deelcodewaardering

```
# Deelcode waardering op basis van DeelcodeWaardering Tool levering 20230907
# Beleid uit notitie: Onbezwaarde waarde, maatwerk BSQ en procedure van bedenkingen Versie 1.0 | 27 juni 2017

require(readr)
require(readxl)
require(dplyr)
require(tidyverse)

# Parameters
options(readr.show_progress = FALSE)
options(readxl.show_progress = FALSE)
options(dplyr.summarise.inform = FALSE)
options(pillar.max_dec_width = 20)

# Paths for Deelcodewaardering
WOZ_informatie_filename <- list.files(pathWOZDataExport, paste0(leveringDatum, paste0("_WOZ_informatie_versie_")), full.names = T) %>% sort(TRUE) %>% head(1)
deelcodes_filename <- list.files(pathWOZDataExport, paste0(leveringDatum, paste0("_WOZ deelcodes Tools_versie_")), full.names = T) %>% sort(TRUE) %>% head(1)
pathDeelcodewaarderingStatic <- r'{G:\OGA\G-PROG-ADVIEZEN\BEG\Knime\01. Grondprijstool\08. DeelcodeWaardering\01. Bronbestanden}'
# Buurtstraten_filename <- file.path(pathDeelcodewaarderingStatic, "Buurtstraten.xlsx")
buurtstraten_filename <- file.path(pathWOZDataExport, "BAG_TAG_buurtstraten_actueel_historisch.csv")
maatwerk_filename <- file.path(pathDeelcodewaarderingStatic, "Maatwerk invoer.xlsx")
maatwerkDeelcodewaardering_filename <- file.path(pathDeelcodewaarderingStatic, "Maatwerk deelcodewaardering.csv")
bronDeelcodes_filename <- file.path(pathDeelcodewaarderingStatic, "BronDeelcodes.csv")
objecten2020_filename <- list.files(pathBelastingDataleveringen, "BEL_DEEL_GENO_2020", full.names = T) %>% sort(TRUE) %>% head(1)
nieuwbouw_filename <- file.path(pathDeelcodewaarderingStatic, "Nieuwbouw buurten en buurtstraten.xlsx")
maatwerkDeelcodewaardering2018_filename <- file.path(pathDeelcodewaarderingStatic, "Maatwerk deelcodewaardering 2018.csv")
defaultwaarden_filename <- file.path(pathDeelcodewaarderingStatic, "20260325_Default_waardes.csv")

#functie die de vereiste bestanden in het globale geheugen laadt
DCW_data_laden <- function() {

    message("[Inlezen WOZ_informatie.csv]")
    # Hier worden de WOZobjectdata ingeladen. Alle kolommen die relevant zijn worden meegenomen
    WOZ_Informatie_01 <<- read_delim(WOZ_informatie_filename,
                              delim = ';',
                              locale = locale('nl', encoding = 'UTF-8', decimal_mark = ","),
                              na = c("", "99999999"),
                              col_types = cols_only(
                                ObjectId = col_number(),
                                GeoCode = col_character(),
                                GeoCodeNieuw = col_character(),
                                Hoofdcode = col_integer(),
                                `Code buurt nieuw`= col_character(),
                                `Code Buurt` = col_character(),
                                `Code Wijk nieuw` = col_character(),
                                `Code Wijk` = col_character(),
                                Straatnaam = col_character()
                              ))

    # Een lijst van maatwerk deelcodewaarderingen per waardecode in buurtstraten
    # die los van maatwerk deelcodewaardering zijn
    message("[Inlezen buurtstraten Maatwerk invoer.xlsx]")
    Maatwerk_invoer_buurtstraat_2014 <<- read_excel(maatwerk_filename,
                                                    sheet = "Buurtstraat")

    # Een lijst van maatwerk deelcodewaarderingen per waardecode in buurten
    # die los van maatwerk deelcodewaardering zijn
    message("[Inlezen buurten Maatwerk invoer.xlsx]")
    Maatwerk_invoer_buurt_2014 <<- read_excel(maatwerk_filename,
                                              sheet = "Buurt") %>%
      select(-Opmerking)


    # Een lijst van maatwerk deelcodewaarderingen per deelcodecode in buurtstraten
    message("[Inlezen Maatwerk deelcodewaardering.csv]")
    Maatwerk_deelcode_waardering_2014 <<- read_delim(maatwerkDeelcodewaardering_filename,
                                                     delim = ";",
                                                     locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                                                   col_types = cols_only(
                                                     Type = col_character(),
                                                     Buurtstraatcode = col_character(),
                                                     WaardeCode = col_number(),
