Intercluster 4Ws cleaning script

04 June, 2023

Contents

Introduction

This document summarises how reporting indicators are calculated for each of the clusters. This is a document in progress.

Run the file `cleaning_script` first.

Cash

```
# Total beneficiaries for cash
# Cash is lacking an ongoing option
cash %>%
  filter(project_status == "Completed" &
           previously_assisted == "No") %>%
  {sum(.$beneficiaries, na.rm = TRUE)}
```

```
## [1] 550668
```

FSL

Food baskets are used for the monthly reached for Food, filtered from `fss_activity_do_not_write` and summed from the column `beneficiaries`.

For the cumulative, FSL uses the AVERAGE of beneficiaries at community level, for some reason FSL is using sub-district. `na.rm = TRUE` is applied, only taking into account months that have beneficiaries.

For disaggregations, the column `females` is the total number of females, so to determine the number of adult women, the number of girls needs to be subtracted from the `females` columns. This is the same for males.

```
# Total beneficiaries for food
# For monthly reached
fsl %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  filter(activity == "Food Baskets" & reported_to_other_sector == "No") %>%
  group_by(admin4pcode) %>%
  slice(which.max(beneficiaries)) %>%
  ungroup() %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "food_monthly")
```

Food Security

```
## # A tibble: 3 x 3
##   month beneficiaries type
##   <chr>         <dbl> <chr>
## 1 feb          206176 food_monthly
## 2 jan          285354 food_monthly
## 3 mar          256953 food_monthly
```

```
# Cumulative calculation for food beneficiaries
fsl %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  filter(activity == "Food Baskets" & reported_to_other_sector == "No") %>%
  group_by(admin4pcode) %>%
  slice(which.max(beneficiaries)) %>%
```

```
  ungroup() %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  summarise(beneficiaries = mean(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "food_cumulative")
```

```
## # A tibble: 1 x 2
##   beneficiaries type
##           <dbl> <chr>
## 1       249494. food_cumulative
```

```
# Monthly for livelihoods
# How does this work with ERL?
fsl %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  filter(activity %in% c("Agricultural inputs",
                         "Animal treatment/vaccination",
                         "Income-generating activities (IGAs)",
                         "Infrastructure rehabilitation",
                         "Livestock asset restoration") &
           reported_to_other_sector == "No") %>%
  group_by(admin4pcode) %>%
  slice(which.max(beneficiaries)) %>%
  ungroup() %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "livelihoods_monthly")
```

Livelihoods

```
## # A tibble: 3 x 3
##   month beneficiaries type
##   <chr>         <dbl> <chr>
## 1 feb           10066 livelihoods_monthly
## 2 jan           38707 livelihoods_monthly
## 3 mar          123006 livelihoods_monthly
```

```
# Cumulative beneficiaries for livelihoods
fsl %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  filter(activity %in% c("Agricultural inputs",
                         "Animal treatment/vaccination",
                         "Income-generating activities (IGAs)",
                         "Infrastructure rehabilitation",
                         "Livestock asset restoration") &
           reported_to_other_sector == "No") %>%
  group_by(admin4pcode) %>%
  slice(which.max(beneficiaries)) %>%
  ungroup() %>%
  group_by(month) %>%
```

```
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "livelihoods_monthly")
```

```
## # A tibble: 3 x 3
##   month beneficiaries type
##   <chr>         <dbl> <chr>
## 1 feb           10066 livelihoods_monthly
## 2 jan           38707 livelihoods_monthly
## 3 mar          123006 livelihoods_monthly
```

WASH

Monthly: Max of beneficiaries reached, grouped, in order by first health facility OR school; second, collective shelter; third, camp_code; and finally, at the community level.

