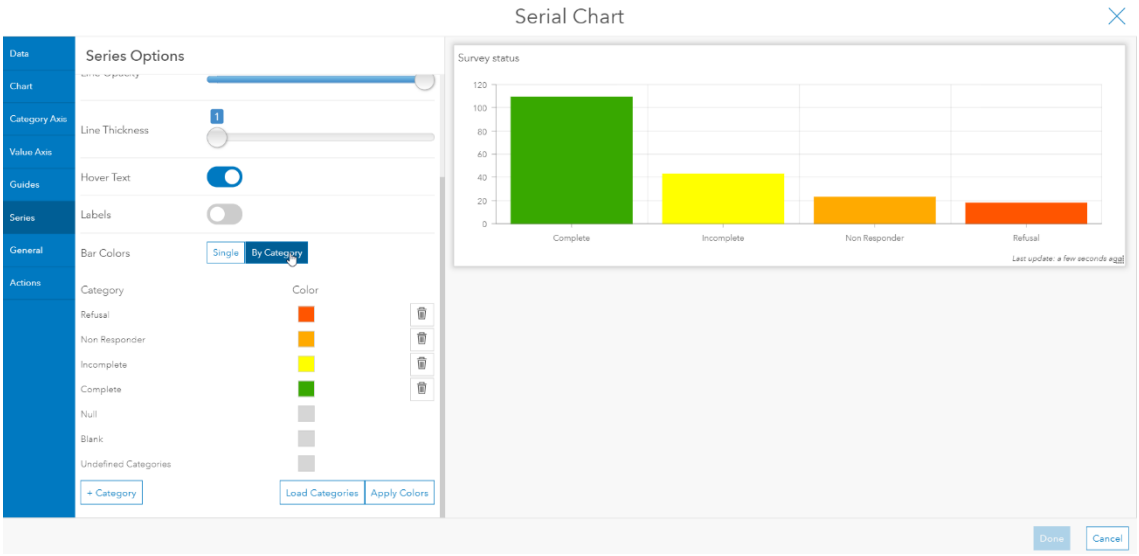


Go to the Series Tab and select Bar Colors by Category.

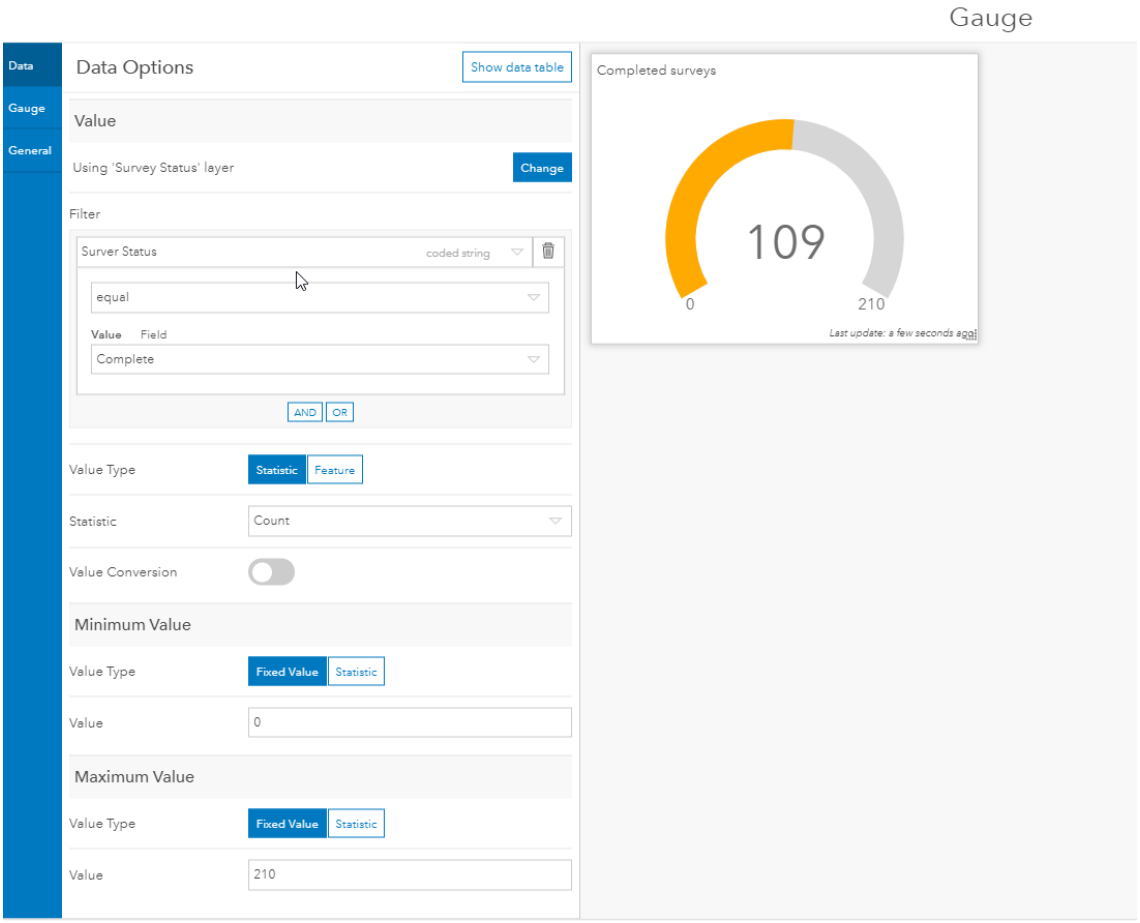
Select the actions tab and configure the following.