                                                     Waarde = col_number(),
```

```
                                          `Bron waarde` = col_character(),
                                          WOZjaar = col_number()
                                        ), na = c("99999999", ""))

# WOZ deelcodes tools is per deelobject per jaar
# de waarde, het woningtype, het opppervlak en of het gereed en beschikt is
message("[Inlezen WOZ deelcodes Tools.csv]")
WOZdeelcodes_01 <- read_delim(deelcodes_filename,
                              delim = "{",
                              locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                              na = c("", "99999999"),
                              col_types = cols_only(
                                ObjectId = col_number(),
                                Deelcode = col_number(),
                                DeelcodeOmschrijving = col_character(),
                                WOZjaar = col_number(),
                                GO = col_number(),
                                WOZwaarde = col_number(),
                                Type = col_character(),
                                MAAT = col_character(),
                                PercgereedDeel = col_number(),
                                WOZWAARDETOTAAL = col_number(),
                                WaardeBeschikt = col_character(),
                                PERCGEREEDTOTAAL = col_number(),
                                WOZpermeter = col_number()
                              ))

# Een lijst met alle deelcodes en de omschrijvingen erbij
message("[Inlezen BronDeelcodes.csv]")
Bron_deelcodes_01 <- read_delim(bronDeelcodes_filename,
                                locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                                col_types = cols_only(
                                  Deelcode = col_number(),
                                  WaardeCode = col_number()
                                ),
                                delim = "{")

message("[Inlezen AMS_BEL_DEEL_GENO_2020.csv]")
TAX_01 <- read_delim(objecten2020_filename,
                                   locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                                   na = c('TAXBB', "", "          "),
                                   col_types = cols_only(
                                     OBJECTID = col_number(),
                                     W6115 = col_number(),
                                     BWJR = col_number()
                                   ))

# Een lijst met alle deelcodes en de omschrijvingen erbij
message("[Inlezen Buurten Nieuwbouw buurten en buurtstraten.xlsx]")
Nieuwbouw_buurten_01 <- read_excel(nieuwbouw_filename,
                                   sheet = "Buurten") %>%
  rename("Code Buurt" = Buurtcode)

# Een lijst met alle deelcodes en de omschrijvingen erbij
message("[Inlezen Buurtstraten Nieuwbouw buurten en buurtstraten.xlsx]")
Nieuwbouw_buurtstraten_01 <- read_excel(nieuwbouw_filename,
                                        sheet = "Buurtstraten") %>%
  rename("GeoCode" = Buurtstraatcode)

# Een lijst met alle deelcodes en de omschrijvingen erbij
# TODO: Vraag: Kan deze rij niet toegevoegd worden aan Maatwerk deelcode waardering?
message("[Inlezen Maatwerk deelcodewaardering 2018.xlsx]")
Maatwerk_deelcode_waardering_2018_01 <- read_delim(maatwerkDeelcodewaardering2018_filename,
                                                   locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                                                   col_types = cols_only(
                                                     Type = col_character(),
                                                     Buurtstraatcode = col_character(),
                                                     `Bron waarde` = col_character(),
                                                     WaardeCode = col_number(),
                                                     Waarde = col_number(),
                                                     `Aantal referenties` = col_number()
                                                   ),
                                                   delim = ",") %>%
  mutate(WOZjaar = 2018)

# Standaard waarderingen voor bepaalde objecten van belastingen.
message("[Inlezen Defaultwaarden]")
Default_waarden <- read_delim(defaultwaarden_filename,
