



AALBORG UNIVERSITY  
DENMARK

EARTH OBSERVATION  
**Sandholt**

# Towards Development of an Optimal Irrigation Management System Using Satellite and Meteorological Data in Denmark



Msc thesis

by

Georgios Karamitilios

Copenhagen 2019

## Abstract

Climate change occurs, changes in Earth's climate system result in new weather patterns that last for at least a few decades. Most of Northern Europe was affected by drought during the summer of 2018. This accentuates the need for irrigation and improved tools for irrigation management, that can be more effective, knowing when, where and how much the irrigation should be. The thesis contains a study on the creation of an irrigation support system. The system takes into account parameters about the soil moisture, evapotranspiration (ET) and precipitation. Evapotranspiration is calculated by the type of crop and weather parameters. All the weather data used in this project was downloaded from Danish Meteorology Institute (DMI). Soil moisture information is coming from remotely sensed information, while the type of soil is implemented in the decision support system. The study area is Viborg municipality, which is located in central north Denmark in an area with agricultural production. The test time period of the project was from March till July of 2018. In the final stage, the model created presents the amount of water that farmers need to irrigate in specific areas in their fields. The test fields chosen were in Viborg municipality and had potato, carrot and corn crops.

This project is an example of how geoinformatics and remote sensing products can offer the possibility of better management of water resources and reduced environmental impact of agricultural production. The output can benefit Danish farmers, and the export potential to other regions of the World is large. The decision support system created can provide information about water needs. The users can have information about crop water requirements to optimize irrigation.

## Contents

Abstract.....	i
Table of Figures.....	iv
List of Tables.....	vi
Abbreviations.....	vii
Acknowledgements.....	viii
1. Introduction.....	1
1.1. Background.....	4
1.1.1. Evapotranspiration.....	4
1.1.1.1. Units.....	5
1.1.1.2. Factors affecting Evapotranspiration.....	5
1.1.1.3. Determining Evapotranspiration.....	5
1.1.1.3.1. Evapotranspiration measurements.....	5
1.1.1.3.2. ET computed from meteorological data.....	8
1.1.1.3.3. Pan Evaporation.....	8
1.1.1.4. Estimating Evapotranspiration.....	8
1.1.1.4.1. Estimating Evapotranspiration using remotely sensed data and energy balance.....	8
1.1.1.4.2. Estimating Evapotranspiration from meteorological data.....	9
1.1.2. Soil.....	15
1.1.2.1. Soil moisture content.....	15
1.1.2.1.1. Saturation.....	16
1.1.2.1.2. Field capacity.....	16
1.1.2.1.3. Permanent wilting point.....	17
1.1.2.2. Available water content.....	17
1.1.3. Water needs.....	19
1.1.3.1. Crop water needs.....	19
1.1.3.2. Irrigation requirements.....	22
1.1.3.3. Effective Rainfall.....	23
1.2. Problem statement.....	24
1.3. Research questions.....	24
2. Study Area.....	25
3. Methodology.....	26
3.1. Stage 1 – Literature Review.....	26

3.2.	Stage 2 – Finding the Formula .....	30
3.3.	Stage 3 – Data Collection and Process .....	30
3.3.1.	Data Collection and Process of Soil Moisture .....	31
3.3.2.	Data Collection and Process for ET .....	36
3.3.3.	Data Collection and Process for rainfall .....	41
3.4.	Stage 4 – Model creation .....	44
3.5.	Stage 5 – Process Automation .....	49
4.	Results .....	53
5.	Discussion.....	60
6.	Conclusion.....	62
7.	References .....	63
8.	Appendix .....	66

## Table of Figures

FIGURE 1: AMOUNT OF PRECIPITATION IN DENMARK FROM APRIL TILL AUGUST, FROM 2011 – 2018 (SOURCE: (DMI (DANISH METEOROLOGY INSTITUTE) 2019)).....	1
FIGURE 2: DROUGHT INDEX FOR DENMARK FROM 2005 – 2018 (SOURCE: (DMI (DANISH METEOROLOGY INSTITUTE) 2019)).....	2
FIGURE 3: EVOLUTION OF LAND USER IRRIGATED AND RAINFED CROPPING (1961 – 2008) (FAO (FOOD AND AGRICULTURE ORGANIZATION) 2014). ....	3
FIGURE 4: THE PARTITIONING OF THE EVAPOTRANSPIRATION INTO EVAPORATION AND TRANSPIRATION OVER THE GROWING PERIOD FOR AN ANNUAL FIELD CROP (RICHARD G. ALLEN ET AL. 1998). ....	5
FIGURE 5: SOIL WATER BALANCE OF THE ROOT ZONE (RICHARD G. ALLEN ET AL. 1998). ....	7
FIGURE 6: LYSIMETERS (BME ABÉT 2008). ....	7
FIGURE 7: EVAPORATION PAN (BROUWER AND HEIBLOEM 1986). ....	8
FIGURE 8: FACTORS AFFECTING EVAPOTRANSPIRATION (RICHARD G. ALLEN ET AL. 1998). ....	10
FIGURE 9: THE CALCULATION OF $ET_0$ , $ET_c$ AND $ET_{c\text{ Adj}}$ (BROUWER AND HEIBLOEM 1986). ....	11
FIGURE 10: TYPICAL $K_c$ FOR DIFFERENT TYPES OF FULL-GROWN CROPS (RICHARD G. ALLEN ET AL. 1998). ....	13
FIGURE 11: THE SINGLE AND DUAL CROP COEFFICIENT VALUES FOR CROP (RICHARD G. ALLEN ET AL. 1998). ....	14
FIGURE 12: THE COMPOSITION OF THE SOIL (BROUWER, GOFFEAU, AND HEIBLOEM 1985). ....	15
FIGURE 13: A SOIL MOISTURE CONTENT OF 150 MM/M (BROUWER, GOFFEAU, AND HEIBLOEM 1985). ....	16
FIGURE 14: SOME SOIL MOISTURE CHARACTERISTICS (BROUWER, GOFFEAU, AND HEIBLOEM 1985). ....	17
FIGURE 15: SATURATION, FIELD CAPACITY, WILTING POINT AND AVAILABLE WATER CONTENT (BROUWER, GOFFEAU, AND HEIBLOEM 1985). ....	18
FIGURE 16: FIELD CAPACITY, WILTING POINT, AVAILABLE AND UNAVAILABLE WATER VS. SOIL TEXTURE (SPRUCEIRRIGATION.COM 2019). ....	19
FIGURE 17: IRRIGATION WATER NEED (BROUWER AND HEIBLOEM 1986). ....	22
FIGURE 18: DENMARK'S LOCATION. ....	25
FIGURE 19: WORKFLOW DIAGRAM. ....	26
FIGURE 20: AN OVERVIEW OF THE VARIABLES, WHICH CAN BE USED TO ESTIMATE THE IRRIGATION REQUIREMENTS. ....	27
FIGURE 21: AN EXAMPLE OF SOIL MOISTURE SATELLITE IMAGE. ....	29
FIGURE 22: EXAMPLE OF SMOS DATA. ....	32
FIGURE 23: GRAPH FOR THE SUBSET OF AN IMAGE. ....	32
FIGURE 24: THE .XML FILE TO SUBSET THE IMAGE. ....	33
FIGURE 25: COMMANDS TO RUN THE SUBSET .XML FILE. ....	33
FIGURE 26: THE SOIL TYPES IN EUROPE (PANAGOS 2006), (LIEDEKERKE, JONES, AND PANAGOS 2006). ....	34
FIGURE 27: MAP OF DENMARK'S SOIL TYPES. ....	35
FIGURE 28: DMI WEATHER DATA ARCHIVE WEBSITE (DMI (DANISH METEOROLOGY INSTITUTE) 2019). ....	37
FIGURE 29: FORMAT AND STRUCTURE OF DMI WEATHER DATA. ....	38
FIGURE 30: UML DIAGRAM OF THE WEATHER DATA ARCHIVE DATABASE. ....	38
FIGURE 31: SHAPEFILE OF FIELDS IN DENMARK IN 2018 (MILJØ, OG, AND FØDEVAREMINISTERIET 2019). ....	40
FIGURE 32: SHAPEFILE PRESENTING MARKS IN VIBORG (MILJØ, OG, AND FØDEVAREMINISTERIET 2019). ....	41
FIGURE 33: SQL QUERY TO RETRIEVE WEATHER DATA MEASUREMENTS FROM THE DATABASE. ....	42
FIGURE 34: SLOPE MAP OF VIBORG. ....	44
FIGURE 35: THE FINAL MODEL USING ARC MAP MODEL BUILDER. ....	45
FIGURE 36: 1 <sup>ST</sup> PART OF THE MODEL. ....	46
FIGURE 37: 2 <sup>ND</sup> PART OF THE MODEL. ....	46
FIGURE 38: FUNCTION FOR POPULATE $ET_c$ FIELD. ....	46
FIGURE 39: 3 <sup>RD</sup> PART OF THE MODEL. ....	47
FIGURE 40: 4 <sup>TH</sup> PART OF THE MODEL. ....	47

