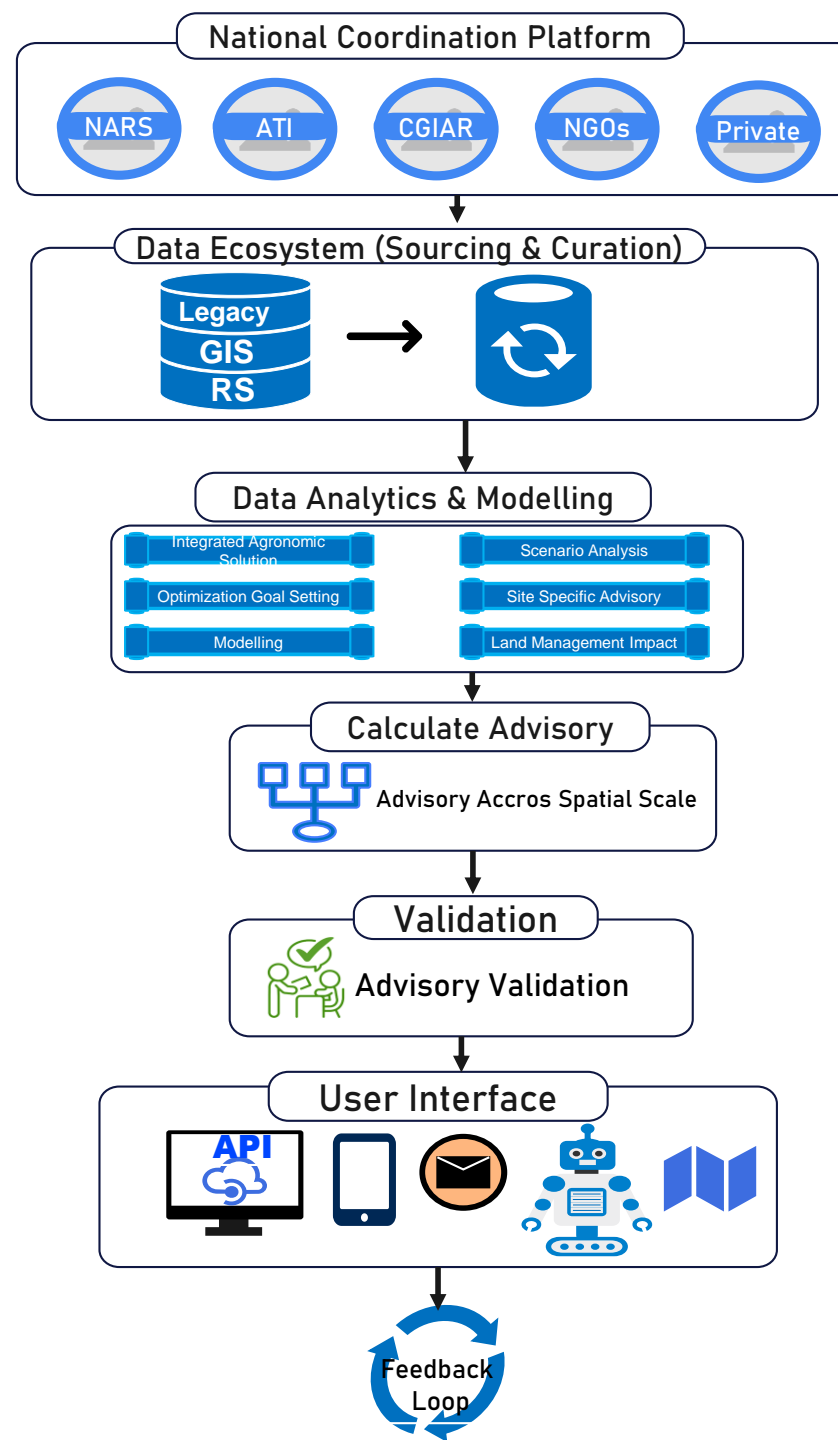


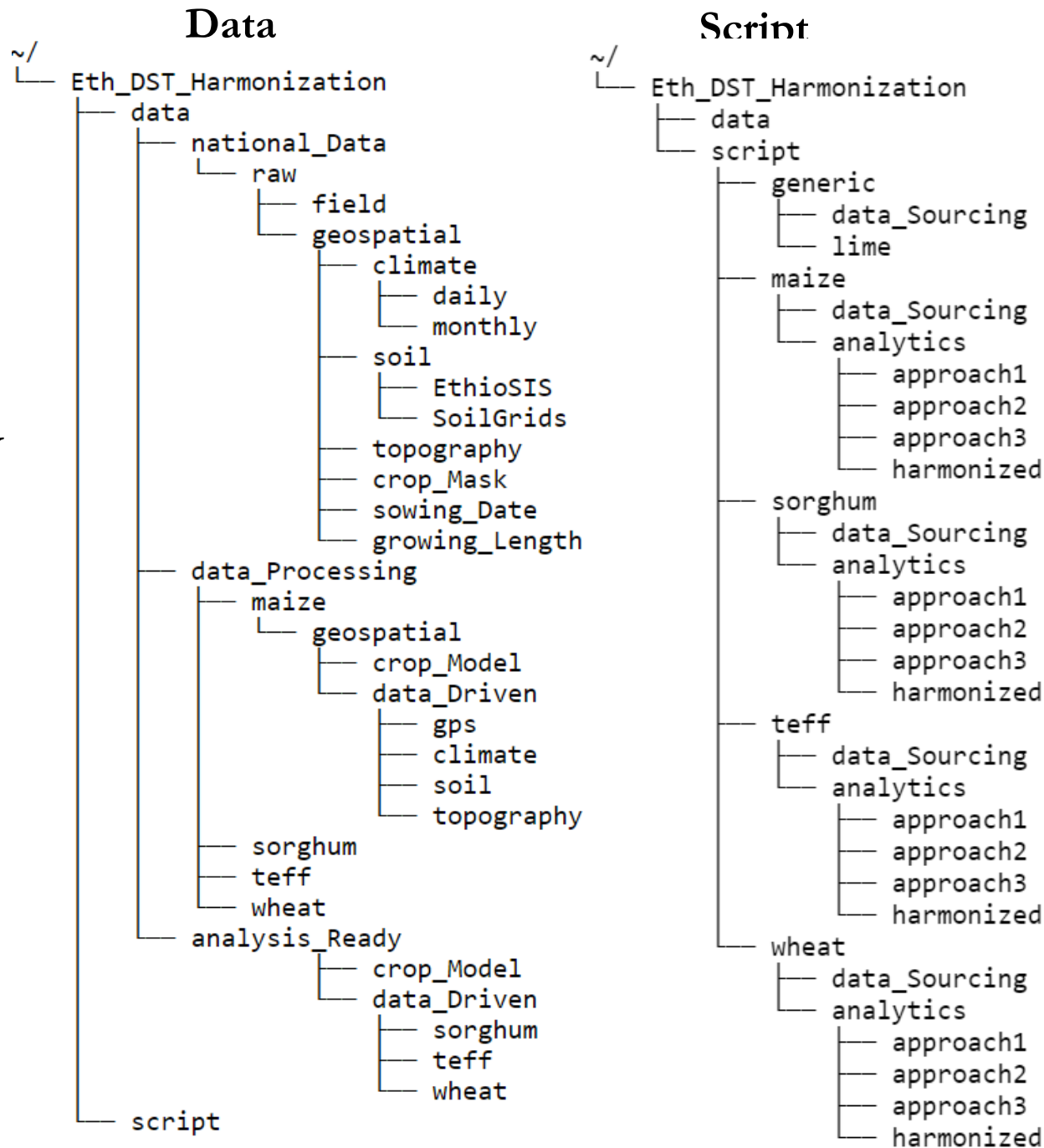
Eth. DST Harmonization Data Ecosystem Module