- Filter the Map

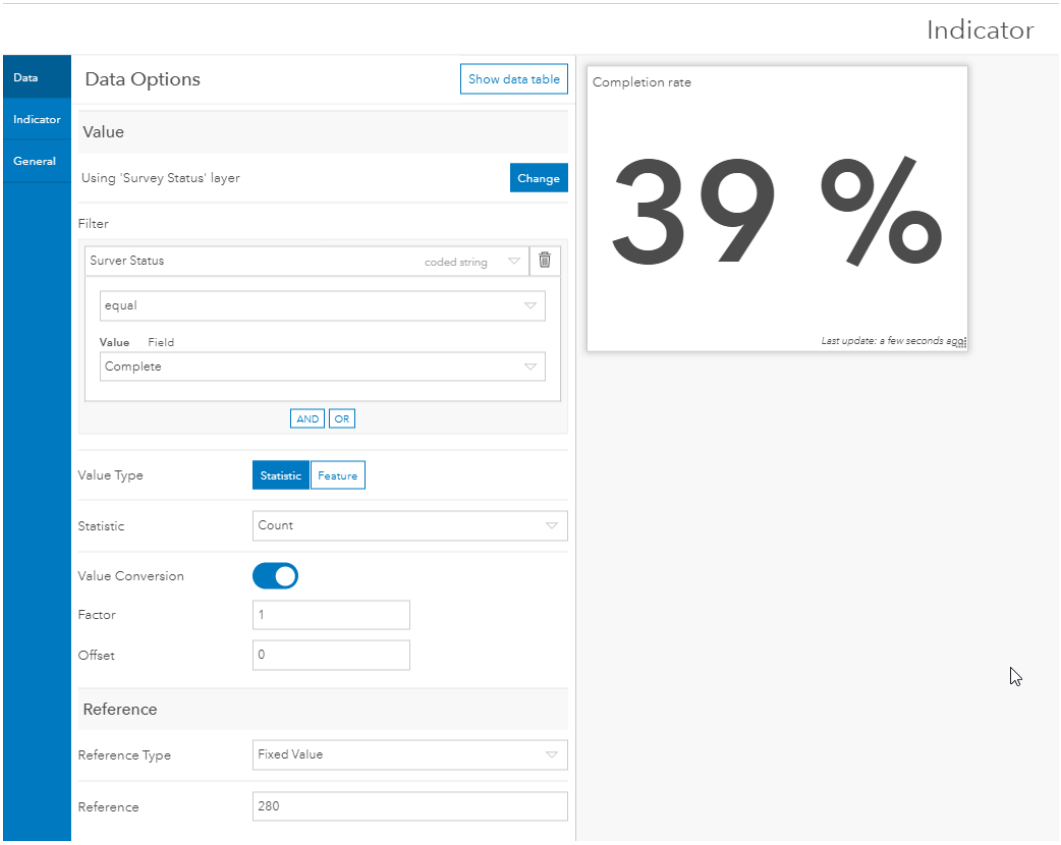




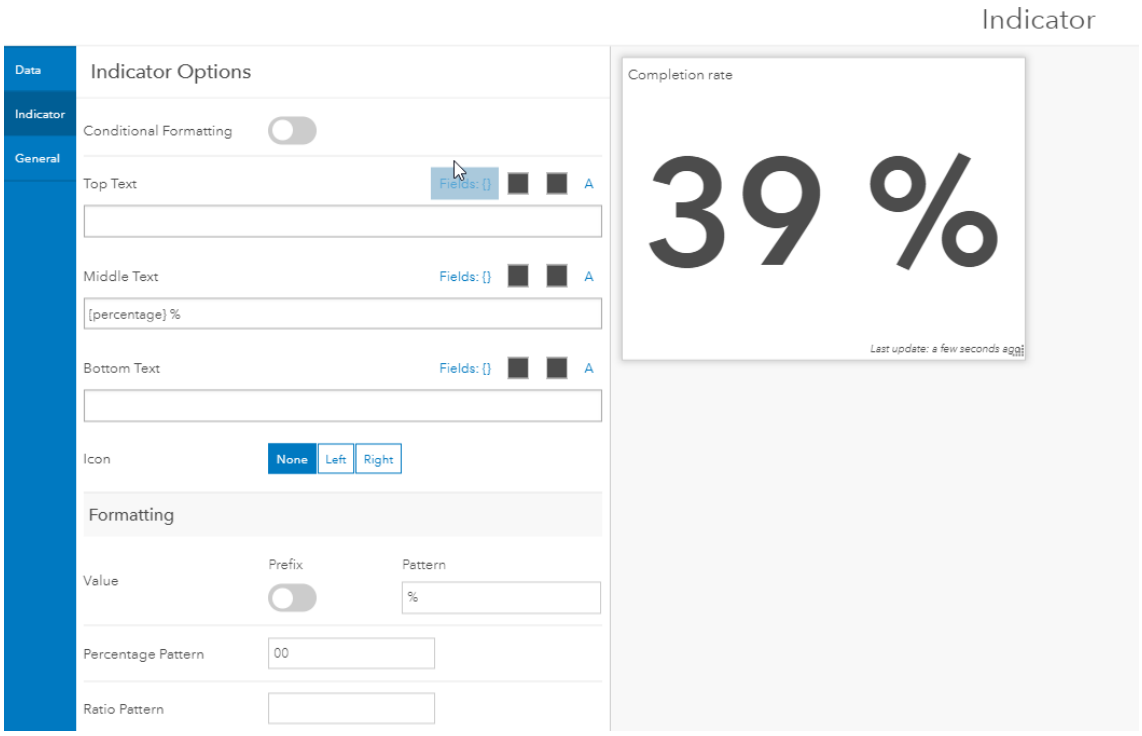
Click in the + symbol and add a Gauge. Click in Filter and Filter the Survey Data by Survey Status = Complete. Set 210 as Maximum Value.



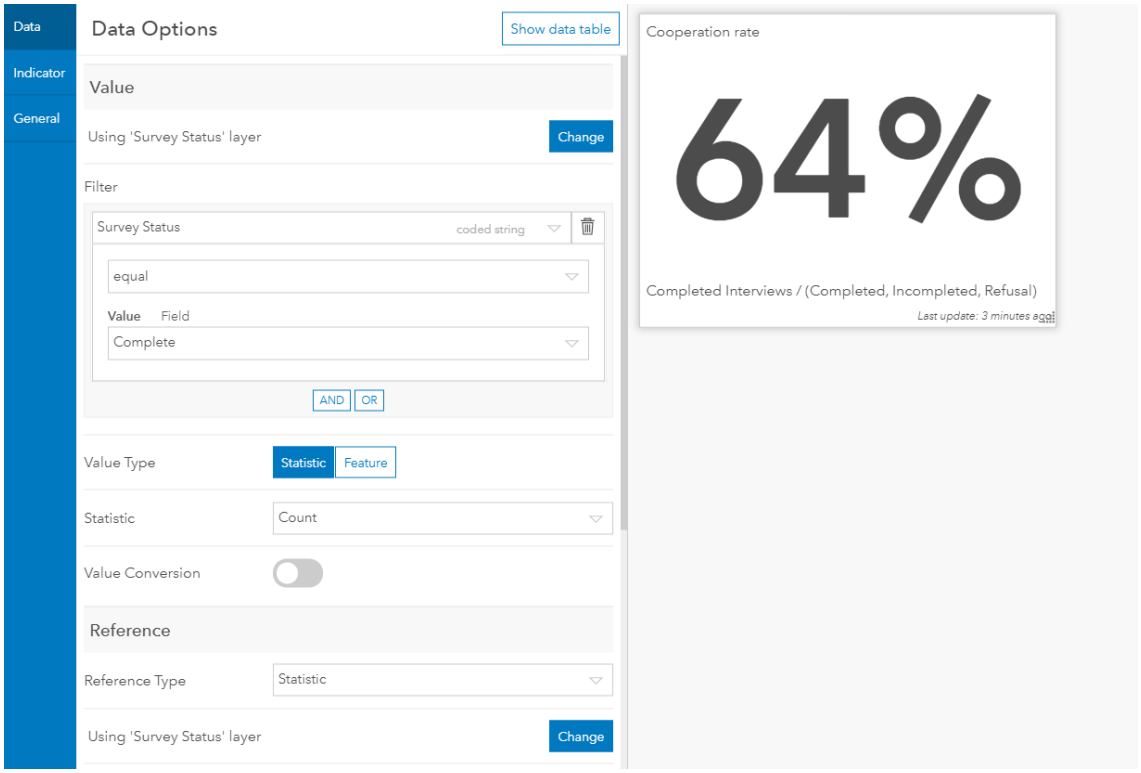
Click in the + symbol and add an Indicator. Click in Filter and filter the Survey Data by Survey Status = Complete. Set 210 as Fixed Value Reference.



Click the indicator Tab and set % in the value field and 00 in the Percentage Pattern.

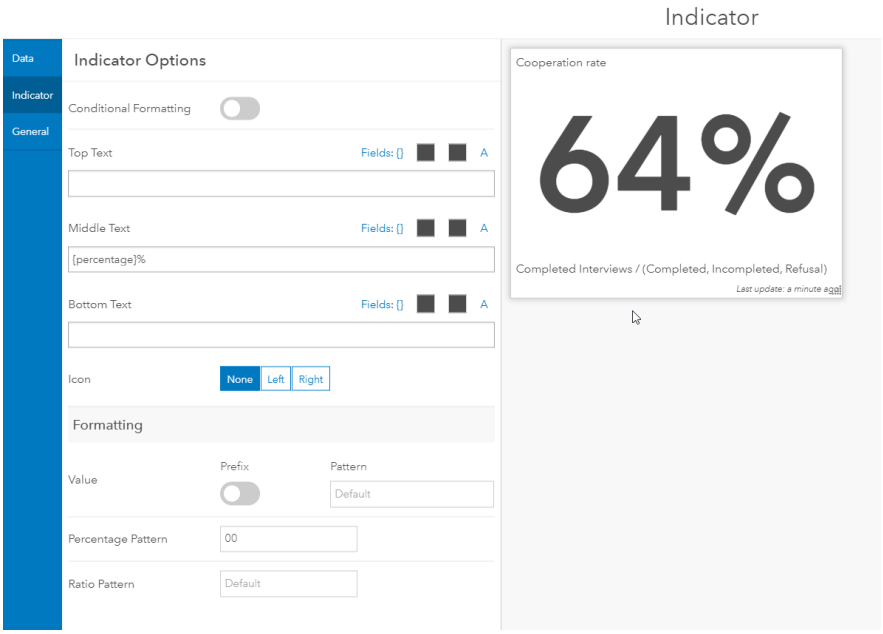


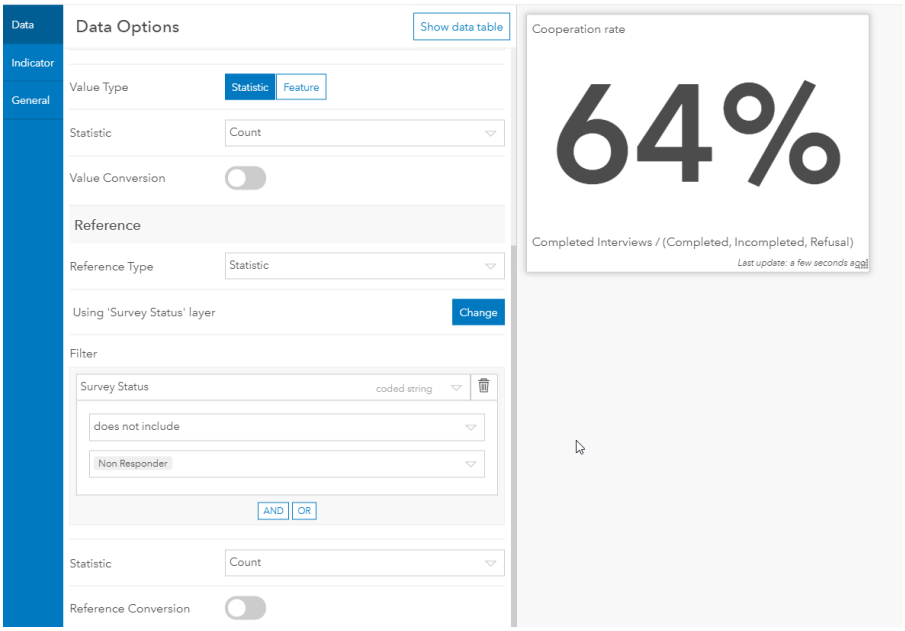
Click in the + symbol and add an Indicator. Click in Filter and filter the Survey Data by Survey Status = Complete.