FIGURE 41: SET NULL FUNCTIONS FOR SOIL TYPE AND MOISTURE. ....	48
FIGURE 42: 5 <sup>TH</sup> PART OF THE MODEL. ....	48
FIGURE 43: CALCULATIONS OF THE FINAL RASTER FILES. ....	49
FIGURE 44: FOLDER CONTAINING SMOS DATA. ....	49
FIGURE 45: PYTHON CODE WITH ARGUMENTS. ....	50
FIGURE 46: PYTHON CODE FOR THE TRANSFORMATION OF OTHER FILES. ....	50
FIGURE 47: JAVA CODE PRESENTING IF FUNCTION. ....	51
FIGURE 48: JAVA CODE PRESENTING THE FOR LOOP. ....	51
FIGURE 49: THE SCRIPT.PROPERTIES FILE. ....	51
FIGURE 50: WEEKLY IRRIGATION NEEDS ON 08/04/2018. ....	54
FIGURE 51: DAILY IRRIGATION NEEDS ON 08/04/2018. ....	55
FIGURE 52: WEEKLY IRRIGATION NEEDS ON 15/07/2018. ....	56
FIGURE 53: DAILY IRRIGATION NEEDS ON 15/07/2018. ....	57
FIGURE 54: WEEKLY IRRIGATION NEEDS ON 25/07/2018. ....	58
FIGURE 55: DAILY IRRIGATION NEEDS ON 25/07/2018. ....	59

## List of Tables

TABLE 1: AVERAGE DAILY WATER NEED OF STANDARD GRASS DURING THE IRRIGATION SEASON (BROUWER AND HEIBLOEM 1986).....	20
TABLE 2: CROP WATER NEEDS IN THE PEAK PERIOD OF VARIOUS FIELD CROPS AS COMPARED TO STANDARD GRASS (BROUWER AND HEIBLOEM 1986) .....	20
TABLE 3: APPROXIMATE VALUES OF SEASONAL CROP WATER NEEDS BY FAO (BROUWER AND HEIBLOEM 1986).....	21
TABLE 4: INDICATIVE VALUES OF THE TOTAL GROWING PERIOD (BROUWER AND HEIBLOEM 1986). ....	22
TABLE 5: AVERAGE DAILY WATER NEED OF STANDARD GRASS DURING THE IRRIGATION SEASON (BROUWER AND HEIBLOEM 1986).....	27
TABLE 6: CROP WATER NEEDS IN THE PEAK PERIOD OF VARIOUS FIELD CROPS AS COMPARED TO STANDARD GRASS (BROUWER AND HEIBLOEM 1986). ....	28
TABLE 7: FIELD CAPACITY, WILTING POINT VS. SOIL TEXTURE (SPRUCEIRRIGATION.COM 2019). ....	29
TABLE 8: SOIL TYPE CATEGORIES IN DENMARK ACCORDING TO WORLD REFERENCE BASE FOR SOIL RESOURCES (WRB). ....	36
TABLE 9: FINAL WEATHER DATA FILE FOR VIBORG MUNICIPALITY INCLUDING THE ET OF GRASS AND EFFECTIVE RAINFALL. ....	43
TABLE 10: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 08/04/2018. ....	54
TABLE 11: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 08/04/2018. ....	55
TABLE 12: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 15/07/2018. ....	56
TABLE 13: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 15/07/2018. ....	57
TABLE 14: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 25/07/2018. ....	58
TABLE 15: AMOUNT OF WATER THAT FARMERS SHOULD IRRIGATE ON 25/07/2018. ....	59

## Abbreviations

<b>AR</b>	Arenosol
<b>CCI</b>	Climate Change Initiative
<b>CM</b>	Cambisol
<b>DEM</b>	Digital Elevation Model
<b>DMI</b>	Danish Metrology Institute
<b>E</b>	Evaporation
<b>ECMWF</b>	European Center for Medium-range Weather Forecasts
<b>ESA</b>	European Space Agency
<b>ESDAC</b>	European Soil Data Centre
<b>ET</b>	Evapotranspiration
<b>FAO</b>	Food and Agriculture Organization
<b>FL</b>	Fluvisol
<b>GL</b>	Gleysol
<b>HS</b>	Histosol
<b>LAI</b>	Leaf Area Index
<b>LV</b>	Luvisol
<b>NASA</b>	National Aeronautics and Space Administration
<b>NDVI</b>	Normalized difference vegetation index
<b>PZ</b>	Podzol
<b>RG</b>	Regosol
<b>SMAP</b>	Soil Moisture Active Passive
<b>SMOS</b>	Soil Moisture Ocean Salinity
<b>SQL</b>	Structured Query Language
<b>SRTM</b>	Shuttle Radar Topography Mission
<b>T</b>	Transpiration
<b>TSEB</b>	Two-Source Energy Balance
<b>UML</b>	Unified Modeling Language
<b>UTM</b>	Universal Transverse Mercator
<b>WGS</b>	World Geodetic System
<b>WRB</b>	World Reference Base for Soil Resources



## Acknowledgements

This thesis is written by Karamitilios Georgios, for the Master of Science (MSc) in Technology, Surveying, Planning and Land Management, Cand.tech in Geoinformatics, Aalborg University, Copenhagen. The thesis was undertaken from 01 February to 06 June 2019 as a collaboration with Sandholt ApS, Copenhagen.

I would like to express my appreciation to my supervisor, Associate Professor Jamal Jokar Arsanjani. I would like to express my deep gratitude to Dr. Inge Sandholt for her patient guidance, and useful critiques of my work at Sandholt ApS. I am particularly grateful for the assistance given by Kenan Vilic and for his enthusiastic encouragement. I would like to offer my special thanks to the people supported me in Copenhagen (Marina, Branko, Thanasis and Olga) and not in Copenhagen (Nikiforos, Olga, Irina, Konstantinos). Finally, I wish to thank my parents for their support throughout my study and for my choices.

## 1. Introduction

Denmark is a country with a large amount of exports of agricultural products, producing three times more the amount of food, which is needed for the domestic population (Michael Falch (Danmarks Statistik) 2010). The type of soil and the moderate climate in the country is helpful to agriculture, while the advanced technology and infrastructure made the sector more productive in recent years. Almost 60% of Denmark is agricultural land (Michael Falch (Danmarks Statistik) 2010). The moderate climate offers a large amount of precipitation in the fields of Denmark, but it should not be taken for granted. Climate change occurs, changes in Earth's climate system result in new weather patterns that last for at least a few decades.

Most of Northern Europe was affected by drought during the summer of 2018. More specific, as it can be seen from the diagram (Figure 1) in Denmark May, June and July were the driest months for the last 8 years. Danish Metrology Institute (DMI) also highlights that the values of the drought index for Denmark for the period from late June till early August exceed the maximum values of the last 13 years (Figure 2). In Denmark, the period from May to July was among the driest in the country in 99 years, with consequences, significant lower agricultural yields and reduced income by farmers. This accentuates the need for irrigation and improved tools for irrigation management, that can be more effective, knowing when, where and how much the irrigation should be.

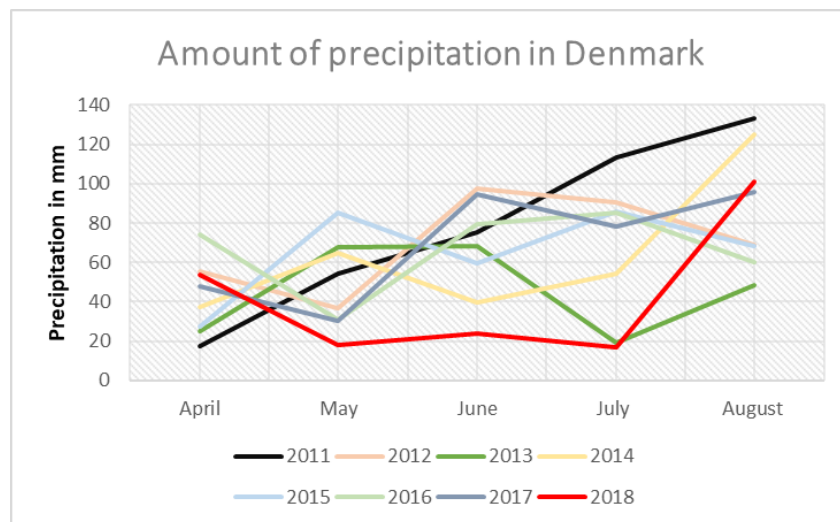


Figure 1: Amount of precipitation in Denmark from April till August, from 2011 – 2018 (Source: (DMI (Danish Meteorology Institute) 2019)).