November 2023

1. Modules of DST Harmonization

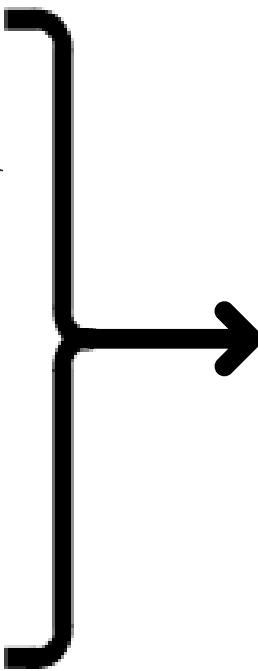


2. Directory Structure



As agreed with the DST Harmonization team, a directory structure for data and the script was created. The data directory contains the input geospatial and field trial data, the intermediate outputs and the final results that are ready for data analytics. The scripts directory contains all the R scripts for data sourcing and Analytics. [CGLabs Link](#)

3. Geospatial Data

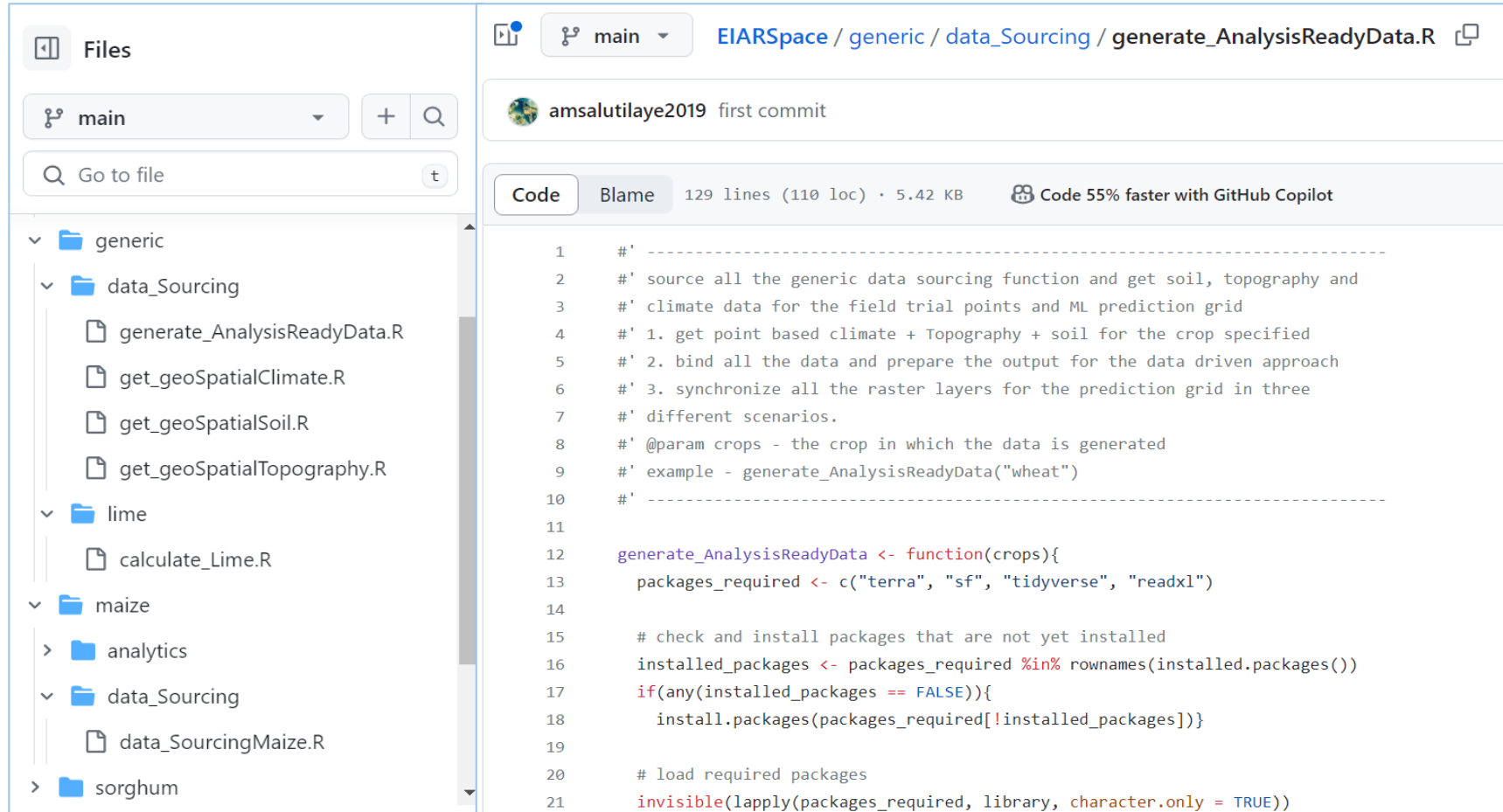
1. Soil (EthioSIS and Soil Grids (the 6 profiles and 0 - 20cm depth))
 2. Topography (SRTM) – 30m
 3. Rainfall (CHIRPS) ~ 5km
 4. Relative Humidity (AgEra5) ~ 10km
 5. Solar Radiation (AgEra5) ~ 10km
 6. Temperature min (AgEra5) ~ 10km
 7. Temperature max (AgEra5) ~ 10km
 8. Wind Speed (AgEra5) ~ 10km
- 