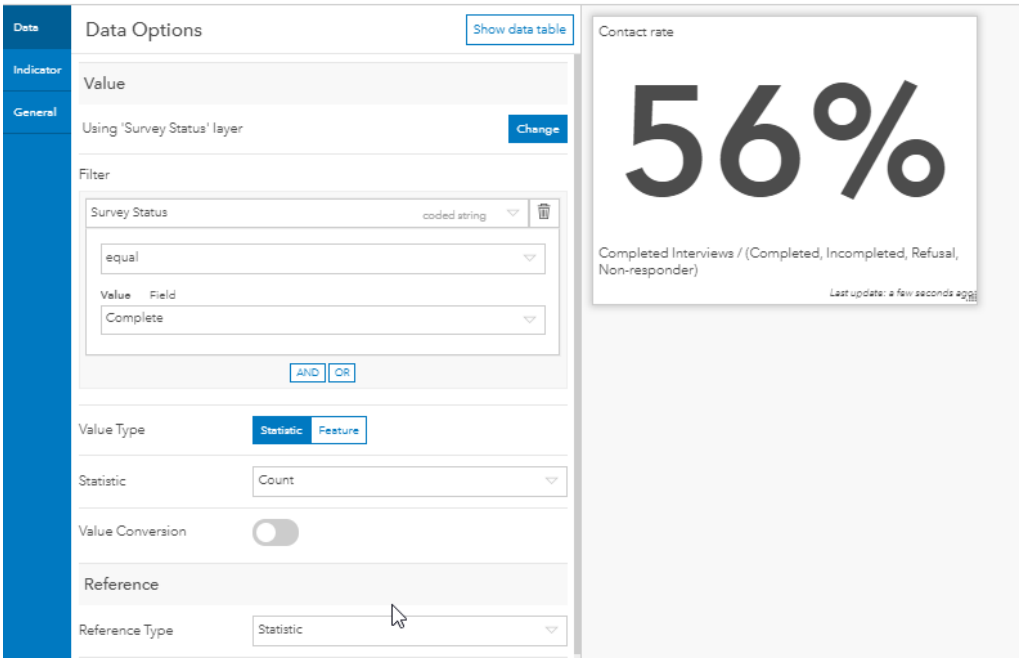
In the Reference Section configure the Survey Data filtered by Survey Status does not include Non-Responder.

Click the indicator Tab and set % in the value field and 00 in the Percentage Pattern

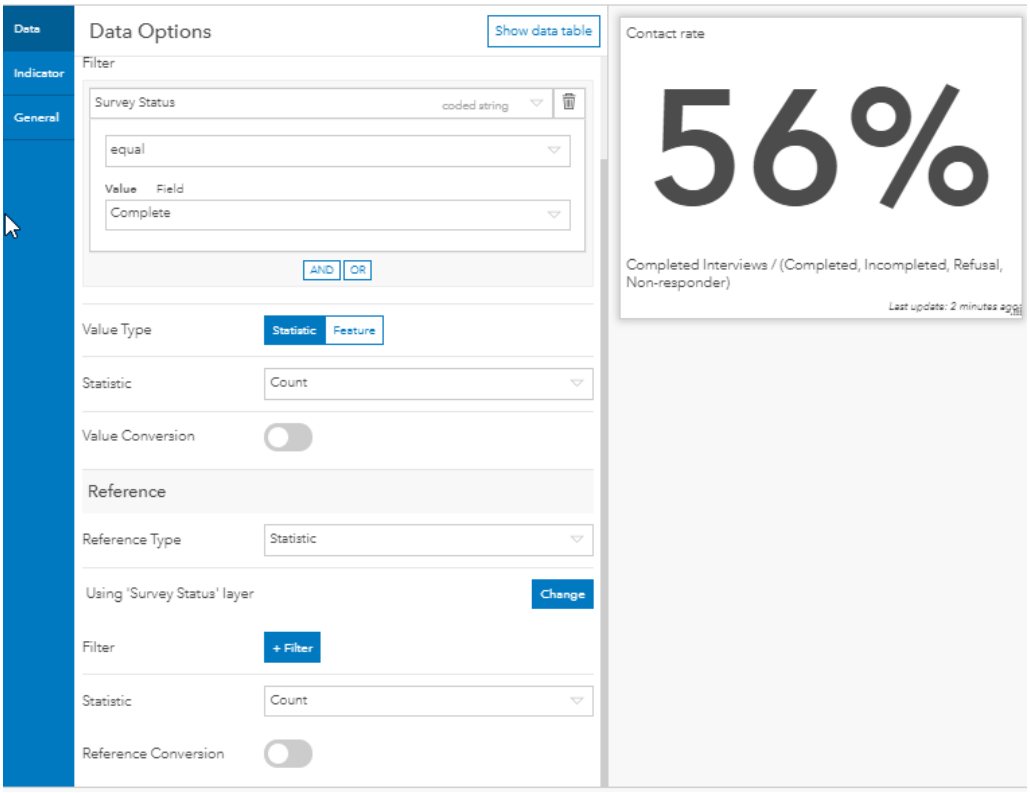




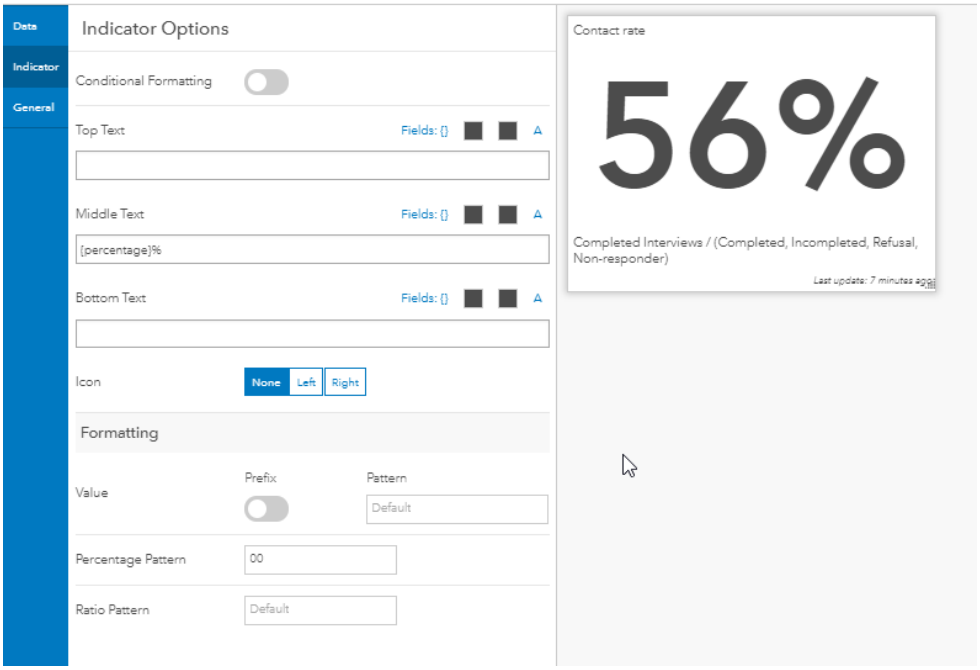
Click in the + symbol and add an Indicator. Click in Filter and filter the Survey Data by Survey Status = Complete