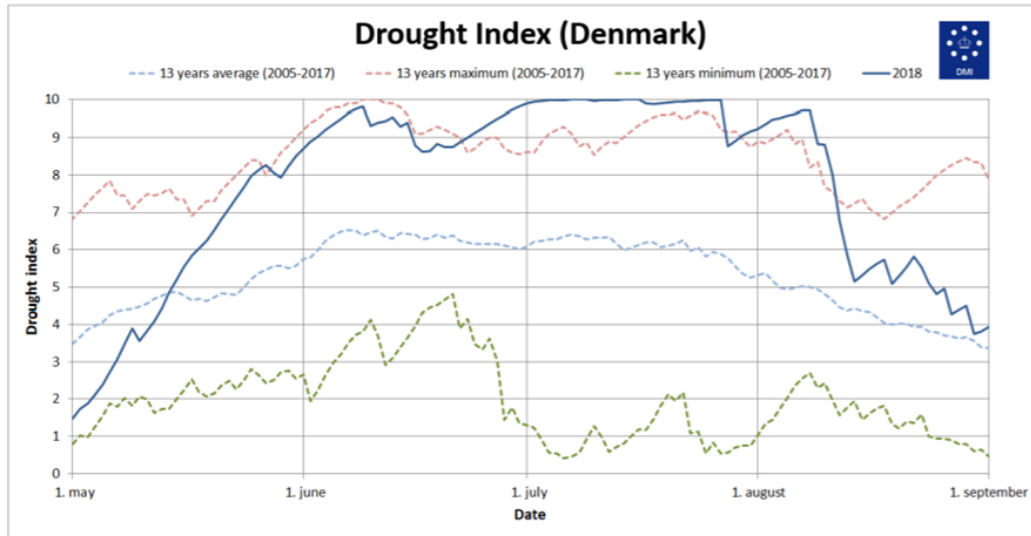


Figure 2: Drought Index for Denmark from 2005 – 2018 (Source: (DMI (Danish Meteorology Institute) 2019)).

Meteorological institutes offer information about the drought on a broad scale, which is not realistic to adopt for every area. Remote sensing offers this possibility using imagery processed data and following a specific methodology. NDVI (Normalized difference vegetation index), for example, is used in some cases to detect water stressed vegetation but the usage of NDVI cannot be effective during the Stage of response as in most of the cases the detection is made when it is already late to respond, meaning that the vegetation is already irreversibly damaged. Using remotely sensed data and meteorological parameters in an irrigation management system, information for the needs of crops, can be retrieved for both small and larger areas.

Optimized irrigation is important not only in Denmark but worldwide and can lead to increase in crop yields, better management of water resources and reduced environmental impact of agricultural production. According to FAO (Food and Agriculture Organization), the amount of crop land's size is stable the last years but there is an increase of irrigated areas (Figure 3). On the other hand, the amount of cropland per person is decreasing. This in conjunction with the constant increase of population, increases also the need for productivity. The increase of cropland's productivity can be achieved by irrigating smart. Irrigating smart among others means to have a correct schedule for irrigation which will increase yield and lead to less leaching of the nitrogen. Nonoptimal scheduling of irrigation can lead to water deficit with consequence yield loss and subsequently leaching of the nitrogen. Similar, water excess can lead to leaching of the nitrogen with consequence yield loss.

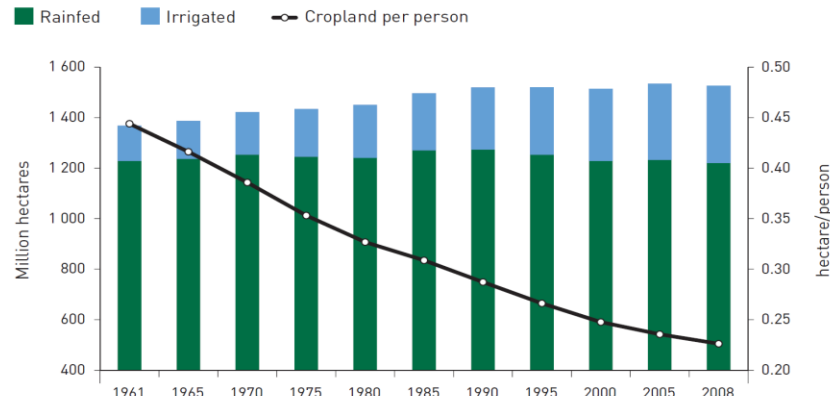


Figure 3: Evolution of land user irrigated and rainfed cropping (1961 – 2008) (FAO (Food and Agriculture Organization) 2014).

The purpose of this study is to create a model so that can be used from farmers, using freely available satellite and weather data. The project was created in cooperation with Sandholt ApS and it was funded by MOIST project via Danish innovation fund.

There is no similar study presently, but there are lots of studies that focusing only on some parts of the characteristics of the plants in order to calculate the water needs without taking into account weather parameters like rainfall. Others are focusing more on the yearly estimation of water needs of the crops while others try to create a local irrigation support system calculating only ground data.

On the other hand, there are lots of studies where describing different and interesting methods about mapping fields like one study in 2018 where a simple methodology for mapping the seasonal cultivated area of the Gash Delta Spate Irrigation System based on satellite images was developed. The methodology of this study combines information from multiple bands to characterize the land surface in terms of spectral indices (e.g., Normalized Difference Vegetation Index (NDVI), and surface temperature (Ts)). The NDVI and Ts values (computed from multi-date images that represented the crop growing period) of some chosen points were used to developed typical NDVI and Ts plots. By analyzing these plots and the cropping calendar, an NDVI and Ts threshold-based algorithm was developed to extract the cultivated area of a given season (Ghebreamlak et al. 2018).

In 2008 Santos et al., combine a water balance model with satellite-based remote-sensing estimates of evapotranspiration (ET) to provide irrigation scheduling guidelines. The satellite-derived ET was used in the daily soil water balance model to improve accuracy of field-by-field ET demands and subsequent field-scale irrigation schedules. The procedure was applied to Genil–Cabra Irrigation Scheme of Spain, where irrigation water supply is often limited by regional drought (Santos et al. 2008).

Scientists are very interested in the correct schedule of irrigation last decades. A group of them already from 1999 tried to explore the water needs in dry regions. They used satellite, ground data, and agrometeorological data in combination to forecast the volume of irrigation water needed during 1993 and 1994 in an irrigation district of 327 km<sup>2</sup> located in the Ebro basin, Spain. Net water requirements of crops were computed from reference evapotranspiration estimates, crop coefficients and effective precipitation (Herrero and Casterad 1999).

## 1.1. Background

The purpose of this chapter is to discover what all the field crops need to grow healthy. In the next pages, significant terms for crop needs and procedures will be described.

### 1.1.1. Evapotranspiration

Evapotranspiration (ET) is a process, which the plants losing water through the stomata (Sandholt, Rasmussen, and Andersen 2002). The loss of water, which are the water needs of the plants, can be obtained from precipitation and soil. In the following Chapter, all these parameters will be described. Evapotranspiration (ET) is a combination of two processes where the soil surface loses water by evaporation and the crop loses water through the stomata by transpiration (Richard G. Allen et al. 1998).

The procedure when energy force water to be moved is called evaporation (E). As evaporation proceeds, the surrounding air becomes gradually saturated and the process will slow down and might stop if the wet air is not transferred to the atmosphere. The replacement of the saturated air with drier air depends mostly on wind speed. Hence, solar radiation, air temperature, air humidity, and wind speed are climatological parameters to consider when assessing the evaporation process (Richard G. Allen et al. 1998).

Transpiration (T) consists of the vaporization of liquid water contained in plant tissues and the vapor removal to the atmosphere. Crops predominately lose their water through stomata. Stomata are small openings on the plant's leaf which gases and water vapor pass (Richard G. Allen et al. 1998). As direct evaporation, transpiration also depends on the energy supply, vapor pressure gradient, and wind. Hence, radiation, air temperature, air humidity, and wind terms should be considered when assessing transpiration. The soil water content and the ability of the soil to conduct water to the roots also determine the transpiration rate, as do waterlogging and soil water salinity. The transpiration rate is also influenced by crop characteristics, environmental aspects and cultivation practices. Different kinds of plants may have different transpiration rates. Not only the type of crop but also the crop development, environment, and management should be considered when assessing transpiration (Richard G. Allen et al. 1998).

As mentioned above evapotranspiration is a complex procedure. Evaporation and transpiration occur at the time and this is the reason why there is no easy way of distinguishing between the two processes. Apart from the water availability in the topsoil, the evaporation from a cropped soil is mainly determined by the fraction of the solar radiation reaching the soil surface. This fraction decreases over the growing period as the crop develops and the crop canopy shades more and more of the ground area. When the crop is small, water is predominately lost by soil evaporation, but once the crop is well developed and completely covers the soil, transpiration becomes the main process (Richard G. Allen et al. 1998). This can be also seen from the diagram follows in Figure 4. The Leaf Area Index (LAI) mentioned in the diagram can be simply defined as the amount of leaf surface area per unit ground area and has broad applications in ecophysiology, water balance modeling, and characterization of vegetation-atmosphere interactions (Rich and Chen 1995). In other words, for the specific diagram, can help to demonstrate the development of the crop and the cover of the soil by the crop.