                              locale = locale('nl', encoding = 'Windows-1252', decimal_mark = ","),
                              col_types = cols_only(
                                WaardeCode = col_number(),
                                `Default waarde` = col_number(),
                                jaar = col_number()
                              ),
```

```
                                ) %>% rename(WOZjaar = `jaar`)

    #Alle boten
    Boten_01 <<- WOZdeelcodes_01 %>% filter(Type == "BOOT") %>%
      inner_join(WOZ_Informatie_01 %>% select(
      ObjectId,
      GeoCode,
      GeoCodeNieuw,
      `Code Wijk`,
      `Code Buurt`,
      `Code buurt nieuw`,
      `Code Wijk nieuw`),
      by = "ObjectId")

    #Alle boten in de oude buurtindeling
    Boten_2014 <<- Boten_01 %>%
      group_by(GeoCode,
               Type,
               `Code Wijk`,
               `Code Buurt`) %>%
      summarise() %>% ungroup() %>% select(GeoCode, `Code Buurt`, `Code Wijk`, Type)

    #Alle boten in de nieuwe buurtindeling
    Boten_2021 <<- Boten_01 %>%
      group_by(GeoCodeNieuw,
               Type,
               `Code buurt nieuw`,
               `Code Wijk nieuw`) %>%
      summarise() %>% ungroup() %>% select(GeoCodeNieuw, `Code buurt nieuw`, `Code Wijk nieuw`, Type)


    BAG_TAG_buurtstraten_actueel_historisch <<- read_delim(buurtstraten_filename,
                                                           na = "")
    #WOZ_Tools, oude Buurtstraten en WOZdeelcodes
    #Nieuwe GeoCodes uit WOZ_informatie, oude buurtstraten uit Buurtstraten
    #Checken op BOOT in de WOZdeelcodes
    Buurtcodes_vertalen <<- WOZ_Informatie_01 %>% filter(!is.na(GeoCode)) %>%
      group_by(GeoCode, GeoCodeNieuw, `Code Buurt`, `Code Wijk`, `Code buurt nieuw`, `Code Wijk nieuw`) %>% summarise() %>%
      ungroup() %>% full_join(BAG_TAG_buurtstraten_actueel_historisch) %>% filter(Actueel == TRUE) %>% select(-c("Naam openbare ruimte", "Woonplaats", "Actuee


    Alle_Buurtstraten_2014 <<- Buurtcodes_vertalen %>% select(GeoCode, `Code Buurt`, `Code Wijk`) %>% unique() %>%
      mutate(Type = "EGWO/STWO") %>%
      separate_rows(sep = "/", Type) %>%
      bind_rows(Boten_2014)

    Alle_Buurtstraten_2021 <<- Buurtcodes_vertalen %>% select(GeoCodeNieuw, `Code buurt nieuw`, `Code Wijk nieuw`) %>%
      unique() %>% mutate(Type = "EGWO/STWO") %>%
      separate_rows(sep = "/", Type) %>%
      bind_rows(Boten_2021) %>% rename("Code Buurt" = `Code buurt nieuw`,
                                       "Code Wijk" = `Code Wijk nieuw`,
                                       "GeoCode" = GeoCodeNieuw)

    Alle_Buurtstraten <<- bind_rows(Alle_Buurtstraten_2014, Alle_Buurtstraten_2021)

    # Alle buurtstraten en Buurtcodes zonder de type uit Alle_buurtstraten_02
    Alle_Buurtstraten_2014 <<- Alle_Buurtstraten_2014 %>% group_by(GeoCode, `Code Buurt`) %>% summarise()

    Maatwerk_2014 <<- inner_join(Maatwerk_invoer_buurt_2014, Alle_Buurtstraten_2014, by = c("Buurtcode"= "Code Buurt"), relationship = "many-to-many") %>% sel
      bind_rows(Maatwerk_invoer_buurtstraat_2014 %>% rename("GeoCode" = Buurtstraatcode)) %>%
      mutate(`Bron waarde` = paste0("Maatwerk ", format(Datum, "%d-%m-%Y"))) %>% select(-c(Datum, Opmerking)) %>%
      full_join(Maatwerk_deelcode_waardering_2014 %>% rename("GeoCode" = Buurtstraatcode), by = c("Type", "GeoCode", "WaardeCode")) %>%
      mutate(`Bron waarde` = coalesce(`Bron waarde.x`, `Bron waarde.y`)) %>% select(-c(`Bron waarde.x`, `Bron waarde.y`)) %>%
      mutate(`Waarde` = coalesce(`Waarde.x`, `Waarde.y`)) %>% select(-c(`Waarde.x`, `Waarde.y`)) %>%
      mutate(WOZjaar = ifelse(is.na(WOZjaar),2014,WOZjaar)) %>% mutate(voorkeur = case_when(`Bron waarde` == "Maatwerk 01-08-2019" ~ 0,
                                                                                            `Bron waarde` == "Maatwerk 26-03-2019" ~ 0,
                                                                                            `Bron waarde` == "Maatwerk 09-04-2021" ~ 0,
                                                                                            `Bron waarde` == "Maatwerk 20-04-2020" ~ 0,
                                                                                            GeoCode %in% c("69644M34f", "69655M34f") ~ 0,
                                                                                            TRUE ~ 4)) %>% inner_join(Alle_Buurtstraten, by = c("Type", "GeoCo