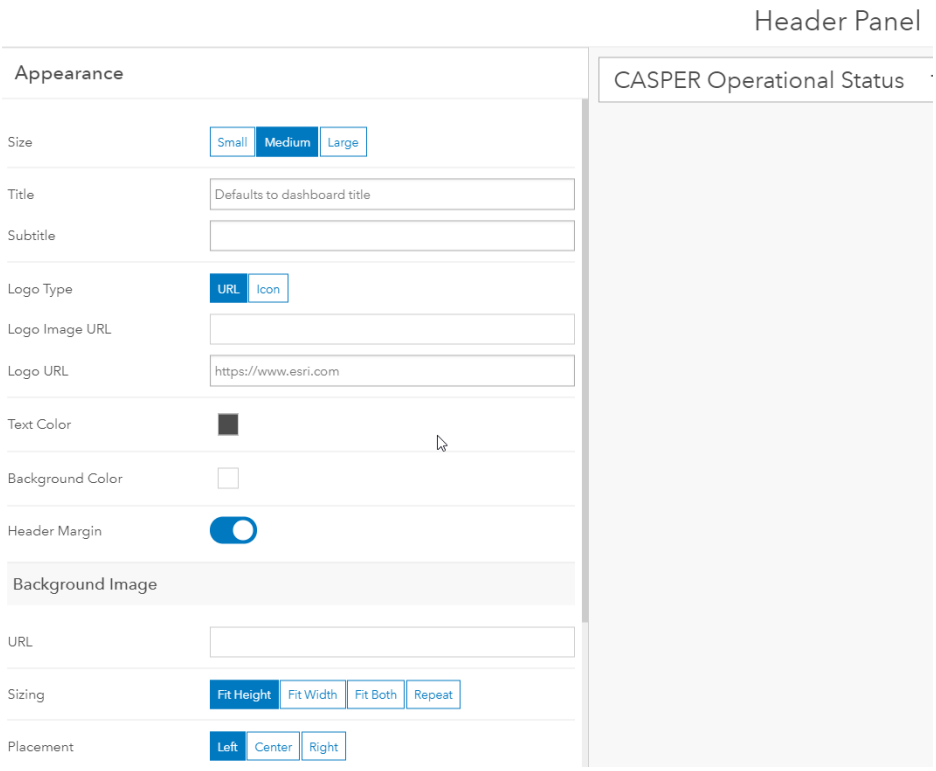
In the Reference Section configure the Survey Data without any filter.



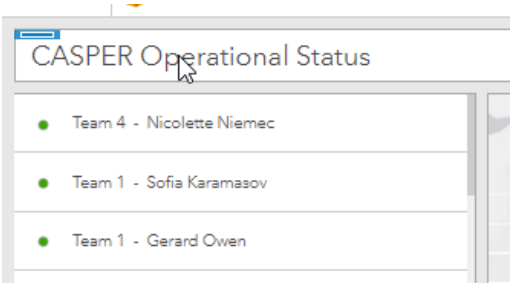
Click the indicator Tab and set % in the value field and 00 in the Percentage Pattern

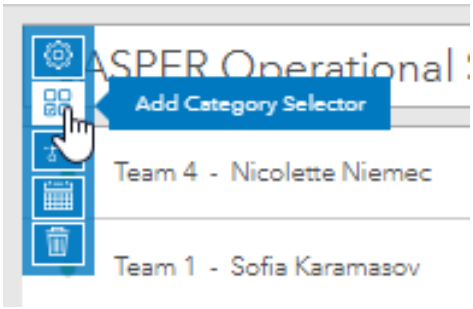


Click in the + symbol and add a Header. Configure the Options and Click ok.

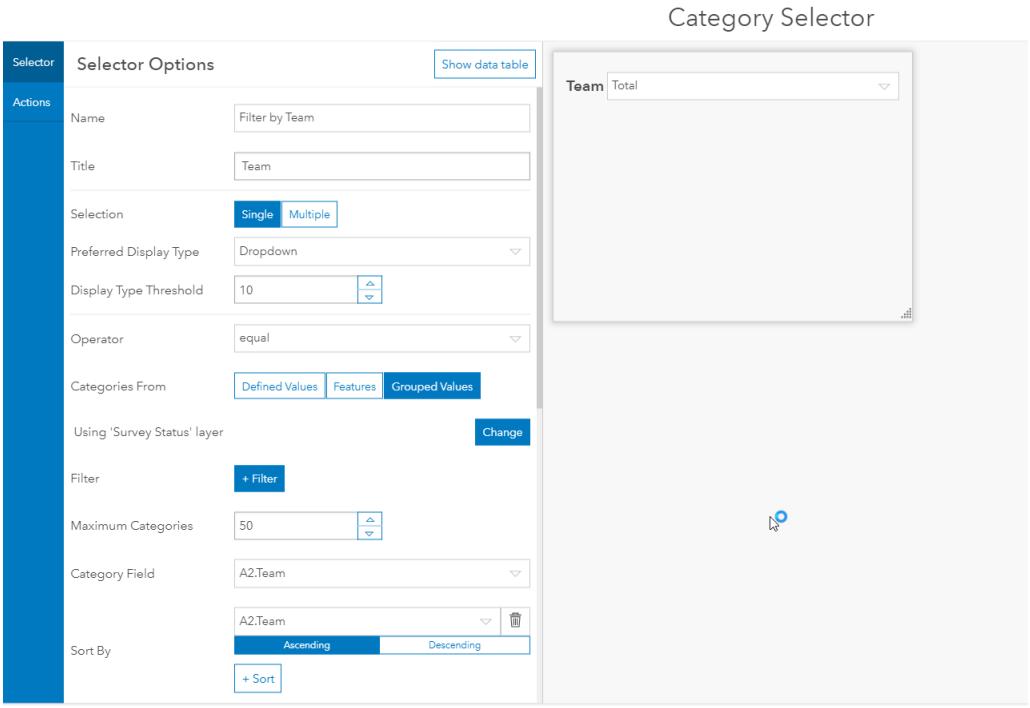


Once the Header is created, Click over the Header Options and select Category Selector.



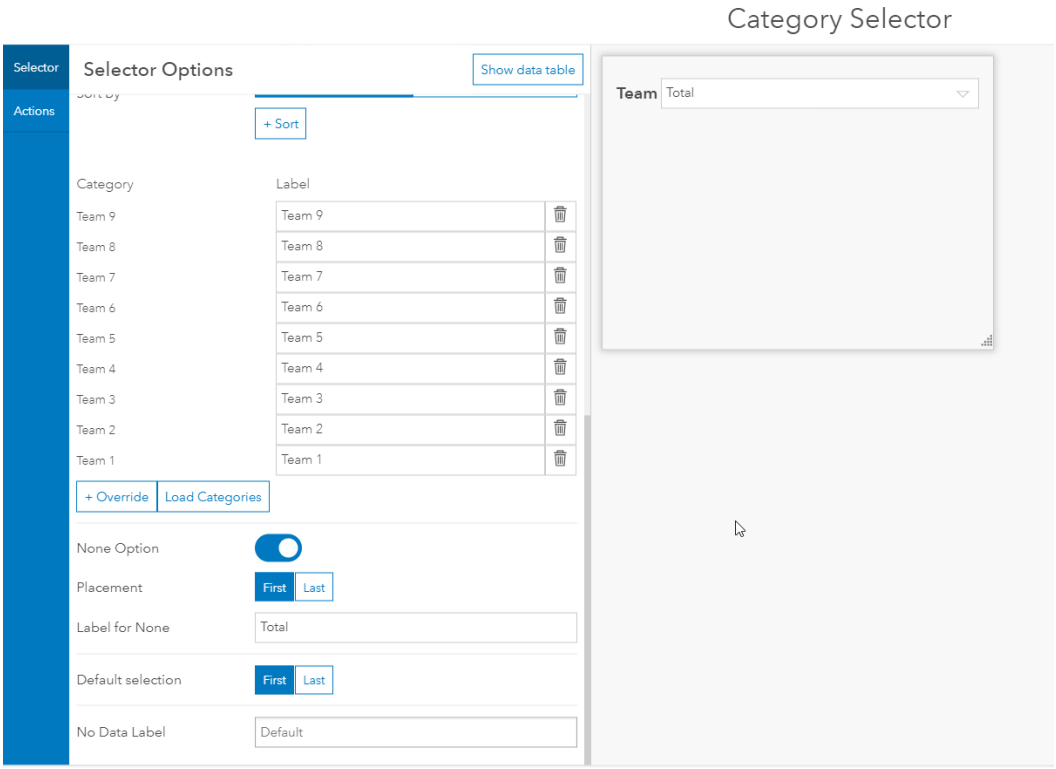


Add a Category Selector. Select A2. Teams as a Category Field.