Cumulative is the max of all months, for the same levels, starting with health facility OR school

Total reached is the `so2` column. `so1` is also calculated for comparison.

```
# Monthly reached
wash %>%
  filter(project_status == "Completed") %>%
  group_by(month,
           health_facility_name,
           school_name,
           collective_shelter_name,
           admin5pcode,
           admin4pcode) %>%
  slice(which.max(so2)) %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(so2, na.rm = TRUE)) %>%
  mutate(type = "wash_monthly")
```

```
## # A tibble: 3 x 3
##   month beneficiaries type
##   <chr>         <dbl> <chr>
## 1 FEB         1215361 wash_monthly
## 2 Jan          647842 wash_monthly
## 3 Mar         1271276 wash_monthly
```

```
# Cumulative
# The cumulative tallies
wash %>%
  filter(project_status == "Completed") %>%
  group_by(health_facility_name,
           school_name,
           collective_shelter_name,
           admin5pcode,
           admin4pcode) %>%
  slice(which.max(so2)) %>%
  ungroup() %>%
  summarise(beneficiaries = sum(so2, na.rm = TRUE)) %>%
  mutate(type = "wash_cumulative")
```

```
## # A tibble: 1 x 2
##   beneficiaries type
##           <dbl> <chr>
## 1       1941066 wash_cumulative
```

```
# so1 calculations
# These tally
wash %>%
  filter(project_status == "Completed") %>%
  group_by(health_facility_name,
           school_name,
           collective_shelter_name,
           camp_name,
           admin5pcode,
           admin4pcode) %>%
  slice(which.max(so1_1_1)) %>%
  ungroup() %>%
  summarise(beneficiaries = sum(so1_1_1, na.rm = TRUE)) %>%
  mutate(type = "wash_so1_1_1_cumulative")
```

```
## # A tibble: 1 x 2
##   beneficiaries type
##           <dbl> <chr>
## 1       1447203 wash_so1_1_1_cumulative
```

```
wash %>%
  filter(project_status == "Completed") %>%
  group_by(health_facility_name,
           school_name,
           collective_shelter_name,
           camp_name,
           admin5pcode,
           admin4pcode) %>%
  slice(which.max(so1_1_2)) %>%
  ungroup() %>%
  summarise(beneficiaries = sum(so1_1_2, na.rm = TRUE)) %>%
  mutate(type = "wash_so1_1_2_cumulative")
```

```
## # A tibble: 1 x 2
##   beneficiaries type
##           <dbl> <chr>
## 1        954587 wash_so1_1_2_cumulative
```

CCCM

CCCM reports two sets of figures:

- Assisted IDPs (only use CCCM assisted, ignore the ISIMM sheet)

- Tracked IDPs (this is not available through the 4Ws)

The cumulative number of beneficiaries is the maximum across all months, for the assisted IDPs on the sheet CCCM.

Raw data is not currently provided to OCHA.

```
cccm %>%
  filter(activity == "CCCM") %>%
  pivot_longer(cols = (jan:dec), names_to = "month", values_to = "beneficiaries") %>%
  group_by(admin4pcode) %>%
  slice(which.max(beneficiaries)) %>%
  ungroup() %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "cccm_cumulative")
```

```
## # A tibble: 1 x 2
##   beneficiaries type
##           <dbl> <chr>
## 1        155077 cccm_cumulative
```

```
cccm %>%
  filter(activity == "CCCM") %>%
  pivot_longer(cols = (jan:dec), names_to = "month", values_to = "beneficiaries") %>%
  group_by(admin4pcode, month) %>%
  slice(which.max(beneficiaries)) %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "cccm_monthly")
```

```
## # A tibble: 3 x 3
##   month beneficiaries type
##   <chr>         <dbl> <chr>
## ## 1 feb          62303 cccm_monthly
## ## 2 jan         100405 cccm_monthly
## ## 3 mar          26017 cccm_monthly
```

SNFI

For the calculation of unique beneficiaries for Shelter, use the `new beneficiaries` column, for any activity.

For NFIs, only `1.1.1.1: Provision of core NFIs` is considered for the calculations. For the cumulative figure, only count `new beneficiaries` reached by that activity.

SNFI disaggregations – girls, boys, men, women, elderly men, elderly women

```
# NFI reached
snfi %>%
  filter(sub_cluster == "nfi" & activity == "1.1.1.1: Provision of core NFIs" &
                 new_beneficiaries == "Yes") %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "nfi_monthly")
```

```
snfi %>%
  filter(sub_cluster == "shelter" & new_beneficiaries == "Yes") %>%
  mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
  group_by(month) %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "shelter_monthly")
```

Health

Health data is only available at admin3.