      mutate("WOZpermeter"= Waarde, "Gemiddeld GO" = 1, "Correctiefactor oppervlakte" = 1)
}

# Functie die deelcodewaardering maakt. dwType is "regulier" of "nieuwbouw"
DCW_waarderen <- function(dwType){

  message("[Waarderingen berekenen voor ", dwType, "]")
  # Waardering uitvoeren op basis van een selectie aan WOZjaren
  WOZdeelcodes_Relevante_Jaren <<- WOZdeelcodes_01 %>% filter(WOZjaar %in%
    c(2014, 2017:2019, 2021:(as.double(format(Sys.Date(), "%Y")) - 2)) & WaardeBeschikt == "Ja") %>%
    filter(!is.na(Deelcode) & !is.na(WOZwaarde)) %>%
    filter((is.na(PERCGEREEDTOTAAL) | PERCGEREEDTOTAAL == 100) & (is.na(PercgereedDeel) | PercgereedDeel == 100))

  WOZ_en_deelcodes_01 <<- inner_join(WOZdeelcodes_Relevante_Jaren, WOZ_Informatie_01, by = "ObjectId")
```

```
WOZ_en_deelcodes_02 <<- inner_join(WOZ_en_deelcodes_01, Bron_deelcodes_01, by = "Deelcode")

#Nieuwbouw definieren
if(dwType == "nieuwbouw") {

  #Alleen de woningen van tijdens en na 2000 filteren
Nieuwbouw_woningen_01 <<- TAX_01 %>% filter(BWJR >= 2000 & W6115 == 1100) %>% group_by(OBJECTID) %>% summarise() %>% rename("ObjectId" = OBJECTID)
  # Alleen nieuwbouw
DeelcodeWaardering_Woningselectie <<- inner_join(WOZ_en_deelcodes_02, Nieuwbouw_woningen_01, by = "ObjectId") %>%
  filter(WOZwaarde > 0) %>%
  filter(!is.na(GeoCodeNieuw) & !(Hoofdcode %in% c("1332", "1333", "1384", "1385", "1386", "1211")) & Type %in% c("STWO", "EGWO", "BOOT"))
} else {
  # Alle bouw
DeelcodeWaardering_Woningselectie <<- WOZ_en_deelcodes_02 %>%
  filter(WOZwaarde > 0) %>%
  filter(!is.na(GeoCodeNieuw) & !(Hoofdcode %in% c("1332", "1333", "1384", "1385", "1386", "1211")) & Type %in% c("STWO", "EGWO", "BOOT"))
}

 DeelcodeWaardering_alle_Geocodes <<- DeelcodeWaardering_Woningselectie %>%
  mutate(GeoCode = case_when(WOZjaar %in% c(2014, 2017, 2018, 2019) ~ GeoCode,
                             WOZjaar %in% 2021:(as.double(format(Sys.Date(), "%Y")) - 2) ~ GeoCodeNieuw)) %>%
  mutate(`Code Buurt` = case_when(WOZjaar %in% c(2014, 2017, 2018, 2019) ~ `Code Buurt`,
                                  WOZjaar %in% 2021:(as.double(format(Sys.Date(), "%Y")) - 2) ~ `Code buurt nieuw`)) %>%
  mutate(`Code Wijk` = case_when(WOZjaar %in% c(2014, 2017, 2018, 2019) ~ `Code Wijk`,
                                  WOZjaar %in% 2021:(as.double(format(Sys.Date(), "%Y")) - 2) ~ `Code Wijk nieuw`)) %>%
  select(-c(`GeoCodeNieuw`, `Code Wijk nieuw`, `Code buurt nieuw`))

# Kleine en grote grondoppervlakken eruit filteren
DeelcodeWaardering_Woningen <<- DeelcodeWaardering_alle_Geocodes %>% filter(Deelcode == 1100) %>%
  filter(GO >= 20 & GO <= 500)

# Tuinen markeren en grootste en kleinste kwantielen filteren
DeelcodeWaardering_Tuin_en_grond <<- DeelcodeWaardering_alle_Geocodes %>% filter(!Deelcode == 1100) %>%
  filter(WaardeCode == 2010 | WaardeCode == 2030) %>%
  group_by(Type,  GeoCode, WOZjaar) %>%
  mutate(`0.1-quantile(GO)` = quantile(GO, probs = .1, type = 6),
           `0.9-quantile(GO)` = quantile(GO, probs = .9, type = 6)) %>%
  ungroup() %>%
  filter(GO >= `0.1-quantile(GO)` & GO <= `0.9-quantile(GO)`)

# Alles wat geen Woning of Tuin is
DeelcodeWaardering_Rest <<- DeelcodeWaardering_alle_Geocodes %>% filter(!Deelcode == 1100) %>%
  filter(!(WaardeCode == 2010 | WaardeCode == 2030))

# Correctiefactor berekenen voor Woningen 1100
DeelcodeWaardering_correctiefactor <<- bind_rows(DeelcodeWaardering_Woningen, DeelcodeWaardering_Tuin_en_grond, DeelcodeWaardering_Rest) %>%
  mutate(`Correctiefactor oppervlakte` = case_when(Type == "EGWO" & Deelcode == 1100 ~ (GO^-0.218)*2.0246,
                                                   Type == "STWO" & Deelcode == 1100 ~ (GO^-0.176)*1.7652,
                                                   TRUE ~ 1)) %>%
  mutate(`Waarde` = case_when(Type == "EGWO" & Deelcode == 1100 ~ WOZwaarde / GO / `Correctiefactor oppervlakte`,
                              Type == "STWO" & Deelcode == 1100 ~ WOZwaarde / GO / `Correctiefactor oppervlakte`,
                              TRUE ~ WOZwaarde / GO)) %>%
  filter(!(Deelcode == 1100 & Waarde < 1000))

# Groeperen op buurtstraat, buurt en wijk niveau
DeelcodeWaardering_groepen <<- DeelcodeWaardering_correctiefactor %>%
  group_by(WOZjaar, Type, GeoCode, WaardeCode) %>%
  mutate(Count_Buurtstraat = n()) %>% ungroup() %>%