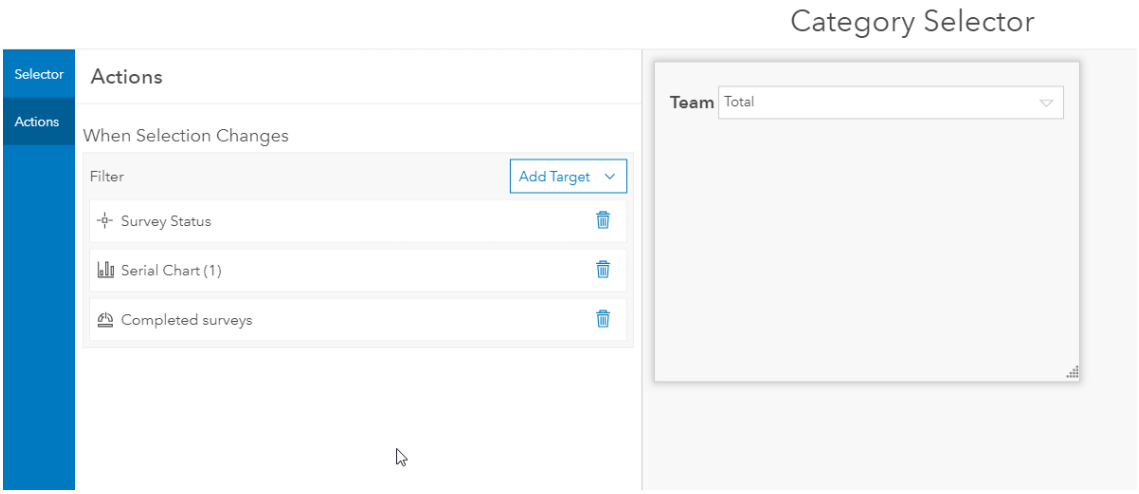


Make Sure all the category values have been loaded.

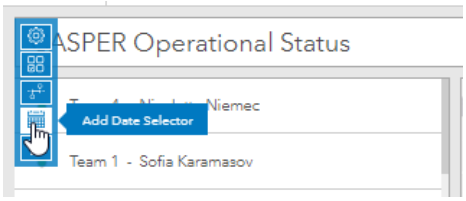
Select the actions tab and configure the following.



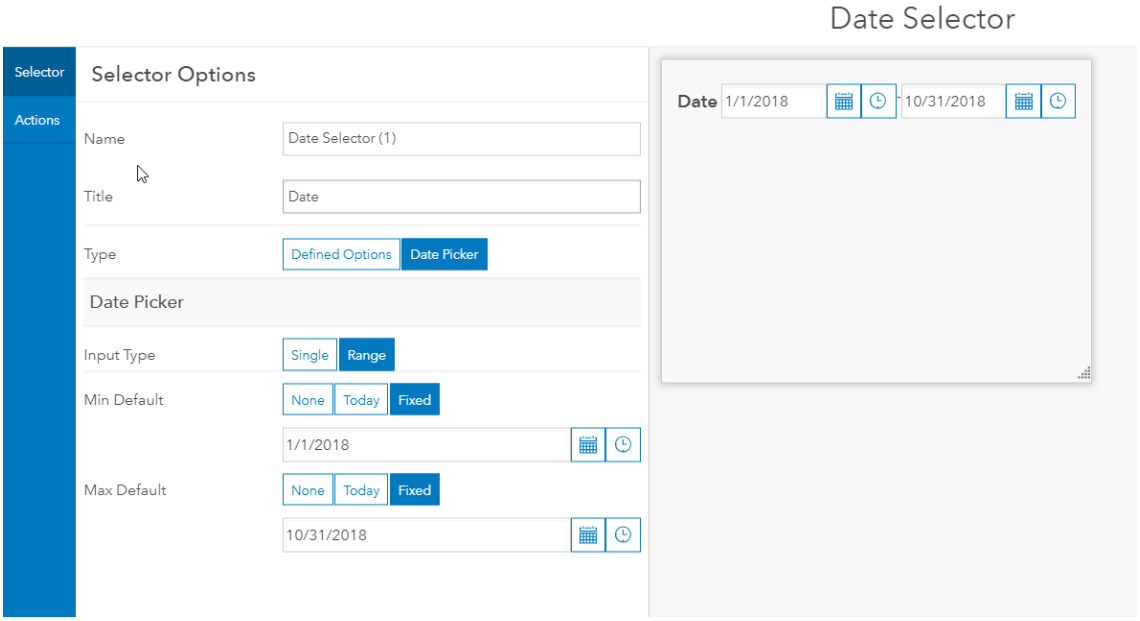
- Filter Map
- Filter Serial Chart
- Filter Gauge



Add a Date Picker.

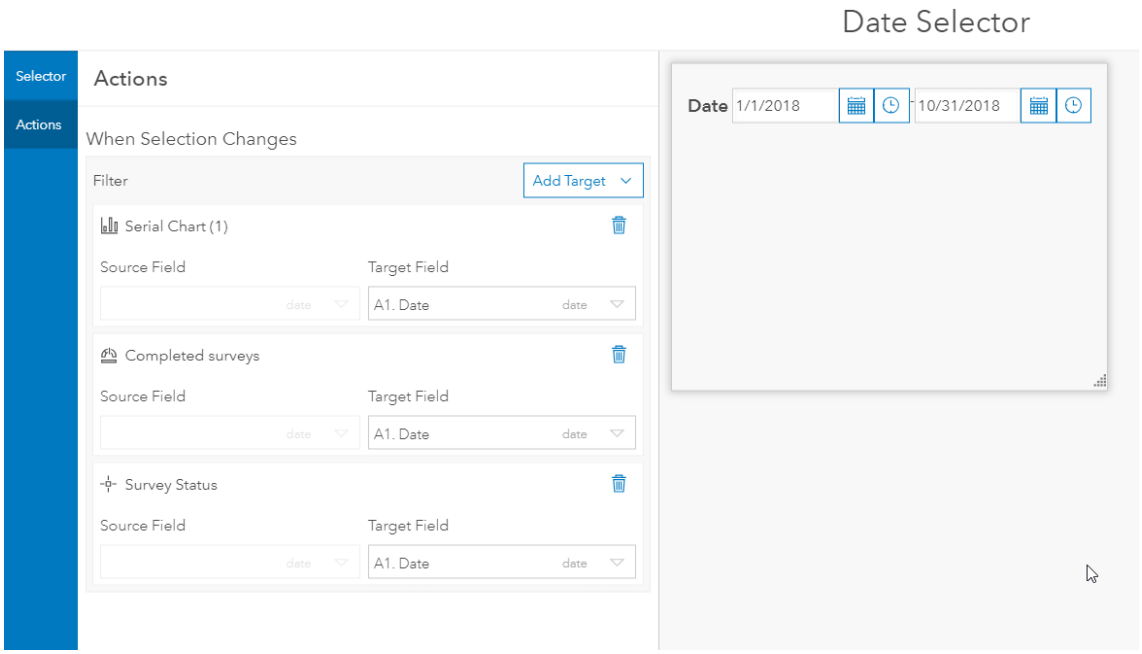


Configure the Range of dates to pick.



Select the actions tab and configure the following.

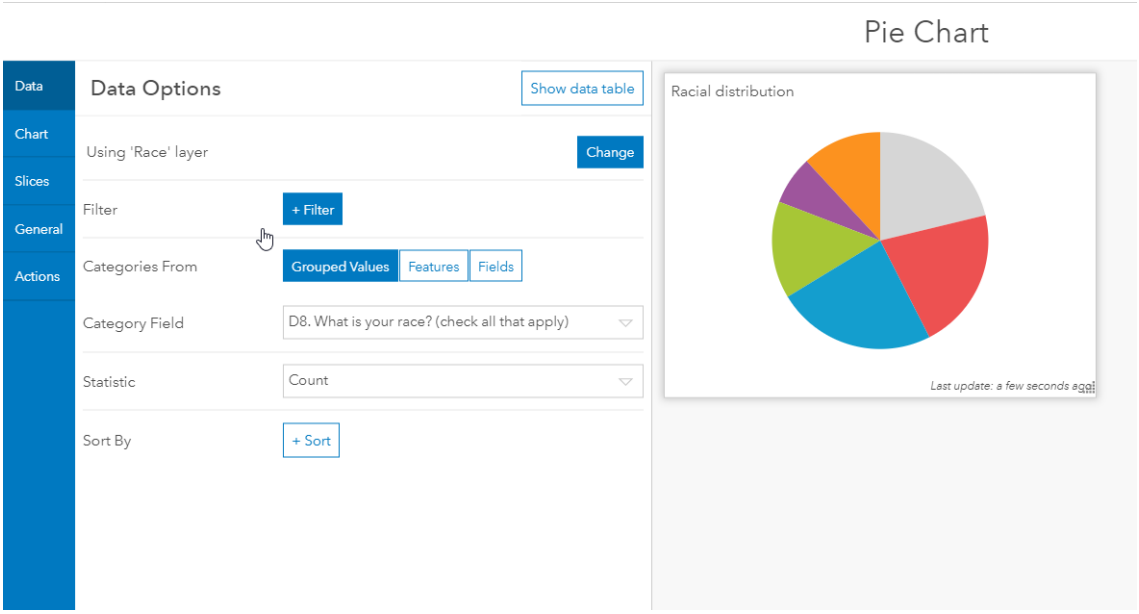
- Filter Map by date field
- Filter Serial Chart by date Field
- Filter Gauge by date field