For the calculation of beneficiaries reached, only the activity `medical_procedures` is considered. The cluster reached (number of interventions) is also the number of `medical_procedures`. The cumulative is the sum of all months (frequencies), interventions as opposed to persons.

One additional indicator is calculated (sum), for comparison with Amman:

-x1_5_1_treatment_courses_delivered_to_health_facilities_drug_treatment_one_

For inclusion in the dashboard, these activities are also summed: x1_1_1_outpatients_cons x1_1_7_cases_referred_speciali x1_1_3_mental_health_cons_supported
x1_1_4_physical_rehab_sess_supported

Sex and age disaggregation can be ignored for Health, this cluster just backfills data using MSNA percentages.

```
health %>%
  mutate(month = case_when(implementation_month == 1 ~ "jan",
                           implementation_month == 2 ~ "feb",
                           TRUE ~ NA_character_)) %>%
  filter(!is.na(month)) %>%
  group_by(month) %>%
  filter(activity == "medical_procedures") %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "health_monthly")

health %>%
  filter(activity == "medical_procedures") %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "health_cumulative")
```

ERL

For the monthly reached, only direct beneficiaries not reported to other clusters For the the cumulative, only direct beneficiaries and not previously assisted and not reported to other clusters. ERL reports on both direct and indirect beneficiaries.

Disaggregations are backfilled so they can be ignored.

```
# Monthly reached
erl %>%
  mutate(month = str_sub(implementing_month, start = 5L, end = 7L),
         month = str_to_lower(month)) %>%
  filter(report_this_to_any_other_sector == "No") %>%
  group_by(month) %>%
```

```
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "erl_monthly")

# Cumulative
erl %>%
  mutate(month = str_sub(implementing_month, start = 5L, end = 7L)) %>%
  filter(report_this_to_any_other_sector == "No" & previously_assisted == "No") %>%
  # These steps are not necessary
  # group_by(admin4pcode) %>%
  # slice(which.max(beneficiaries)) %>%
  # ungroup() %>%
  summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
  mutate(type = "erl_cumulative")
```

Education

The monthly reached is of beneficiaries from all activities. The cumulative is the sum of all months.

However, the data calculation table provided in the 4Ws is wrong. Education has yet to provide how their beneficiaries are calculated from raw data.

Nutrition

To understand how cumulative and monthly are calculated, you have to refer to the formulas (measures) inside the pivot.

However, I don't understand how to calculate the max by admin4pcode for Nutrition, as in even following the specified formula inside the pivot, I am still unable to reproduce the figures reported by Nutrition.

Below is a list of calculated values used by Nutrition:

Total reached: = [Total CU5]+[Total PLWs]+[MNT Adolescents]

Total PLWs in the max of all PLWs at the community level. Max PLWs is the max of micronutrient PLWs, HEB PLWs, MAM PLWs, IYCF counselling PLWs, IYCF messaging, Mothers' support groups and FMA; however, when checking the totals, inside the sheet `Data` in the Nutrition submissions, many more activities have been included.

MNT adolescents is the number of girls who received micronutrients

Cumulative under 5 is the max of all under fives at the community level. Max U5 is the sum of max girls and max boys at admin4. The specific activities included in the calculation of both max girls or boys are LNS, U5 MNP, U5 HEB, U5 Vitamin A, U5 in-patient SAM, U5 out-patient, de-worming and MAM.

However, much like the calculations for PLWs, when reviewing which data actually comprises the columns `Total U5 boys` and `Total U5 girls`, we note that the range of activities included is much broader than specified

I still cannot replicate Nutrition's calculations.