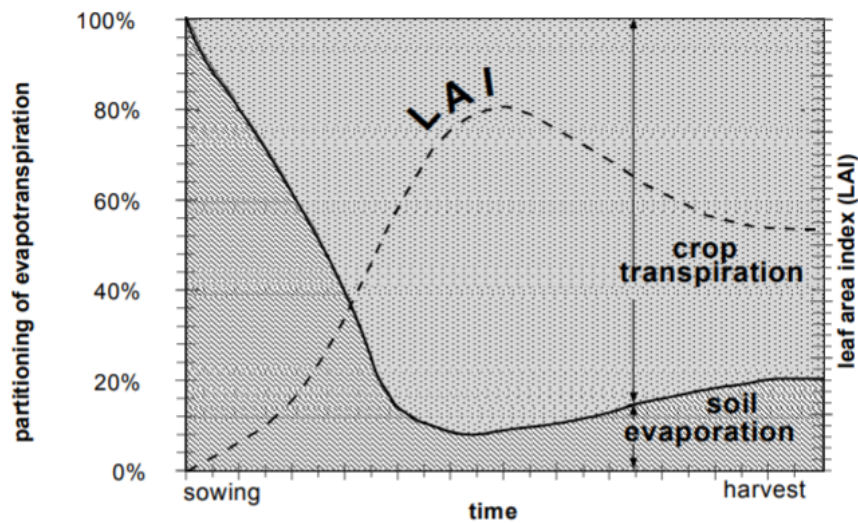


Figure 4: The partitioning of the evapotranspiration into evaporation and transpiration over the growing period for an annual field crop (Richard G. Allen et al. 1998).

#### 1.1.1.1. Units

Normally, the evapotranspiration rate is expressed in millimeters (mm) per unit time. The time unit differs and can be an hour, decade, month or even an entire growing period or year. The rate expresses the amount of water lost from a cropped surface in units of water depth. As one hectare has a surface of 10,000 m<sup>2</sup> and 1 mm is equal to 0.001 m, a loss of 1 mm of water corresponds to a loss of 10 m<sup>3</sup> of water per hectare. In other words, 1 mm day<sup>-1</sup> is equivalent to 10 m<sup>3</sup> ha<sup>-1</sup> day<sup>-1</sup> (Richard G. Allen et al. 1998).

#### 1.1.1.2. Factors affecting Evapotranspiration

Weather parameters, crop characteristics, management, and environmental aspects are factors affecting evaporation and transpiration. Radiation, air temperature, humidity and wind speed are the weather parameters affecting evapotranspiration the most. Crop factors such as the variety and the development stage of the crop affect also evapotranspiration. Differences in resistance to transpiration, crop height, crop roughness, reflection, ground cover and crop rooting characteristics result in different evapotranspiration levels in different types of crops under identical environmental conditions. Factors such as soil salinity, poor land fertility, limited application of fertilizers, the presence of hard or impenetrable soil horizons, the absence of control of diseases and pests and poor soil management may limit the crop development and reduce the evapotranspiration (Richard G. Allen et al. 1998).

#### 1.1.1.3. Determining Evapotranspiration

Evapotranspiration can be estimated from meteorological data, measured using Lysimeters, by calculating the soil water balance, by using both energy balance and microclimatological methods or using a pan and more.

##### 1.1.1.3.1. Evapotranspiration measurements

Evapotranspiration is difficult to measure. There are specific devices and accurate measurements of various physical parameters or the soil water balance in lysimeters to determine evapotranspiration. Nonetheless, the methods are often expensive, and they demand high accurate measurements and can only be fully exploited by well-trained research personnel. Although the methods are inappropriate for

routine measurements, they remain important for the evaluation and calibration of evapotranspiration estimates obtained by more indirect methods (Richard G. Allen et al. 1998).

#### *Energy balance and microclimatological methods*

Evaporation of water requires relatively large amounts of energy, either in the form of sensible heat or radiant energy. Therefore, the evapotranspiration process is governed by energy exchange at the vegetation surface and is limited by the amount of energy available. The energy arriving at the surface must equal the energy leaving the surface for the same time period. The equation for an evaporating surface can be written as is described in equation 1

$$R_n - G - \lambda ET - H = 0 \quad (1)$$

*Equation 1: Energy balance equation (Richard G. Allen et al. 1998).*

where  $R_n$  is the net radiation,  $H$  the sensible heat,  $G$  the soil heat flux and  $\lambda ET$  the latent heat flux. The various terms can be either positive or negative. Positive  $R_n$  supplies energy to the surface and positive  $G$ ,  $\lambda ET$ , and  $H$  remove energy from the surface (Richard G. Allen et al. 1998).

#### *Soil water balance*

Evapotranspiration can also be determined by measuring the various components of the soil water balance. The method consists of assessing the incoming and outgoing water flux into the crop root zone over some time period and can be calculated as is described in equation 2

$$ET = I + P - RO - DP + CR \pm \Delta SF \pm \Delta SW \quad (2)$$

*Equation 2: Soil water balance equation (Richard G. Allen et al. 1998).*

where  $I$ , is irrigation and  $P$ , is the rainfall which adds water to the root zone. An amount of water from  $I$  and  $P$  can be lost by surface runoff ( $RO$ ) and by deep percolation ( $DP$ ) that will eventually recharge the water table. Water might also be transported upward by the capillary rise ( $CR$ ) from a shallow water table towards the root zone or even transferred horizontally by subsurface flow in ( $SFin$ ) or out of ( $SFout$ ) the root zone. In many situations, however, except under circumstances with large slopes,  $SFin$  and  $SFout$  are minor and can be ignored. Soil evaporation and crop transpiration deplete water from the root zone. If all fluxes other than evapotranspiration ( $ET$ ) can be assessed, the evapotranspiration can be deduced from the change in soil water content ( $\Delta SW$ ) over the time period (Richard G. Allen et al. 1998). The above mentioned are illustrated in Figure 5.



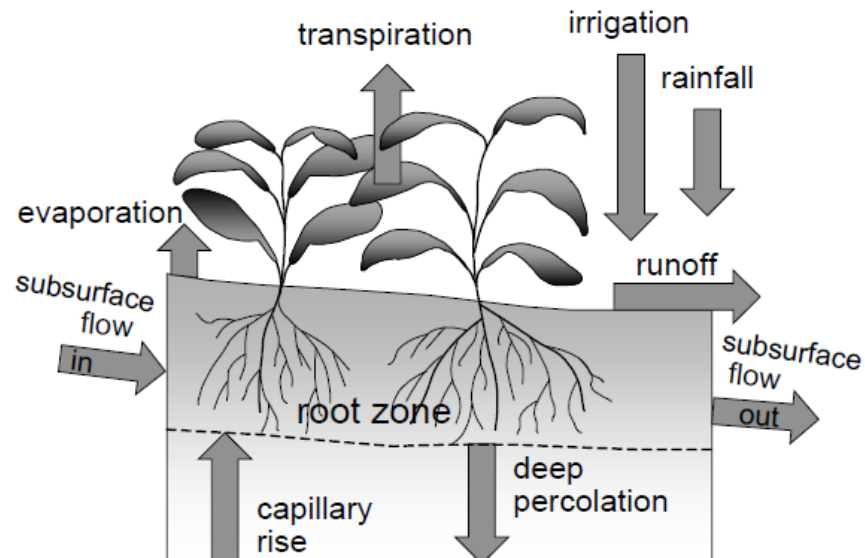


Figure 5: Soil water balance of the root zone (Richard G. Allen et al. 1998).

### Lysimeters

Lysimeters isolate the crop root zone from its environment and control the processes that are difficult to measure. The result is that the different terms in the soil water balance equation can be determined with greater accuracy. This is done in lysimeters (Figure 6) where the crop grows in isolated tanks filled with either disturbed or undisturbed soil (Richard G. Allen et al. 1998).



Figure 6: Lysimeters (BME ABÉT 2008).



#### 1.1.1.3.2. ET computed from meteorological data

Owing to the difficulty of obtaining accurate field measurements, ET is commonly computed from weather data. A large number of empirical or semi-empirical equations have been developed for assessing crop or reference crop evapotranspiration from meteorological data. Some of the methods are only valid under specific climatic and agronomic conditions and cannot be applied under conditions different from those under which they were originally developed (Allen 2006).

Numerous researchers have analyzed the performance of the various calculation methods for different locations. As a result of an Expert Consultation held in May 1990, the FAO Penman-Monteith method is now recommended as the standard method for the definition and computation of the reference evapotranspiration,  $ET_0$ . The ET from crop surfaces under standard conditions is determined by crop coefficients ( $K_c$ ) that relate  $ET_c$  to  $ET_0$ . The ET from crop surfaces under non-standard conditions is adjusted by a water stress coefficient ( $K_s$ ) and/or by modifying the crop coefficient (Allen 2006).

#### 1.1.1.3.3. Pan Evaporation

Evaporation from an evaporation pan (Figure 7) has been used for several centuries as an approximate measure of the potential evapotranspiration ( $E_p$ ). However, research over the last several decades points to the need for caution in the use of general pan evaporation data to estimate lake evaporation or evapotranspiration from the land. Daily pan evaporation ( $E_{pan}$ ) is computed from the change in water storage in the pan, inputs of precipitation and water added to maintain an adequate supply. Owing to differing thermal characteristics between the pan and large water bodies,  $E_{pan}$  tends to overestimate the total amount of evaporation and distort the seasonal distribution (Allen 2006).

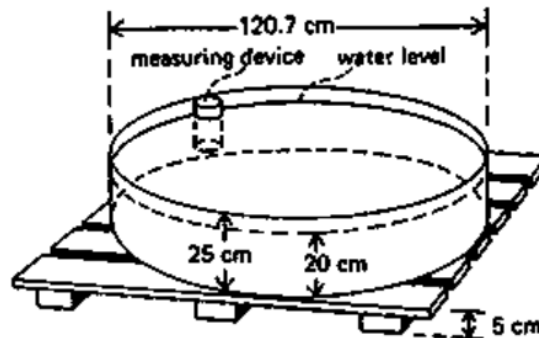


Figure 7: Evaporation pan (Brouwer and Heibloem 1986).