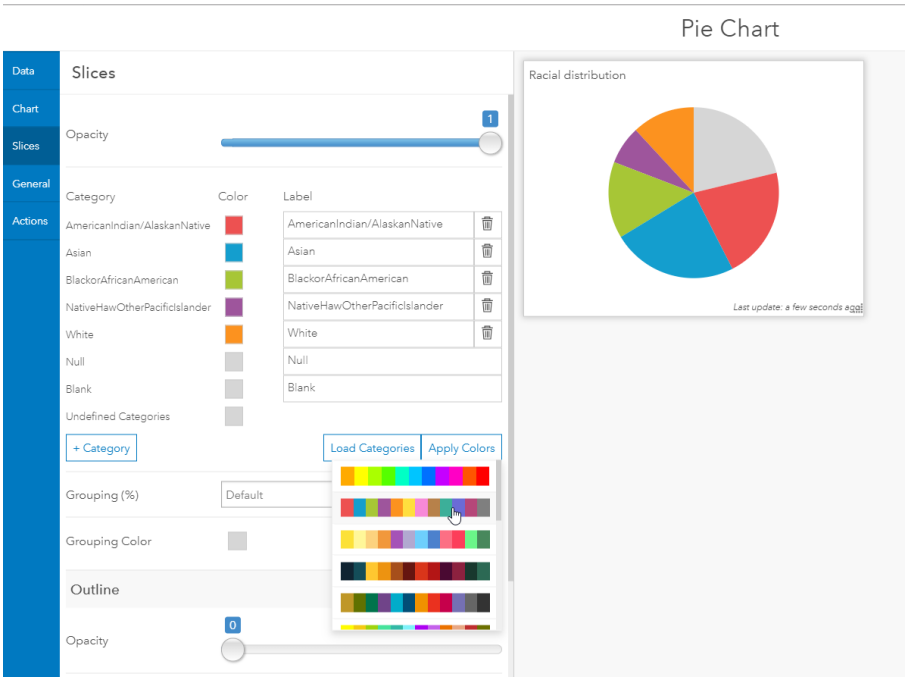
Configure the Demographics Dashboard

Follow the same Step to create a new Web Map and start configuring the Demographic dashboard.

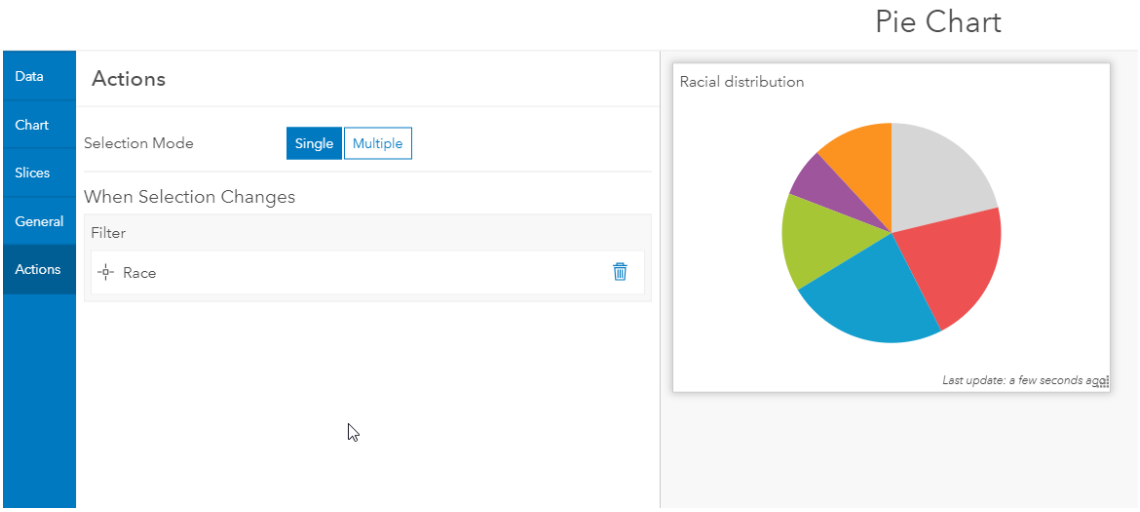
Click in the + symbol and add a Pie Chart. Select the D8. What is your race question?



Click the Slices Tab and click Load Categories.

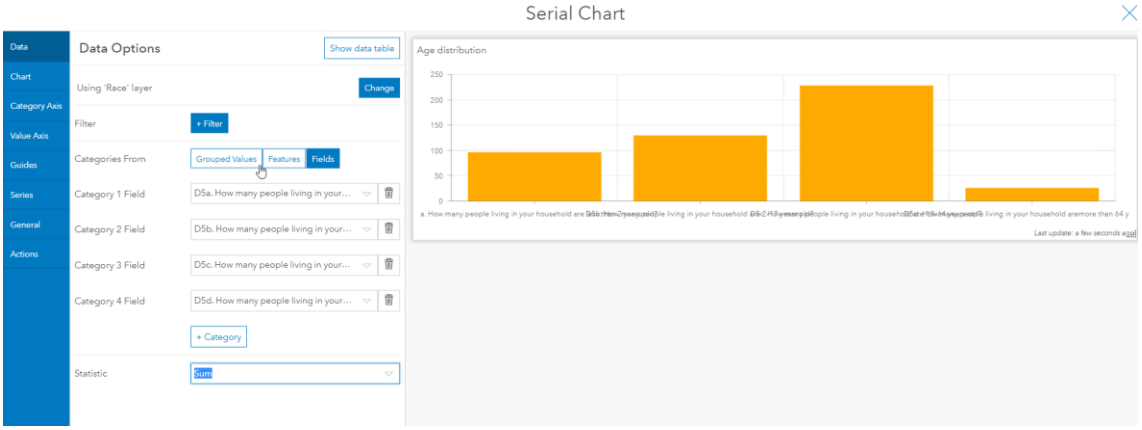


Click the Actions Tab and Filter the Map.

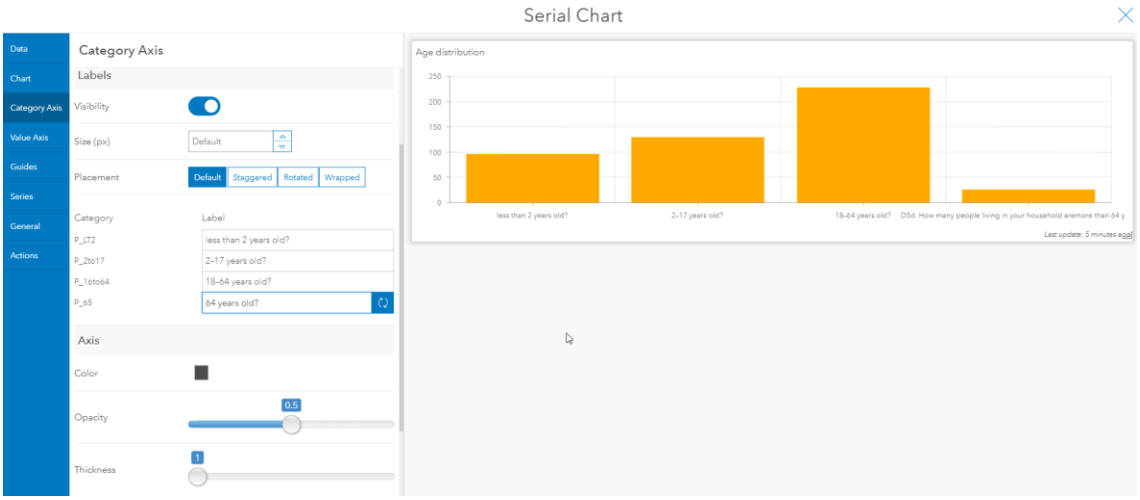


Click in the + symbol and add a Serial Chart. In the Categories Option Select Fields.