  group_by(WOZjaar, Type, `Code Buurt`, WaardeCode) %>%
  mutate(Count_Buurt = n()) %>% ungroup() %>%
  group_by(WOZjaar, Type, `Code Wijk`, WaardeCode) %>%
  mutate(Count_Codewijk = n()) %>% ungroup() %>% filter(WaardeCode != 1600)

DeelcodeWaardering_groepen_buurt <<- DeelcodeWaardering_groepen %>% filter(!(Type == "STWO" & GeoCode == "65536N71h" & WOZjaar == 2014))

#Aanhorigheden filteren voor later gebruik
DeelcodeWaardering_Aanhorigheden <<- DeelcodeWaardering_groepen_buurt %>% filter(WaardeCode %in% c("1340", "1350", "1490", "1500", "1590", "1900"))

# WaterVerdedigingsWerken
if(dwType == "regulier") {
  WaterVerdedigingsWerken_buurt <<- DeelcodeWaardering_groepen %>% filter(Deelcode == "2030") %>%
    filter(Count_Codewijk >= 5) %>%
    group_by(WOZjaar, `Code Buurt`, Type, WaardeCode, Count_Buurt) %>%
    summarise(`Waarde` = max(Waarde),
              `Gemiddeld GO` = mean(GO),
              `WOZpermeter` = max(WOZpermeter),
              `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>%
    inner_join(Alle_Buurtstraten, by = c("Code Buurt", "Type"), relationship = "many-to-many") %>%
    mutate(voorkeur = 1, `Bron waarde` = "Buurt") %>% rename("Aantal referenties" = Count_Buurt)

  WaterVerdedigingsWerken_wijk <<- DeelcodeWaardering_groepen %>% filter(Deelcode == "2030") %>%
    filter(Count_Codewijk >= 5) %>%
    group_by(WOZjaar, `Code Wijk`, Type, WaardeCode, Count_Codewijk) %>%
    summarise(`Waarde` = max(Waarde),
```

```
              `Gemiddeld GO` = mean(GO),
              `WOZpermeter` = max(WOZpermeter),
              `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>%
    inner_join(Alle_Buurtstraten, by = c("Code Wijk", "Type"), relationship = "many-to-many") %>%
    mutate(voorkeur = 2, `Bron waarde` = "Wijk") %>% rename("Aantal referenties" = Count_Codewijk)

  WaterVerdedigingsWerken <<- bind_rows(WaterVerdedigingsWerken_buurt, WaterVerdedigingsWerken_wijk) %>%
    slice_min(voorkeur, n = 1) %>% mutate(WaardeCode = 2090,
                                          Waarde = (Waarde * -1),
                                          WOZpermeter = (WOZpermeter * -1))
} else {
  WaterVerdedigingsWerken <<- tibble()
}

DeelcodeWaardering_geen_aanhorigheden <<- anti_join(DeelcodeWaardering_groepen_buurt, DeelcodeWaardering_Aanhorigheden, by = c("ObjectId", "Deelcode")) %>%

DeelcodeWaardering_Buurtstraat <<- DeelcodeWaardering_groepen %>%
  filter(Count_Buurtstraat >= 5) %>%
  group_by(WOZjaar, Type, GeoCode, WaardeCode, Count_Buurtstraat) %>%
  summarise(`Waarde` = mean(Waarde),
         `Gemiddeld GO` = mean(GO),
         `WOZpermeter` = mean(WOZpermeter),
         `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>%
  inner_join(Alle_Buurtstraten, by = c("Type", "GeoCode")) %>%
  mutate(`Bron waarde` = "Buurtstraat",
         `voorkeur` = 1) %>%
  rename("Aantal referenties" = Count_Buurtstraat)

DeelcodeWaardering_Wijk <<- DeelcodeWaardering_groepen_buurt %>%
  filter(Count_Codewijk >= 5) %>%
  group_by(WOZjaar, Type, `Code Wijk`, WaardeCode, Count_Codewijk) %>%
  summarise(`Waarde` = mean(Waarde),
            `Gemiddeld GO` = mean(GO),
            `WOZpermeter` = mean(WOZpermeter),
            `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>%
  inner_join(Alle_Buurtstraten,  by = c("Type", "Code Wijk"), na_matches = "never", relationship = "many-to-many") %>%
  mutate(`Bron waarde` = "Wijk",
         `voorkeur` = 3) %>%
  rename("Aantal referenties" = Count_Codewijk)

DeelcodeWaardering_Buurt_01 <<- DeelcodeWaardering_Aanhorigheden %>%
  filter(Count_Buurt >= 1) %>%
  group_by(WOZjaar, Type, `Code Buurt`, WaardeCode, Count_Buurt) %>%
  summarise(`Waarde` = mean(Waarde),
            `Gemiddeld GO` = mean(GO),
            `WOZpermeter` = mean(WOZpermeter),
            `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>% ungroup()

DeelcodeWaardering_Buurt_02 <<- DeelcodeWaardering_geen_aanhorigheden %>%
  filter(Count_Buurt >= 5) %>%
  group_by(WOZjaar, Type, `Code Buurt`, WaardeCode, Count_Buurt) %>%
  summarise(`Waarde` = mean(Waarde),
            `Gemiddeld GO` = mean(GO),
            `WOZpermeter` = mean(WOZpermeter),
            `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`)) %>% ungroup()

#Buurtniveau
DeelcodeWaardering_Buurt <<- bind_rows(DeelcodeWaardering_Buurt_01,
                                       DeelcodeWaardering_Buurt_02) %>%