Protection

Child protection   For child protection beneficiaries, filter in only rows under the AOR "Child Protection", the `analysis unit` "# people" and filter out "None" and "NAs" from the column `hrp indicator`.

```
prot %>%
  filter(ao_r == "Child Protection" &
           analysis_unit == "# people" &
           hrp_indicator %out% c("1.3.1", "None") &
           !is.na(hrp_indicator)) %>%
  summarise(beneficiaries = sum(total_reached, na.rm = TRUE)) %>%
  mutate(type = "child_protection_cumulative")

prot %>%
  filter(ao_r == "Child Protection" & analysis_unit == "# people") %>%
  summarise(interventions = sum(total_cumulative_interventions, na.rm = TRUE)) %>%
  mutate(type = "child_protection_interventions")
```

GBV   For the calculation of GBV beneficiaries, only filter only rows with the `code_activity` "GBV100" and filter out the "None" and "NAs" from the column `hrp indicator`, then sum the column `total reached`.

For GBV interventions, filter rows with the `code activity` "GBV200", "GBV300", "GBV500" and "GBV600", sum the column `total cumulative interventions`, irrespective of `hrp indicator` or `analysis unit`.

```
prot %>%
  filter(code_activity == "GBV100" &
           !is.na(hrp_indicator) &
           hrp_indicator %out% c("None") &
           analysis_unit == "# people") %>%
    summarise(beneficiaries = sum(total_reached, na.rm = TRUE)) %>%
  mutate(type = "GBV_cumulative")

prot %>%
  filter(code_activity %in% c("GBV200",
                              "GBV300",
                              "GBV500",
                              "GBV600")) %>%
  summarise(interventions = sum(total_cumulative_interventions, na.rm = TRUE)) %>%
  mutate(type = "GBV_interventions")
```

Mine Action   For mine action beneficiaries, filter only the rows under the `Humanitarian Mine Action` AOR and the `analyis unit` "# people", and sum the column `Total reached`.

For mine action interventions, filter in all HRP indicators except 1.3.1 and "none" and sum the column `total cumulative interventions`. The analysis unit should be `# people`.

```
prot %>%
  filter(ao_r = "Humanitarian Mine Action" &
           analysis_unit == "# people") %>%
  summarise(benefiaries = sum(total_reached, na.rm = TRUE)) %>%
  mutate(type = "mine_action_cumulative")

prot %>%
  filter(ao_r = "Humanitarian Mine Action" &
           analysis_unit == "# people" &
           !is.na(hrp_indicator) &
```

```
        hrp_indicator %out% c("1.3.1")) %>%
  summarise(benefiaries = sum(total_cumulative_interventions, na.rm = TRUE)) %>%
  mutate(type = "mine_action_interventions")
```

General protection The calculation for beneficiaries reached for general protection has not been provided.

For general protection interventions, filter in all HRP indicators except 1.3.1 and "none" and sum the column `total cumulative interventions`. The analysis unit should be `# people`. -People reached (data and calculation pending)

```
prot %>%
  filter(ao_r == "Protection" &
           !is.na(hrp_indicator) &
           hrp_indicator %out% c("1.3.1")) %>%
  summarise(interventions = sum(total_cumulative_interventions, na.rm = TRUE)) %>%
  mutate(type = "general_protection_interventions")
  count(ao_r)
```

Total protection Beneficiaries are the sum of beneficiaries in child protection, mine action, GBV and general protection. This cannot be calculated right now due to general protection not having provided how their beneficiaries are calculated.

-beneficiaries is the sum of beneficiaries in child protection, mine action, GBV and general protection