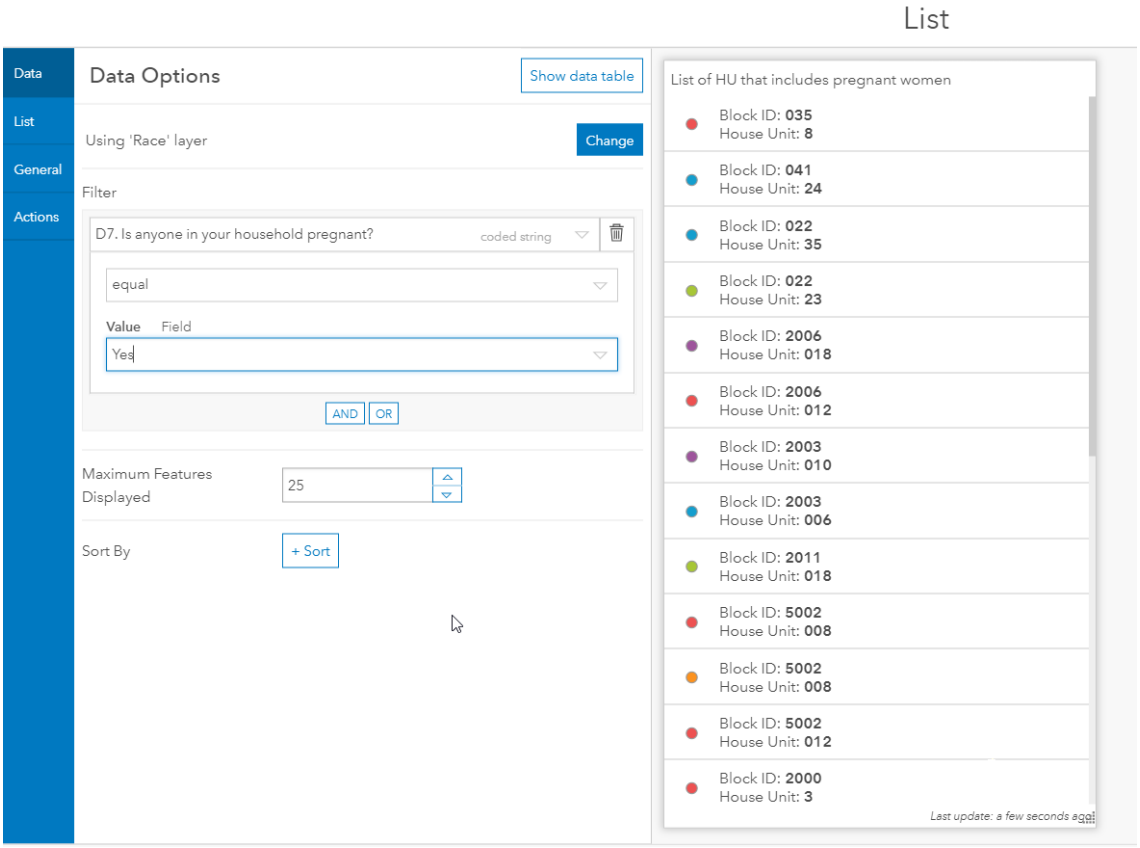
Select the D5a to D5d questions as independent Categories.



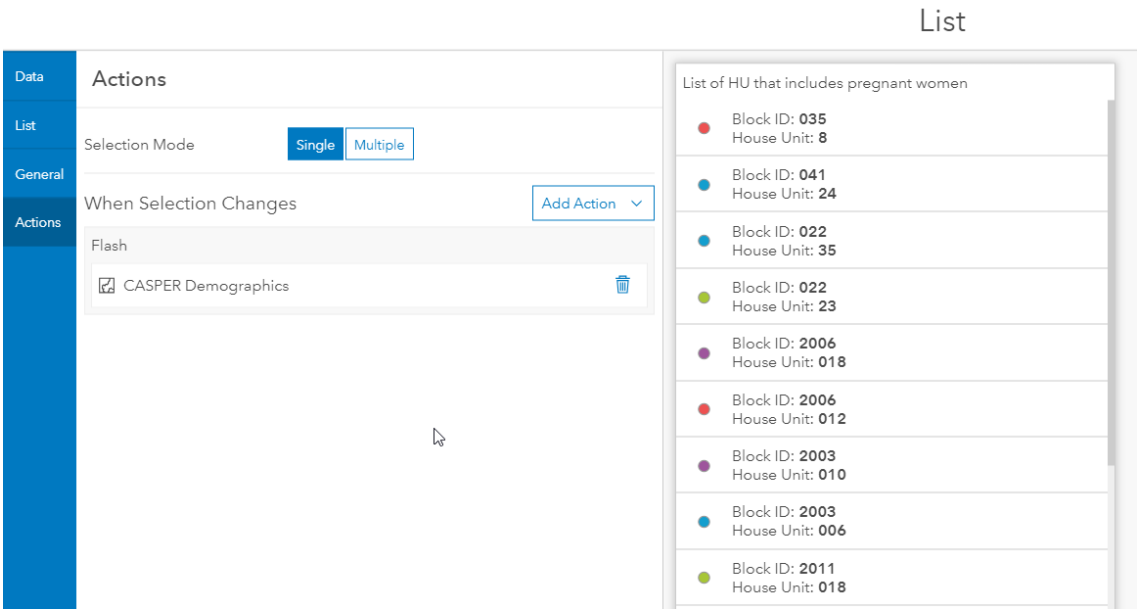
Click to the Category Axes tab and configure the Category Labels for the Chart.



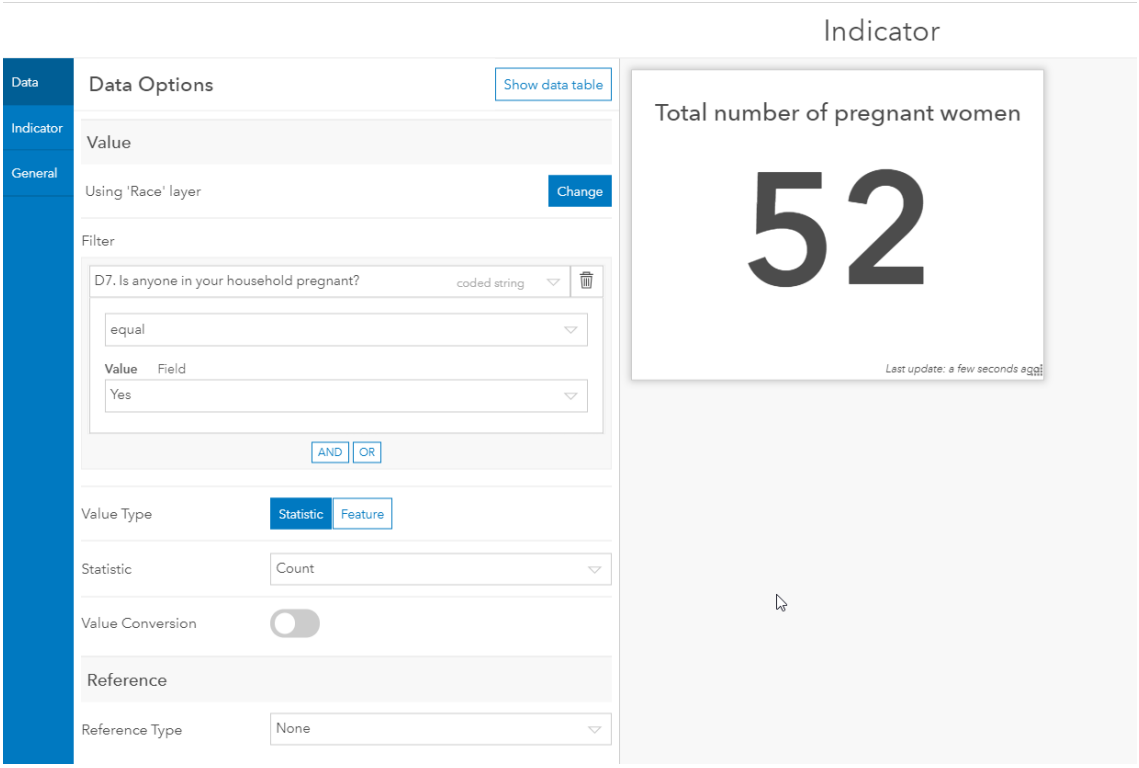
Click in the + symbol and add a List. Filter the Survey data by the D7. question = Yes.



Click the Action Tab and Filter the Map.