  inner_join(Alle_Buurtstraten, by = c("Type", "Code Buurt"), relationship = "many-to-many") %>%
  mutate(`Bron waarde` = "Buurt",
         `voorkeur` = 2) %>%
  rename("Aantal referenties" = Count_Buurt)

#Uitzondering 2
Buurtstraat_andere_buurt <<- DeelcodeWaardering_Buurt %>% filter(Type == "STWO" & `Code Buurt` == "N71c" & WOZjaar == 2014) %>%
  mutate(GeoCode = "24732N60a, 69701N60a") %>% separate_rows(GeoCode) %>% mutate(`Bron waarde` = paste0("Buurt ", `Code Buurt`)) %>%
  group_by(Type, WOZjaar, WaardeCode, `Aantal referenties`, Waarde, WOZpermeter, `Bron waarde`, GeoCode) %>%
  summarise() %>% mutate(voorkeur = 0) %>% mutate("Gemiddeld GO" = 1, "Correctiefactor oppervlakte" = 1)

#Uitzondering 3
Buurtstraat_andere_buurtstraat <<- DeelcodeWaardering_Buurt %>% filter(Type == "EGWO" & GeoCode == "02755K90a" & WOZjaar == 2014) %>%
  mutate(GeoCode = "61581K23d") %>% mutate(`Bron waarde` = paste0("Buurtstraat ", "02755K90a")) %>%
  group_by(Type, WOZjaar, WaardeCode, `Aantal referenties`, Waarde, WOZpermeter, `Bron waarde`, GeoCode) %>%
  summarise() %>% mutate(voorkeur = 0) %>% mutate(`Code Buurt` = "K23d") %>% mutate("Gemiddeld GO" = 1, "Correctiefactor oppervlakte" = 1) %>%
  rows_delete(tibble(WaardeCode = 2030), by = "WaardeCode")

# Verschil in Knime
if(dwType == "regulier") {
  DeelcodeWaardering_alle_combinaties <<- bind_rows(DeelcodeWaardering_Buurt,
                                    DeelcodeWaardering_Buurtstraat,
                                    DeelcodeWaardering_Wijk)
} else {
  DeelcodeWaardering_alle_combinaties <<- bind_rows(DeelcodeWaardering_Buurt,
```

```
                                    DeelcodeWaardering_Buurtstraat,
                                    DeelcodeWaardering_Wijk,
                                    Buurtstraat_andere_buurt,
                                    Buurtstraat_andere_buurtstraat,
                                    Maatwerk_2014)
}

# Iedere waardering heeft een waardering gekregen, de beste worden gekozen.
DeelcodeWaardering_selectie <<- DeelcodeWaardering_alle_combinaties %>% group_by(Type,
                                  GeoCode,
                                  WaardeCode,
                                  WOZjaar) %>%
  slice_min(voorkeur, n = 1)

Buurt_andere_buurt <<- DeelcodeWaardering_selectie %>% filter(Type == "STWO" & ((`Code Buurt` %in% c("N71e", "N71f", "N71g") & WOZjaar == 2014) |
                                                                                 (`Code Buurt` == "N71g" & WOZjaar == 2018))) %>%
  select("WOZjaar", "Type", "WaardeCode", "Code Buurt", "GeoCode") %>% mutate("Buurtwaarde" = "N71c") %>%
  inner_join(DeelcodeWaardering_Buurt_02, by = c("WOZjaar", "Type", "WaardeCode", "Buurtwaarde" = "Code Buurt")) %>% rename("Aantal referenties" = Count_Buu
  select(-Buurtwaarde) %>% mutate(voorkeur = 0, "Code Wijk" = "N71", "Bron waarde" = "Buurt")

DeelcodeWaardering_selectie <<- bind_rows(DeelcodeWaardering_selectie, Buurt_andere_buurt) %>% group_by(Type,
                                                                                                        GeoCode,
                                                                                                        WaardeCode,
                                                                                                        WOZjaar) %>%
  slice_min(voorkeur, n = 1)

DeelcodeWaardering_met_waterverdedigingswerken <<- bind_rows(DeelcodeWaardering_selectie, WaterVerdedigingsWerken) %>%
  inner_join(Bron_deelcodes_01, by = "WaardeCode", relationship = "many-to-many")

DeelcodeWaardering_dakterassen_01 <<- DeelcodeWaardering_met_waterverdedigingswerken %>% filter(Deelcode == 1340) %>%
  select(-Deelcode) %>% pivot_wider(id_cols = c("GeoCode", "WOZjaar"),
                                    names_from  = Type,
                                    values_from = WOZpermeter,
                                    values_fn = last) %>%
  mutate(Waarde = case_when(is.na(EGWO) ~ STWO,
                            is.na(STWO) ~ EGWO,
                            TRUE ~ NA_real_)) %>%
  select(-c(EGWO, STWO)) %>% filter(!is.na(Waarde))

#Voor alle geocodes en types aanmerken of 1340 bestaat
DeelcodeWaardering_dakterassen_02 <<- DeelcodeWaardering_met_waterverdedigingswerken %>%
  pivot_wider(id_cols = c(Type, GeoCode, WOZjaar),
              names_from = WaardeCode,
              values_from = Waarde,
              values_fn = first) %>% ungroup() %>%