#### 1.1.1.4. Estimating Evapotranspiration

The estimation nor the measurement of the evapotranspiration as mentioned above is not an easy task. There are different formulas that scientists use, to calculate evapotranspiration. Every method uses different or similar approaches from each other. The chapter that follows tries to give an overview of some of the methods and the methods that are used for the delivery of the results in the following chapters.

##### 1.1.1.4.1. Estimating Evapotranspiration using remotely sensed data and energy balance

TSEB (Two-Source Energy Balance) method uses the basic equation (Equation 3) of the energy balance at the surface which can be expressed as:

$$\begin{aligned}
R_n &\approx H + \lambda E + G \\
R_{n,S} &\approx H_s + \lambda E_s + G \\
R_{n,C} &\approx H_c + \lambda E_c
\end{aligned}
\tag{3}$$

*Equation 3: Two-Source Energy Balance method equation (Nieto et al. 2019).*

with  $R_n$  being the net radiation,  $H$  the sensible heat flux,  $\lambda E$  the latent heat flux or Evapotranspiration, and  $G$  the soil heat flux.  $C$  and  $S$  subscripts refer to canopy and soil layers respectively. The symbol  $\approx$  appears since there are additional components of the energy balance that are usually neglected, such as heat advection, storage of energy in the canopy layer or energy for the fixation of  $CO_2$  (Hillel et al. 1998). The key in TSEB models is the partition of sensible heat flux into the canopy and soil layers, which depends on the soil and canopy temperatures ( $T_s$  and  $T_c$  respectively) (Nieto et al. 2019).

The input data required to run the evapotranspiration models comes from three main and two ancillary sources. The main sources are: Sentinel-2 optical observations, Sentinel-3 thermal observations and European Center for Medium-range Weather Forecasts (ECMWF) ERA-5 meteorological reanalysis data. The ancillary sources are a digital elevation model (DEM) from the Shuttle Radar Topography Mission (SRTM) satellite, and land cover map created as part of the ESA Climate Change Initiative (CCI) (Nieto et al. 2019).

The actual evapotranspiration ( $ET_a$ ) can be calculated using the above-mentioned method.

#### 1.1.1.4.2. Estimating Evapotranspiration from meteorological data

The term evapotranspiration ( $ET$ ), developed over the last half-century (Thornthwaite 1948), is used as a simplifying concession to describe the combined vapor fluxes from soil and vegetation. Some scientists argue that evaporation from soil and evaporation from plants via transpiration both constitute evaporation, so that only the term evaporation is necessary, and the use of the term transpiration is redundant. However, the term  $ET$  is useful for the efficiency of communication and visualization, since the use of only the term evaporation generally requires qualification as “evaporation from the soil” or “evaporation from vegetation leaf surfaces” or “evaporation of transpired water within leaves”, or the sum of all these processes. For communication efficiency, the sum of all three of these evaporation components, which constitutes the total flux of vapor from a vegetated surface, is commonly referred to as  $ET$  (if one wishes, one can think of the term  $ET$  as representing “evaporation total”) (Allen 2006).

Calculations which are made using meteorological data in order to compute  $ET$  usually give the results of the potential evapotranspiration ( $E_p$ ) of the crop. The term “potential evapotranspiration” ( $ET_p$ ) had widespread usage from the 1940s through the 1970s (Allen 2006), when the term was used to represent a maximum evaporative index from which to derive estimates of actual  $ET$  ( $ET_a$ ) from vegetation (Penman 1963). However, there are several majors, contrasting definitions for  $ET_p$  and several challenges associated with its usage. One of the primary definitions used for  $ET_p$  is as the rate of evaporation and transpiration from a saturated (free-water) vegetated surface so that the evaporation process occurs at the potential level. This definition can in principle be applied to any surface, including vegetation, having a free-water (saturated) surface (Allen 2006).

As can be seen from Figure 8, ET can be calculated considering different factors or a combination of factors. Only weather parameters can be used to calculate ET, but these can be combined with crop factors and management and environmental factors for more accurate results.

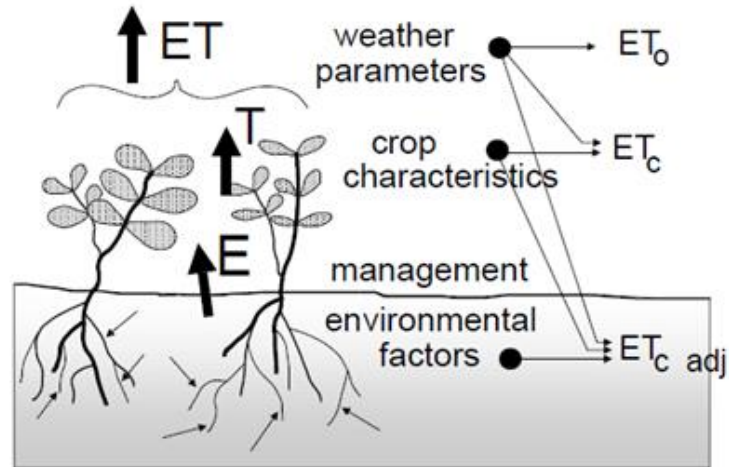


Figure 8: Factors affecting evapotranspiration (Richard G. Allen et al. 1998).

In this chapter will be described what is the reference evapotranspiration ( $ET_0$ ) and how can be calculated and how  $ET_0$  can be used to calculate the crop evapotranspiration considering the crop characteristics ( $ET_c$ ) or the environmental factors ( $ET_{c \text{ adj}}$ ). The concepts of the calculations for the  $ET_c$  and  $ET_{c \text{ adj}}$  are described in Figure 9.

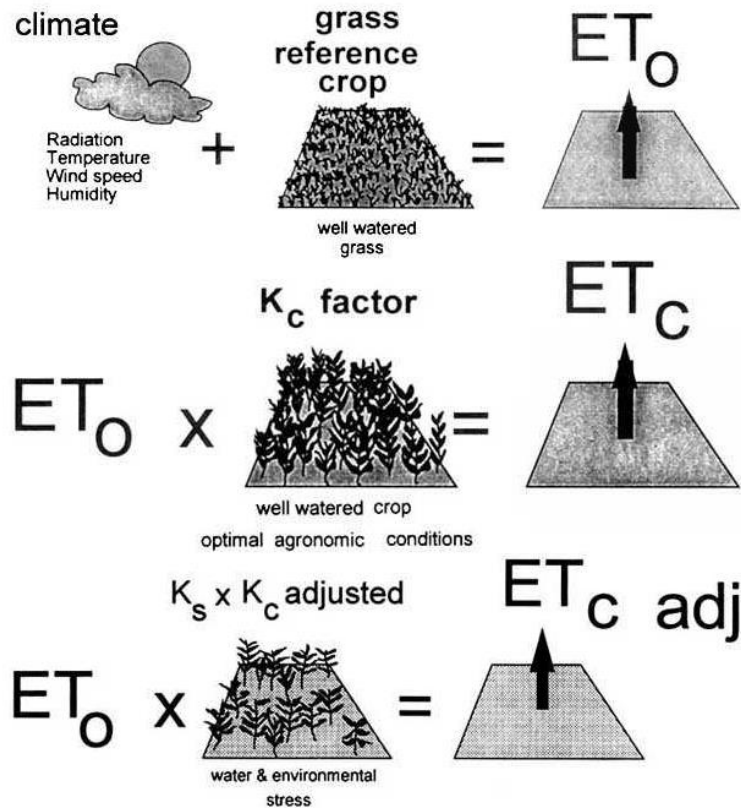


Figure 9: The calculation of  $ET_0$ ,  $ET_c$  and  $ET_{c \text{ adj}}$  (Brouwer and Heibloem 1986).

#### Reference Evapotranspiration ( $ET_0$ )

The evapotranspiration rate from a reference surface, not short of water, is called the reference crop evapotranspiration or reference evapotranspiration and is denoted as  $ET_0$ . The reference surface is a hypothetical grass reference crop with specific characteristics. The only factors affecting  $ET_0$  are climatic parameters. The main weather parameters that affect evapotranspiration are radiation, air temperature, humidity and wind speed. Thus,  $ET_0$  is a climatic parameter and can be computed from weather data.  $ET_0$  expresses the evaporating power of the atmosphere at a specific location and time of the year and does not consider the crop characteristics and soil factors (Richard G. Allen et al. 1998).

#### Penman – Monteith Equation

In 1948, Penman combined the energy balance with the mass transfer method and derived an equation to compute the evaporation from an open water surface from standard climatological records of sunshine, temperature, humidity and wind speed. The Penman-Monteith approach includes all parameters that govern energy exchange and corresponding latent heat flux (evapotranspiration) from uniform expanses of vegetation. The parameters are measured or can be calculated from weather data. The equation can be utilized for the direct calculation of any crop evapotranspiration as the surface and aerodynamic resistances are crop specific (Richard G. Allen et al. 1998).