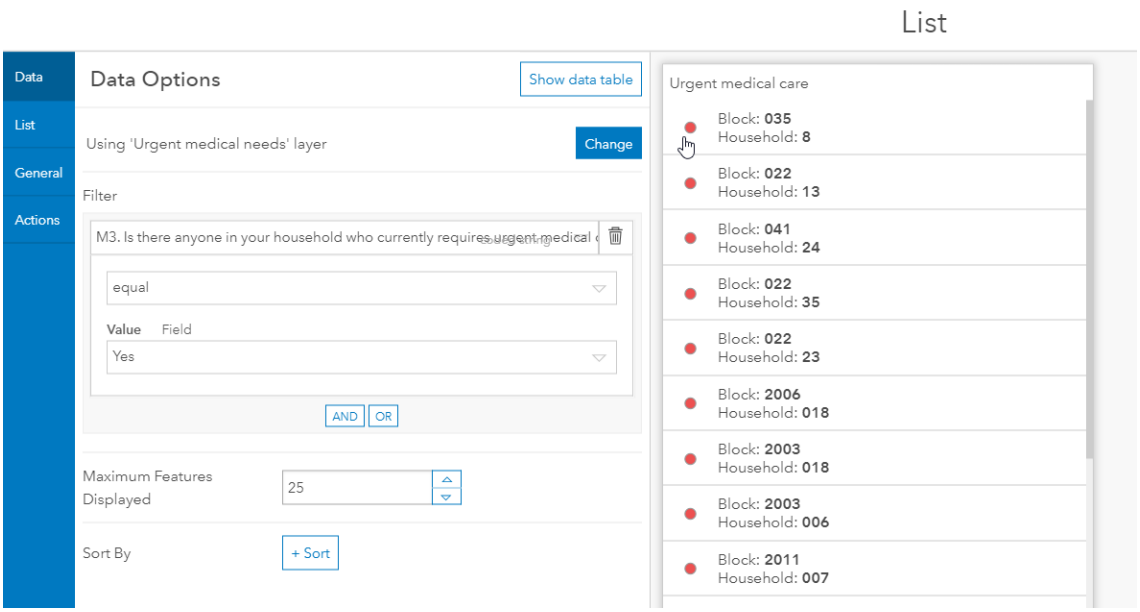
Click in the + symbol and add an Indicator. Filter the Survey data by the D7 question = Yes.



Configure the Medical Needs Dashboard

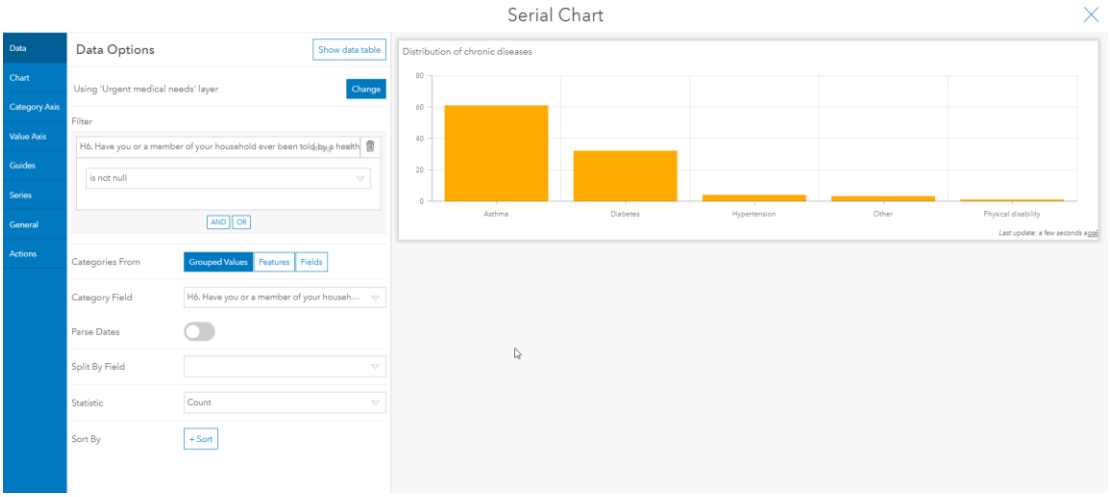
Follow the same Step to create a new Web Map and start configuring the Medical Needs dashboard.

Click in the + symbol and add a List. Filter the Survey Data by the M3. Question = Yes.

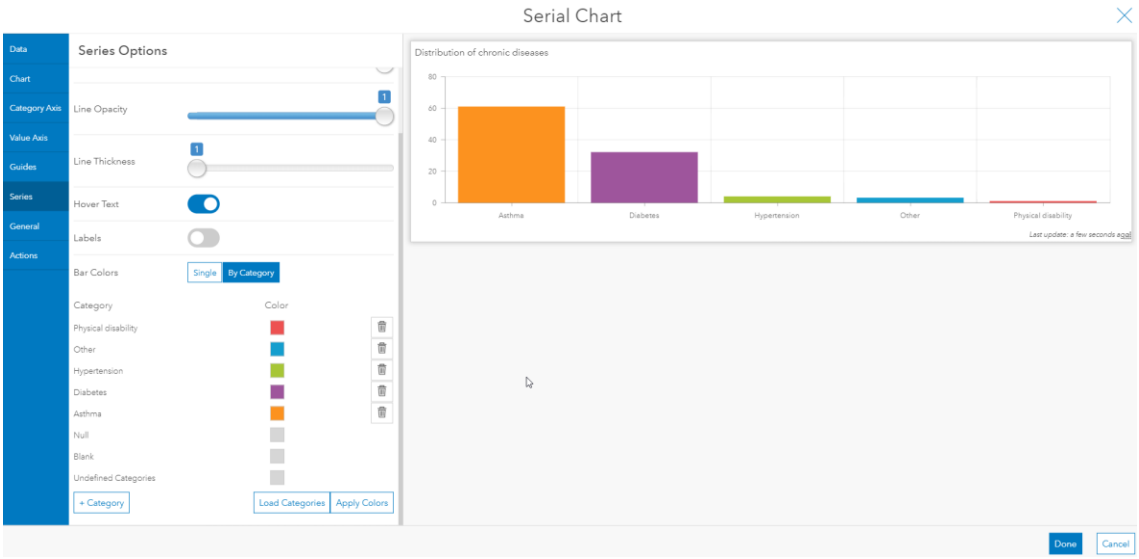


Click in the + symbol and add a Gauge. Filter the Survey Data by the M3. Question = Yes. Add 210 as the Maximum Value.

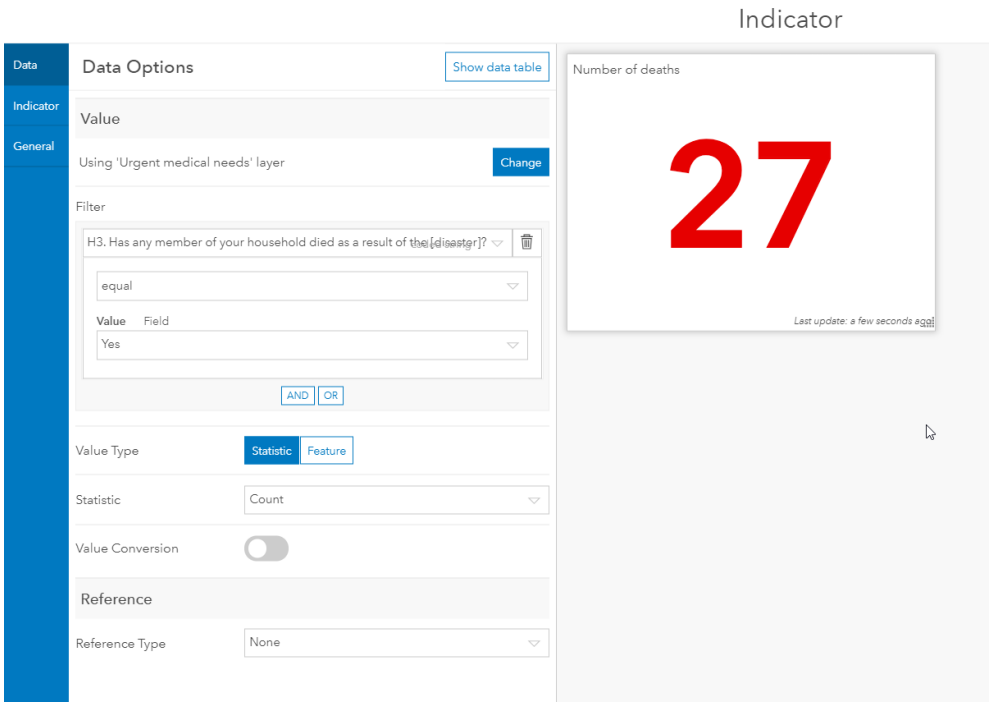
Click in the + symbol and add a Serial Chart. Filter the Survey Data by the H6. Question = Is not null



Click in the Series Tab and select Bar Colors by Categories. Click in Load Categories.



Click in the + symbol and add an Indicator. Filter the Data Source by the H3 Question = Yes



Click in the + symbol and add an Indicator. Filter the Data Source by the H5 Question = Yes