  mutate(across(-c(GeoCode, Type, WOZjaar), ~ifelse(is.na(.), 1, 0))) %>%
  select(`GeoCode`, `Type`,  `WOZjaar`, `1340`) %>%
  pivot_wider(id_cols = c(GeoCode, WOZjaar),
               names_from = Type,
               values_from = `1340`) %>%
  filter(!is.na(STWO), !is.na(EGWO))

#Er zijn geen boten in de nieuwbouwdata, alleen filteren in regulier
if(dwType == "regulier") {
  DeelcodeWaardering_dakterassen_02 <<- DeelcodeWaardering_dakterassen_02 %>% select(-any_of("BOOT"))
}

DeelcodeWaardering_alle_dakterassen <<- inner_join(DeelcodeWaardering_dakterassen_01, DeelcodeWaardering_dakterassen_02, by = c("GeoCode", "WOZjaar")) %>%
  mutate(EGWO = EGWO * Waarde,
         STWO = STWO * Waarde,
         WaardeCode = 1340) %>% select(-Waarde) %>%
  pivot_longer(cols = c(EGWO, STWO),
               values_to = "Waarde",
               names_to = "Type") %>%
  filter(Waarde != 0)

DeelcodeWaardering_deelcode_aanwezig <<- DeelcodeWaardering_met_waterverdedigingswerken %>%
  pivot_wider(id_cols = c(Type, GeoCode, WOZjaar),
              names_from = WaardeCode,
              values_from = Waarde,
              values_fn = first) %>% ungroup() %>%
  mutate(across(-c(GeoCode, Type, WOZjaar), ~ifelse(is.na(.), 1, 0))) %>%
  select(any_of(c("GeoCode", "Type",  "WOZjaar", "1350", "1490", "1590", "1600", "1840", "1900")))  %>%
  pivot_longer(cols = !c(Type, GeoCode, WOZjaar),
               names_to = "WaardeCode",
               names_transform = list("WaardeCode" = as.integer),
               values_to = "Aanwezig") %>%
  inner_join(Default_waarden, by = c("WOZjaar", "WaardeCode")) %>% rename("Waarde" = `Default waarde`) %>%
filter(Aanwezig == 1) %>% select(-Aanwezig)

DeelcodeWaardering_default <<- bind_rows(DeelcodeWaardering_alle_dakterassen,
                                  DeelcodeWaardering_deelcode_aanwezig) %>%
  left_join(Bron_deelcodes_01 %>% select(Deelcode, WaardeCode), by = "WaardeCode", relationship = "many-to-many") %>%
  mutate("Bron waarde" = "Default") %>% inner_join(Alle_Buurtstraten, by = c("GeoCode", "Type"), relationship = "many-to-many") %>%
```

```
    mutate("Gemiddeld GO" = 1,
           "Correctiefactor oppervlakte" = 1,
           "WOZpermeter" = Waarde)

  DeelcodeWaardering_gewaardeerd <<- bind_rows(DeelcodeWaardering_default, DeelcodeWaardering_met_waterverdedigingswerken)

  if(dwType == "nieuwbouw") {

    #Nieuwbouw bewaren om te koppelen aan regulier.
    Alle_nieuwbouw <<- DeelcodeWaardering_gewaardeerd %>% select("Type", "GeoCode", "WOZjaar", "Deelcode", "WaardeCode", "Aantal referenties", "Bron waarde")

    Nieuwbouw_buurten_02 <<- Buurtcodes_vertalen %>% group_by(`Code Buurt`, `Code buurt nieuw`) %>%
      summarise() %>% inner_join(Nieuwbouw_buurten_01, relationship = "many-to-many", by = "Code Buurt") %>% ungroup()

    Nieuwbouw_buurten_03 <<- bind_rows(Nieuwbouw_buurten_02 %>% select(`Code Buurt`, Gebruikscode),
                                       Nieuwbouw_buurten_02 %>% select(`Code buurt nieuw`, Gebruikscode) %>%
                                          rename("Code Buurt" = `Code buurt nieuw`)) %>% distinct()

    Nieuwbouw_buurtstraten_02 <<- Buurtcodes_vertalen %>% inner_join(Nieuwbouw_buurtstraten_01, by = "GeoCode")

    Nieuwbouw_buurtstraten_03 <<- bind_rows(
        Nieuwbouw_buurtstraten_02 %>% select(Gebruikscode, GeoCode),
        Nieuwbouw_buurtstraten_02 %>% select(Gebruikscode, GeoCodeNieuw) %>%
          rename("GeoCode" = `GeoCodeNieuw`))

    Nieuwbouw_geen_maatwerk <<- DeelcodeWaardering_gewaardeerd %>% filter(!grepl("Maatwerk", `Bron waarde`))
    Alleen_maatwerk <<- DeelcodeWaardering_gewaardeerd %>% filter(grepl("Maatwerk", `Bron waarde`))

    DeelcodeWaardering_nieuwbouw_buurten <<- Nieuwbouw_geen_maatwerk %>% inner_join(Nieuwbouw_buurten_03, by = c("Code Buurt", "Type" = "Gebruikscode"))

    DeelcodeWaardering_nieuwbouw_overig <<- anti_join(Nieuwbouw_geen_maatwerk, DeelcodeWaardering_nieuwbouw_buurten,
                                                      by = c("GeoCode", "WOZjaar", "WaardeCode", "Type", "Waarde", "Deelcode", "Bron waarde"))

    DeelcodeWaardering_nieuwbouw_buurtstraten <<- DeelcodeWaardering_nieuwbouw_overig %>% inner_join(Nieuwbouw_buurtstraten_03, by = c("GeoCode", "Type" = "Ge

    Nieuwbouw_Deelcodewaardering <<- bind_rows(DeelcodeWaardering_nieuwbouw_buurtstraten, DeelcodeWaardering_nieuwbouw_buurten, Alleen_maatwerk)
  }