The reference surface is a hypothetical grass reference crop with an assumed crop height of 0.12 m, a fixed surface resistance of  $70 \text{ s m}^{-1}$  and an albedo (part of the energy from the sunlight that casts back into the atmosphere) of 0.23. The reference surface closely resembles an extensive surface of green, well-watered grass of uniform height, actively growing and completely shading the ground. The fixed surface

resistance of  $70 \text{ s m}^{-1}$  implies a moderately dry soil surface resulting from about a weekly irrigation frequency (Richard G. Allen et al. 1998). The Penman – Monteith equation (Equation 4) can be expressed as:

$$ET_0 = \frac{0.408 \Delta (R_n - G) + \gamma \frac{900}{T + 273} u_2 (e_s - e_a)}{\Delta + \gamma (1 + 0.34 u_2)} \quad (4)$$

*Equation 4: Penman – Monteith equation (Richard G. Allen et al. 1998).*

Where  $ET_0$  is reference evapotranspiration ( $\text{mm day}^{-1}$ ),  $R_n$  is the net radiation at the crop surface ( $\text{MJ m}^{-2} \text{ day}^{-1}$ ),  $G$  is the soil heat flux density ( $\text{MJ m}^{-2} \text{ day}^{-1}$ ),  $T$  is the mean daily air temperature at 2m height ( $^{\circ}\text{C}$ ),  $u_2$  is the wind speed at 2m height ( $\text{m s}^{-1}$ ),  $e_s$  is the saturation vapor pressure (kPa),  $e_a$  is the actual vapor pressure (kPa),  $e_s - e_a$  is the saturation vapor pressure deficit (kPa),  $\Delta$  is the slope vapor pressure curve ( $\text{kPa } ^{\circ}\text{C}^{-1}$ ) and  $\gamma$  is the psychrometric constant ( $\text{kPa } ^{\circ}\text{C}^{-1}$ ).

#### Blaney – Criddle Method

There are many theoretical methods to determine  $ET_0$ . The most commonly used theoretical method is the Penman method but this method, though, can be complicated. Blaney – Criddle is a theoretical method with which the reference evapotranspiration ( $ET_0$ ) can be calculated. This method is simple and uses temperature measurements only, however, this method is not very accurate. It provides a rough estimate. Especially under "extreme" climatic conditions, the Blaney-Criddle method is inaccurate: in windy, dry, sunny areas, the  $ET_0$  is underestimated (up to some 60%), while in calm, humid, clouded areas, the  $ET_0$  is overestimated (up to some 40%) (Brouwer and Heibloem 1986).

The Blaney – Criddle method always refers to mean monthly values, both for the temperature and the  $ET_0$  and it can be expressed as it can be seen in equation 5.

$$ET_0 = p (0.46 T_{\text{mean}} + 8) \quad (5)$$

*Equation 5: Blaney – Criddle method equation (Richard G. Allen et al. 1998).*

Where  $ET_0$  is the reference crop evapotranspiration ( $\text{mm/day}$ ) as an average for a period of 1 month,  $T_{\text{mean}}$  is the mean daily temperature ( $^{\circ}\text{C}$ ) and  $p$  is the mean daily percentage of the annual daytime hours.

#### Crop evapotranspiration under standard conditions ( $ET_c$ )

The crop evapotranspiration under standard conditions, denoted as  $ET_c$ , is the evapotranspiration from disease-free, well-fertilized crops, grown in large fields, under optimum soil water conditions, and achieving full production under the given climatic conditions. The effects of various weather conditions on evapotranspiration are incorporated into  $ET_0$ . The effects of characteristics that distinguish the cropped surface from the reference surface are integrated into the crop coefficient. By multiplying  $ET_0$  by the crop coefficient,  $ET_c$  is determined (Richard G. Allen et al. 1998).

Evaporation and transpiration differences between field crops and the reference grass can be integrated in a single crop coefficient ( $K_c$ ) (single crop coefficient approach) or separated into two coefficients. A crop transpiration ( $K_{cb}$ ) and a soil evaporation coefficient ( $K_e$ ) (dual crop coefficient approach). In the single crop coefficient approach, the difference in evapotranspiration between the cropped and reference grass is combined into one single coefficient. In the dual crop coefficient approach, the crop coefficient

is split into two factors describing separately the differences in evaporation and transpiration between the crop and reference surface (Richard G. Allen et al. 1998).

#### Single crop coefficient approach

In the crop coefficient approach the crop evapotranspiration,  $ET_c$ , is calculated by multiplying the reference crop evapotranspiration,  $ET_0$ , by a crop coefficient,  $K_c$ . The equation (Equation 6) of the single approach can be expressed as:

$$ET_c = K_c ET_0 \quad (6)$$

Equation 6: Single crop coefficient equation (Richard G. Allen et al. 1998).

where  $ET_c$  is the crop evapotranspiration ( $\text{mm d}^{-1}$ ),  $K_c$  is the crop coefficient and it is dimensionless, and  $ET_0$  is the reference crop evapotranspiration ( $\text{mm d}^{-1}$ ).  $ET_0$  represents an index of climatic demand,  $K_c$  varies mostly with the specific crop characteristics and only to a limited extent with climate. This enables the transfer of standard values for  $K_c$  between locations and between climates. This has been a primary reason for the global acceptance and usefulness of the crop coefficient approach (Richard G. Allen et al. 1998).

Some examples of values of  $K_c$  are presented in Figure 10. The values of the coefficient referring only at the mid-season of the crop. The grass coefficient, which is 0, can also be seen in the figure. Crops with larger  $K_c$  have larger  $ET_c$  values and crops with minor coefficient take smaller values of  $ET_c$ .

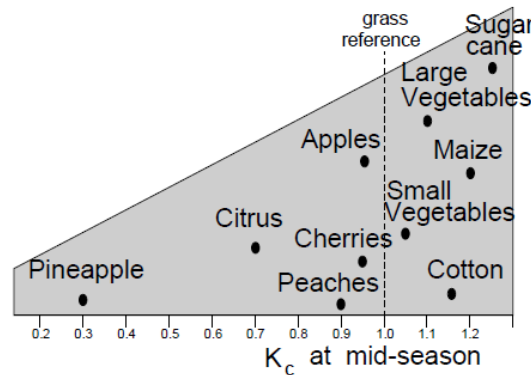


Figure 10: Typical  $K_c$  for different types of full-grown crops (Richard G. Allen et al. 1998).

#### Dual crop coefficient approach

In the dual crop coefficient approach the crop evapotranspiration,  $ET_c$ , is calculated by multiplying the reference crop evapotranspiration,  $ET_0$ , by two separate coefficients, one for crop transpiration, i.e., the basal crop coefficient ( $K_{cb}$ ), and one for soil evaporation ( $K_e$ ). The equation (Equation 7) of the dual approach can be expressed as:

$$ET_c = (K_{cb} + K_e) ET_0 \quad (7)$$

Equation 7: Dual crop coefficient equation (Richard G. Allen et al. 1998).

The dual crop coefficient approach is more complicated and more computationally intensive than the single crop coefficient approach ( $K_c$ ). The procedure is conducted daily and is intended for applications

using computers. It is recommended that the approach be followed when improved estimates for  $K_c$  are needed, for example to schedule irrigations for individual fields daily (Richard G. Allen et al. 1998).

The differences between the single and dual crop coefficient can be seen from the following figure (Figure 11). The  $K_c$  value is the same but using the dual coefficient the values of the transpiration and evaporation can be distinguished. Crop coefficient curves showing the basal  $K_{cb}$  (thick line), soil evaporation  $K_e$  (thin line) and the corresponding single  $K_c$  (dashed line). The values are calculated by the crop development.

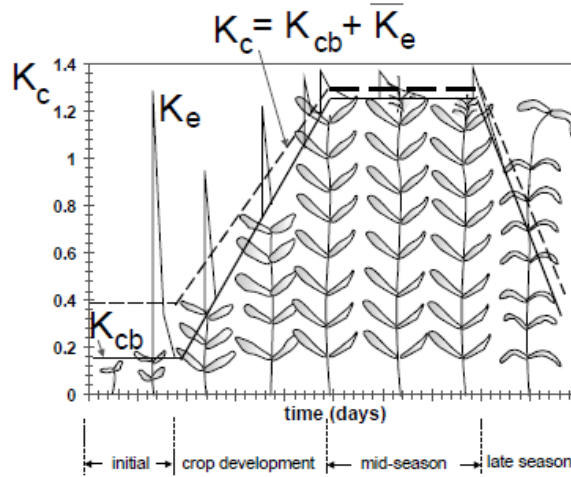


Figure 11: The single and dual crop coefficient values for crop (Richard G. Allen et al. 1998).

#### *Crop evapotranspiration under non – standard conditions ( $ET_{c\ adj}$ )*

Soil salinity, limited application of fertilizers, poor land fertility, the absence of control of diseases and pests, the presence of hard or impenetrable soil horizons and poor soil management are factors that may limit the crop development and reduce the evapotranspiration. Forces acting on the soil water decrease its potential energy and make it less available for plant root extraction. When the soil is wet, the water has a high potential energy, is relatively free to move and is easily taken up by the plant roots instead in dry soils, the water has a low potential energy and is strongly bound by capillary and absorptive forces to the soil matrix and is less easily extracted by the crop (Richard G. Allen et al. 1998).