The geospatial covariates data consists the historical data (1981 – 2022). The data is sourced from CHIRPS and Agera5 with temporal resolution of both Daily & monthly data.

4. Scripts

```
script
├── generic
│   ├── data_Sourcing
│   │   ├── get_geoSpatialClimate.R
│   │   ├── get_geoSpatialSoil.R
│   │   ├── get_geoSpatialTopography.R
│   │   └── generate_AnalysisReadyData.R
│   └── lime
│       └── calculate_Lime.R
└── maize
    ├── data_Sourcing
    └── data_SourcingMaize.R
```

5. Github



The screenshot displays a GitHub repository interface. On the left, the 'Files' sidebar shows the repository structure: a 'generic' folder containing a 'data_Sourcing' subfolder with the file 'generate_AnalysisReadyData.R', and other folders like 'lime' and 'maize'. The main area shows the selected file 'generate_AnalysisReadyData.R' from the 'generic / data_Sourcing' path. The file is owned by 'amsalutilaye2019' and is the 'first commit'. The code is written in R and includes comments describing its purpose: to source generic data, get soil/topography/climate data, and generate analysis-ready data for a specified crop. The code defines a function 'generate_AnalysisReadyData' that takes 'crops' as an argument, checks for required packages, and returns the analysis-ready data.

```
1  #' -----
2  #' source all the generic data sourcing function and get soil, topography and
3  #' climate data for the field trial points and ML prediction grid
4  #' 1. get point based climate + Topography + soil for the crop specified
5  #' 2. bind all the data and prepare the output for the data driven approach
6  #' 3. synchronize all the raster layers for the prediction grid in three
7  #' different scenarios.
8  #' @param crops - the crop in which the data is generated
9  #' example - generate_AnalysisReadyData("wheat")
10 #' -----
11
12 generate_AnalysisReadyData <- function(crops){
13   packages_required <- c("terra", "sf", "tidyverse", "readxl")
14
15   # check and install packages that are not yet installed
16   installed_packages <- packages_required %in% rownames(installed.packages())
17   if(any(installed_packages == FALSE)){
18     install.packages(packages_required[!installed_packages])
19   }
20
21   # load required packages
22   invisible(lapply(packages_required, library, character.only = TRUE))
```



**Private github repo
created and owned by
EIAR**