  # Nieuwbouw GeoCodes filteren uit de reguliere deelcodewaardering, vervolgens de correcte nieuwbouw toevoegen.
  DeelcodeWaardering_zonder_nieuwbouw <<- DeelcodeWaardering_gewaardeerd %>% anti_join(Nieuwbouw_Deelcodewaardering, by = c("Type", "GeoCode", "WOZjaar", "Waa
    bind_rows(Nieuwbouw_Deelcodewaardering) %>% select(-`Code Buurt`) %>% select(-`Code Wijk`) %>% select(-voorkeur) %>% arrange(GeoCode, Deelcode, WaardeCode

  DeelcodeWaardering_met_nieuwbouw <<- DeelcodeWaardering_zonder_nieuwbouw %>%
    left_join(Alle_nieuwbouw, by = c("Type", "GeoCode", "WOZjaar", "Deelcode", "WaardeCode"), relationship = "many-to-many") %>%
    mutate("Alleen nieuwbouw" = case_when(`Aantal referenties.x` == `Aantal referenties.y` &
                                          `Bron waarde.x` == `Bron waarde.y` &
                                          !is.na(`Aantal referenties.x`) ~ "ja",
                                        TRUE ~ "nee")) %>%
    select(-c(`Bron waarde.y`, `Aantal referenties.y`)) %>%
    rename("Aantal referenties" = `Aantal referenties.x`,
           "Bron waarde" = `Bron waarde.x`) %>%
    mutate(WaardeOud = Waarde,
           Waarde = case_when(Waarde - floor(Waarde) == 0.5 ~ floor(Waarde) + 1,
                              TRUE ~ round(Waarde)),
           WOZpermeter = case_when(WOZpermeter - floor(WOZpermeter) == 0.5 ~ floor(WOZpermeter) + 1,
                                   TRUE ~ round(WOZpermeter))
    )
}

DCW_megafile_maken <- function() {
  message("[Megafile samenstellen]")
  Straatcodes <<- read_delim(r'{G:\OGA\G-PROG-ADVIEZEN\BEG\Team Data\data\WOZ\input\20201110\20201110Straatcodes.csv}',
                             locale = locale('nl', encoding = 'windows-1252', decimal_mark = ","),
                             col_types = cols_only(
                               Straatnaam = col_character(),
                               Straatcode = col_character()
                             ))
  Straatcodes_handmatig <<- read_delim(deliverPathToLatestFile(pathToStraatcodes, file_name_pattern = "Straatcodes handmatig_versie_"),
                                       col_types = cols_only(
                                         Straatnaam = col_character(),
                                         Straatcode = col_character()
                                       ))
  Straatcodes <<- bind_rows(Straatcodes, Straatcodes_handmatig)

  DeelcodeWaardering_met_nieuwbouw <<- bind_rows(
    DeelcodeWaardering_met_nieuwbouw %>% filter(!Type == "BOOT"),
    DeelcodeWaardering_met_nieuwbouw %>% filter(Type == "BOOT" & WaardeCode %in% c(2250, 1170)) %>%
      mutate(Type = "EGWO/STWO") %>% separate_rows(Type))

  DeelcodeWaardering_berging <<- DeelcodeWaardering_met_nieuwbouw %>% group_by(GeoCode, WOZjaar, Type) %>% summarise() %>%
    mutate(WaardeCode = 1600, `Bron waarde` = "Default") %>% inner_join(Default_waarden) %>% rename("Waarde" = `Default waarde`)

  DeelcodeWaardering_met_nieuwbouw <<- bind_rows(DeelcodeWaardering_met_nieuwbouw,
                                                 DeelcodeWaardering_berging)
```

```
  megafile_01 <<- DeelcodeWaardering_met_nieuwbouw %>% group_by(
    Type, GeoCode, WaardeCode, Waarde, WOZjaar) %>%
    summarise(Deelcode = first(Deelcode),
              `Bron waarde` = first(`Bron waarde`),
              `Aantal referenties meegenomen` = first(`Aantal referenties`),
              Waarde = mean(Waarde),
              WOZpermeter = mean(WOZpermeter),
              `Gemiddeld GO` = mean(`Gemiddeld GO`),
              `Correctiefactor oppervlakte` = mean(`Correctiefactor oppervlakte`),
              `Alleen nieuwbouw` = first(`Alleen nieuwbouw`)) %>%
    mutate(Straatcode = str_sub(GeoCode, 0, 5)) %>%
  left_join(Straatcodes, by = "Straatcode")  %>%
    mutate(Hulp = paste0(GeoCode, as.character(WaardeCode))) %>% ungroup() %>%
    select(Type, WaardeCode, Waarde, WOZjaar, `Bron waarde`, WOZpermeter,
           `Aantal referenties meegenomen`, Hulp, GeoCode, WaardeCode, Straatcode, Straatnaam,
           `Correctiefactor oppervlakte`) %>%
    arrange(Hulp, Type, WaardeCode, WOZjaar)

  if(write_output){
    write_delim(delim = ";",
                megafile_01,
                file.path(pathWOZDataExport,
                          paste0(leveringDatum, "_megafile_onbezwaarde_waardetool_versie_",
                                 format(Sys.Date(), "%Y%m%d"), ".csv")))
  }

  message("[Megafile klaar!]")
}

DCW_data_laden()
DCW_waarderen("nieuwbouw")
DCW_waarderen("regulier")
DCW_megafile_maken()
```