Crop evapotranspiration under non – standard conditions ( $ET_{c\ adj}$ ) can be expressed with the following equation using the single crop coefficient (Equation 8) or the dual crop coefficient (Equation 9) mentioned in the previous chapter.

$$ET_{c\ adj} = K_s K_c ET_o \quad (8)$$

Equation 8: Crop evapotranspiration under non – standard conditions ( $ET_{c\ adj}$ ) equation using the single crop coefficient (Richard G. Allen et al. 1998).

$$ET_{c\ adj} = (K_s K_{cb} + K_e) ET_o \quad (9)$$

Equation 9: Crop evapotranspiration under non – standard conditions ( $ET_{c\ adj}$ ) equation using the dual crop coefficient (Richard G. Allen et al. 1998).

$K_s$  describes the effect of water stress on crop transpiration and for soil water limiting conditions,  $K_s < 1$ . When there is no soil water stress,  $K_s = 1$ .

### 1.1.2. Soil

The soil particles even if it seems to touch each other, they have spaces in between. These spaces are called pores. The pores are mainly filled with air when the soil is dry, in contrast, after irrigation or rainfall, the pores are mainly filled with water. Living material is found in the soil and it can be live roots as well as beetles, worms, larvae, etc. The living material helps to aerate the soil and thus create favorable growing conditions for the plant roots (Figure 12) (Brouwer, Goffeau, and Heibloem 1985).

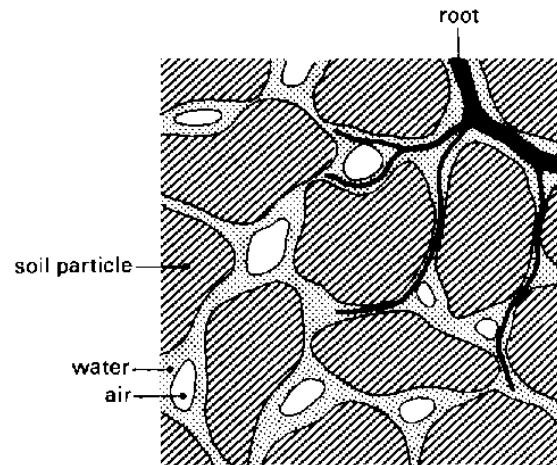


Figure 12: The composition of the soil (Brouwer, Goffeau, and Heibloem 1985).

#### 1.1.2.1. Soil moisture content

The soil moisture content indicates the amount of water present in the soil. It is commonly expressed as the amount of water (in mm of water depth) present in a depth of one meter of soil. For example: when an amount of water (in mm of water depth) of 150 mm is present in a depth of one meter of soil, the soil moisture content is 150 mm/m (Figure 13) (Brouwer, Goffeau, and Heibloem 1985).



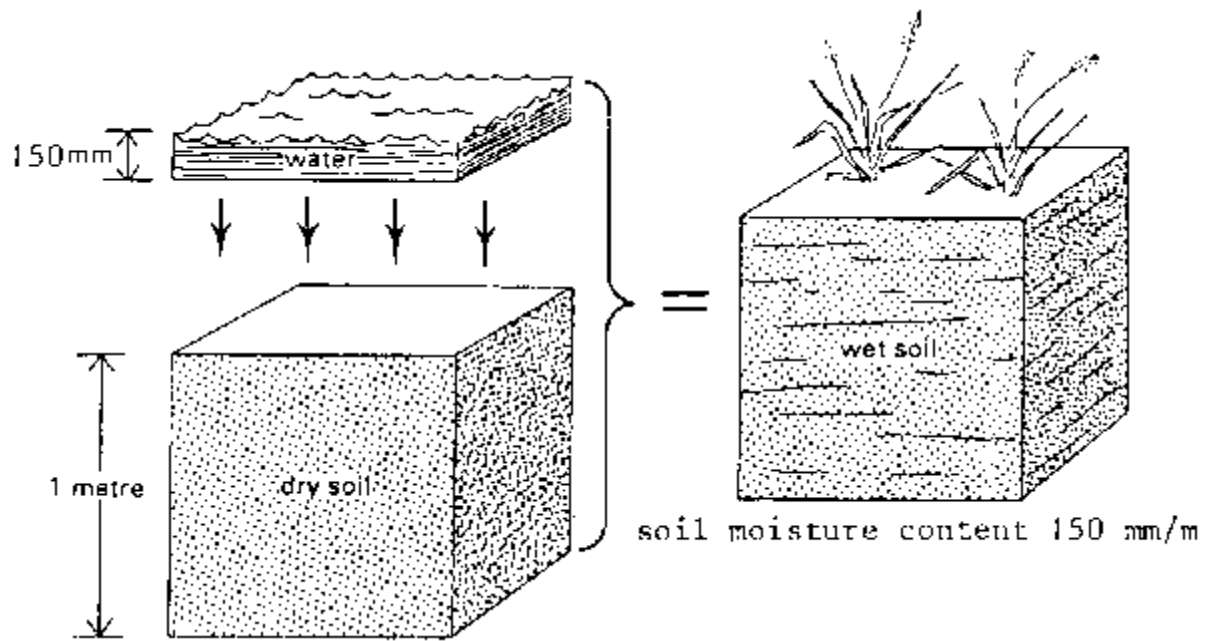


Figure 13: A soil moisture content of 150 mm/m (Brouwer, Goffeau, and Heibloem 1985).

The soil moisture content can also be expressed in percent of volume. In the example above, 1 m<sup>3</sup> of soil (e.g. with a depth of 1 m, and a surface area of 1 m<sup>2</sup>) contains 0.150 m<sup>3</sup> of water (e.g. with a depth of 150 mm = 0.150 m and a surface area of 1 m<sup>2</sup>). This results in a soil moisture content in volume percent of:

$$\frac{0.150\text{m}^3}{1\text{m}^3} \times 100\% = 15\%$$

Thus, a moisture content of 100 mm/m corresponds to a moisture content of 10% volume. The amount of water stored in the soil is not constant with time but may vary (Brouwer, Goffeau, and Heibloem 1985). Depending on the amount of water existing in the soil, the soil can be saturated, the soil moisture can reach the field capacity level or the permanent wilting point.

#### 1.1.2.1.1. Saturation

When a rain or irrigation event happens, the soil pores will fill with water. The soil is said to be saturated if all soil pores are filled with water. In this case, there is no air left inside the soil. Plants need except of water in the soil, air. In the case of soil saturation, the plants are suffering because of the lack of air (Figure 14 a). There are many crops that cannot withstand saturated soil conditions for a period of more than 2 – 5 days. There are also some exceptions to this rule e.g. rice. The period of saturation of the topsoil usually does not last long. After the rain or the irrigation event, part of the water will move downward. This process is called drainage or percolation. The water drained from the pores s replaced by air. In coarse textured (sandy) soils, drainage is completed within a period of a few hours. In fine textured (clayey) soils, drainage may take some (2-3) days (Brouwer, Goffeau, and Heibloem 1985).

#### 1.1.2.1.2. Field capacity

Soon after the drainage, the large soil pores are filled with both air and water and the smaller pores are still full of water. At this stage, the soil is said to be at field capacity. At field capacity, the water and air

contents of the soil are considered to be ideal for crop growth (Figure 14 b) (Brouwer, Goffeau, and Heibloem 1985).

#### 1.1.2.1.3. Permanent wilting point

The water which is stored in the soil is taken up by the plant roots or evaporated from the soil to the atmosphere and eventually, the soil gradually dries out if no additional water is supplied in it. Drier soil means larger difficult for the plant roots to extract the water from it. The stage that the uptake of water is not sufficient to meet the plant's needs, the plant loses freshness and wilts, the leaves change color from green to yellow and finally, the plant dies is called permanent wilting point. The soil still contains some water, but it is too difficult for the roots to suck it from the soil (Figure 14 c) (Brouwer, Goffeau, and Heibloem 1985).