Monthly reached

This dataset is apparently regularly requested for sharing with OCHA Amman.

```
admin4 <- rbind(
  wash %>%
    mutate(month = str_to_lower(month)) %>%
    filter(project_status == "Completed") %>%
    group_by(month,
             health_facility_name,
             school_name,
             collective_shelter_name,
             admin5pcode,
             admin4pcode) %>%
    slice(which.max(so2)) %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(so2, na.rm = TRUE)) %>%
    ungroup() %>%
    mutate(cluster = "WASH"),

  cash %>%
    group_by(month = month(end_date, label = TRUE)) %>%
    filter(!is.na(month)) %>%
    mutate(month = str_to_lower(month)) %>%
    filter(project_status == "Completed") %>%
    mutate(month = ifelse(month == "nov", month(Sys.Date()), month)) %>%
    group_by(month, admin4pcode) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE),
```

```r
                      .groups = "drop") %>%
    ungroup() %>%
    mutate(cluster = "Cash"),

  cccm %>%
    filter(activity == "CCCM") %>%
    pivot_longer(cols = (jan:dec), names_to = "month", values_to = "beneficiaries") %>%
    group_by(admin4pcode, month) %>%
    slice(which.max(beneficiaries)) %>%
    ungroup() %>%
    select(admin4pcode, beneficiaries, month) %>%
    mutate(cluster = "CCCM"),

  # Monthly reached
  erl %>%
    mutate(month = str_sub(implementing_month, start = 5L, end = 7L),
           month = str_to_lower(month)) %>%
    filter(report_this_to_any_other_sector == "No") %>%
    group_by(month, admin4pcode) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
    mutate(cluster = "ERL"),

  fsl %>%
    mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
    filter(activity == "Food Baskets" & reported_to_other_sector == "No") %>%
    group_by(admin4pcode) %>%
    slice(which.max(beneficiaries)) %>%
    ungroup() %>%
    group_by(month, admin4pcode) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
    mutate(cluster = "Food"),

  fsl %>%
    mutate(month = case_when(reporting_month == "January" ~ "jan",
                             TRUE ~ NA_character_)) %>%
    filter(activity %in% c("Agricultural inputs",
                           "Animal treatment/vaccination",
                           "Income-generating activities (IGAs)",
                           "Infrastructure rehabilitation",
                           "Livestock asset restoration") &
             reported_to_other_sector == "No") %>%
    group_by(admin4pcode) %>%
    slice(which.max(beneficiaries)) %>%
    ungroup() %>%
    group_by(month, admin4pcode) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE)) %>%
    mutate(cluster = "Livelihoods"),

  nut %>%
    mutate(month = str_to_lower(str_sub(implementing_month, start = 1L, end = 3L))) %>%
    group_by(admin4pcode, month) %>%
    slice(which.max(beneficiaries)) %>%
    ungroup() %>%
```

```r
    select(admin4pcode, beneficiaries, month) %>%
    mutate(cluster = "Nutrition"),

  edu %>%
    mutate(month = str_to_lower(str_sub(reporting_month, start = 4L, end = 6L))) %>%
    group_by(admin4pcode, month) %>%
    slice(which.max(beneficiaries)) %>%
    ungroup() %>%
    select(admin4pcode, beneficiaries, month) %>%
    mutate(cluster = "Education"),

  prot %>%
    mutate(month = str_to_lower(str_sub(month, start = 6L, end = 8L))) %>%
    filter(ao_r == "Child Protection" &
             analysis_unit == "# people" &
             hrp_indicator %out% c("1.3.1", "None") &
             !is.na(hrp_indicator)) %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(total_reached, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "Child Protection"),

  prot %>%
    mutate(month = str_to_lower(str_sub(month, start = 6L, end = 8L))) %>%
    filter(code_activity == "GBV100" &
             !is.na(hrp_indicator) &
             hrp_indicator %out% c("None") &
             analysis_unit == "# people") %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(total_reached, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "GBV"),

  prot %>%
    mutate(month = str_to_lower(str_sub(month, start = 6L, end = 8L))) %>%
    filter(ao_r == "Humanitarian Mine Action" &
             analysis_unit == "# people") %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(total_reached, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "Mine Action"),

  prot %>%
    mutate(month = str_to_lower(str_sub(month, start = 6L, end = 8L))) %>%
    filter(ao_r == "Protection" &
             analysis_unit == "# people") %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(total_reached, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "General Protection"),

  snfi %>%
    filter(sub_cluster == "nfi" & activity == "1.1.1.1: Provision of core NFIs" &
```

```r
                    new_beneficiaries == "Yes") %>%
    mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "NFI"),

  snfi %>%
    filter(sub_cluster == "shelter" & new_beneficiaries == "Yes") %>%
    mutate(month = str_to_lower(str_sub(reporting_month, start = 1L, end = 3L))) %>%
    group_by(admin4pcode, month) %>%
    summarise(beneficiaries = sum(beneficiaries, na.rm = TRUE),
              .groups = "drop") %>%
    mutate(cluster = "Shelter")

)

admin4 %>%
  write_csv("./data/admin4_cluster_months_beneficiaries.csv")
```