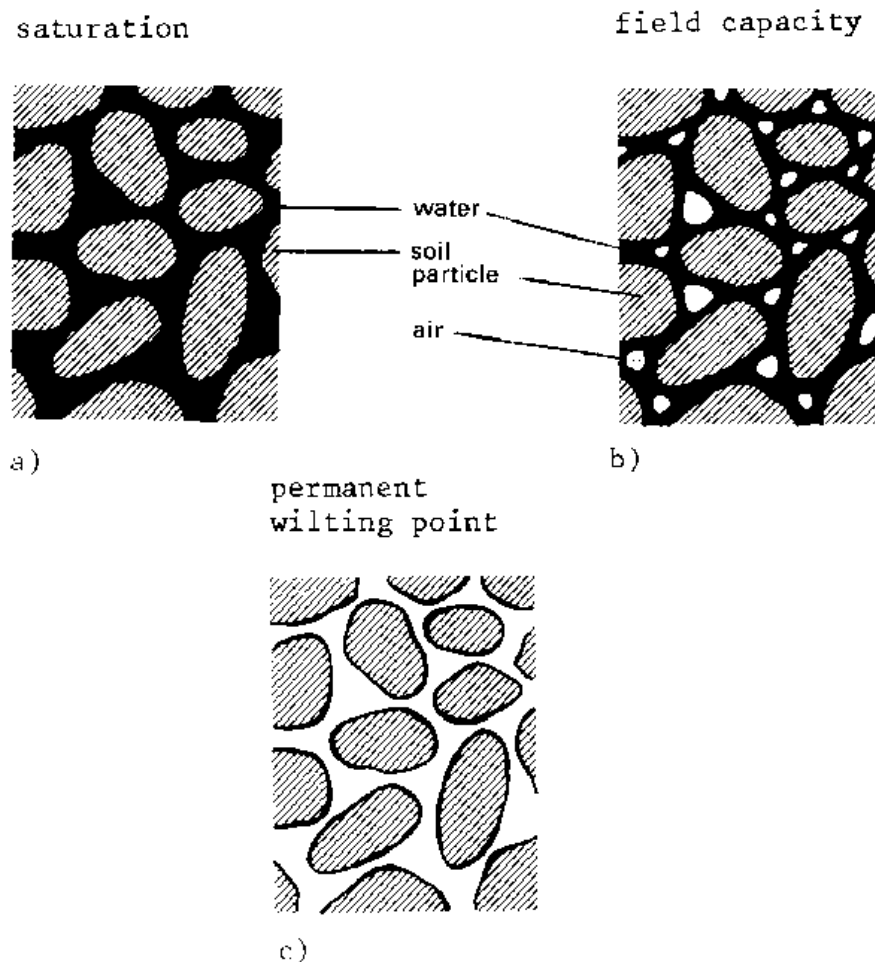


Figure 14: Some soil moisture characteristics (Brouwer, Goffeau, and Heibloem 1985).

#### 1.1.2.2. Available water content

The soil can be compared to a water reservoir for the plants. When the soil is saturated, the reservoir is full. Some water drains rapidly below the root zone before the plant can use it (Figure 15 a) but when this water has drained away, the soil is at field capacity (Figure 15 b). The plant roots draw water from what remains in the reservoir. When the soil reaches the permanent wilting point (Figure 15 c), the remaining water is no longer available to the plant. The amount of water actually available to the plant is the amount

of water stored in the soil at field capacity minus the water that will remain in the soil at permanent wilting point (Brouwer, Goffeau, and Heibloem 1985). This is illustrated in figure 15 d.

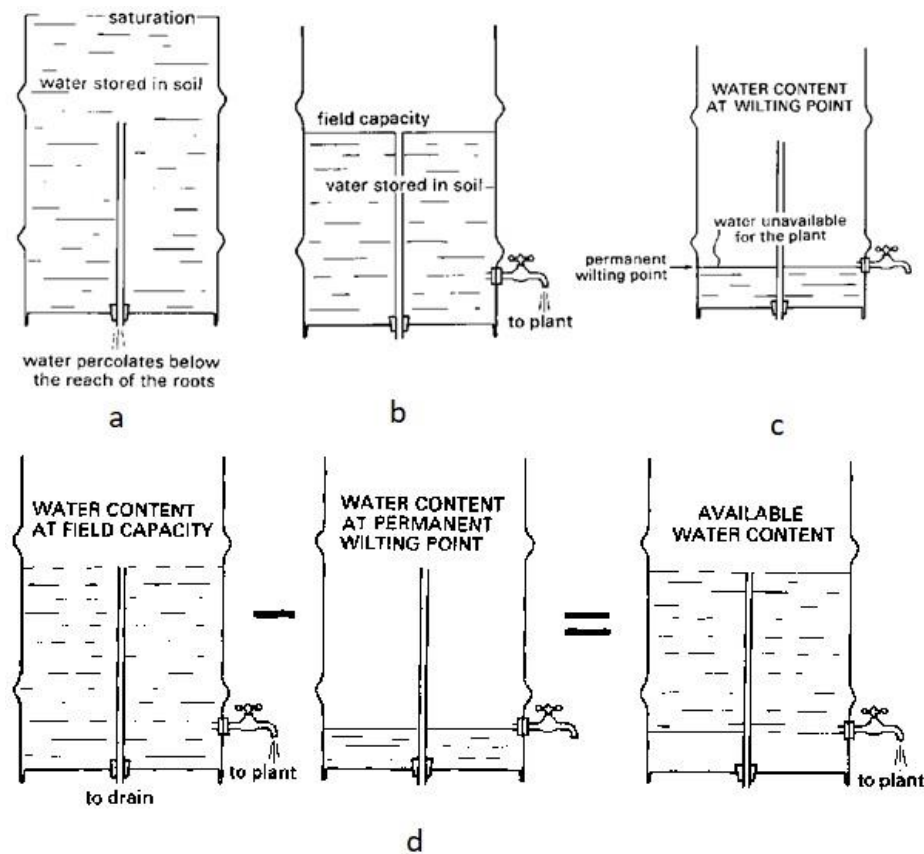


Figure 15: Saturation, field capacity, wilting point and available water content (Brouwer, Goffeau, and Heibloem 1985).

The available water content depends greatly on the soil texture and structure. A range of values for different types of soil is given in the following figure. As it can be seen from the diagram in figure 16 the percentage of the water content for saturated soil, the soil at field capacity and soil at the permanent wilting point differs for each soil type. Sandy soils need a smaller percentage of water in the soil than clay. The percentage in between the red and blue line is the ideal for most of the crops.

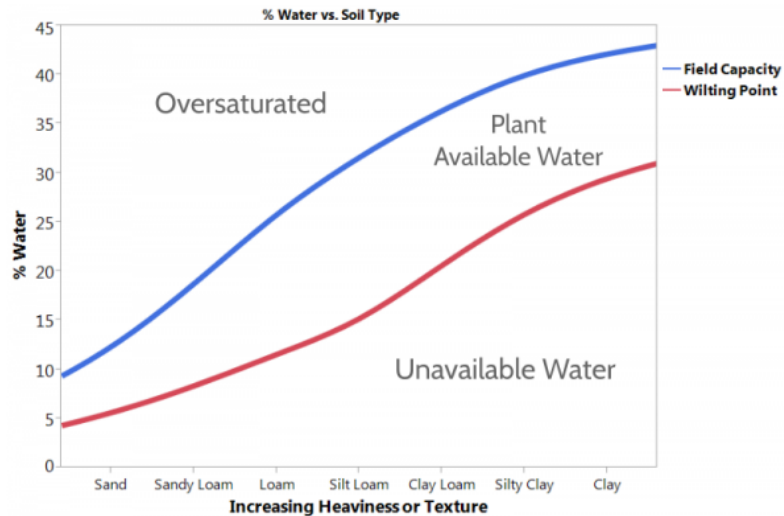


Figure 16: Field Capacity, Wilting Point, Available and Unavailable water vs. Soil Texture (Spruceirrigation.com 2019).

### 1.1.3. Water needs

As mentioned above, all field crops need soil, water, air and light (sunshine) to grow. The soil gives stability to the plants and it also stores the water and nutrients which the plants can take up through their roots. The energy which is necessary for plant growth is provided from the sun. Water is another factor that affects the wellbeing of the plant. Crops need water for transpiration and evaporation. Without it, the crops cannot grow while too much water is not good for many crops either. If there is too little rain, water must be supplied from other sources, so irrigation is needed. The amount of irrigation water which is needed depends not only on the amount of water already available from rainfall but also on the total amount of water needed by the various crops (Brouwer and Heibloem 1986).

#### 1.1.3.1. Crop water needs

The water need of a crop is usually expressed in mm/day, mm/month or mm/season. Suppose the water need of a certain crop in a very hot, dry climate is 10 mm/day. This means that each day the crop needs a water layer of 10 mm over the whole area on which the crop is grown. It does not mean that this 10 mm must indeed be supplied by rain or irrigation every day. It is, of course, still possible to supply, for example, 50 mm of irrigation water every 5 days. The irrigation water will then be stored in the root zone and gradually be used by the plants: every day 10 mm (Brouwer and Heibloem 1986).

The crop water need depends on three parameters, the climate, the crop type and the growth stage of the plants. A specific crop needs more water per day if the area has no clouds, is hot, the humidity is low, or it is windy but needs less water in cloudy areas or cooler areas and areas with little wind or high humidity. The crop type is also influencing the daily crop water needs. FAO uses a reference type of crop for all calculations which is the grass. As can be seen from the following table (Table 1) the daily water needs of the grass depending on the climatic zone and daily temperature.

Table 1: Average daily water need of standard grass during the irrigation season (Brouwer and Heibloem 1986).

Climatic zone	Mean daily temperature		
	low	medium	high
	(less than 15°C)	(15-25°C)	(more than 25°C)
Desert/arid	4-6	7-8	9-10
Semi arid	4-5	6-7	8-9
Sub-humid	3-4	5-6	7-8
Humid	1-2	3-4	5-6

According to the above figure, the standard grass crop grown in a Sub-humid climate with a mean temperature of 17 °C needs approximately 5.5 mm of water per day. The same grass crop grown in a Semi-arid climate with a mean temperature of 30°C needs some 7.5 mm of water per day. According to FAO this daily water need of the standard grass crop is also called "reference crop evapotranspiration" (Brouwer and Heibloem 1986).

There are some types of crops that have the same water needs in peak period as standard grass but there are other types that they need different amount of water. For the various field crops, it is possible to determine how much water they need compared to the standard grass using the following table (Table 2).

Table 2: Crop water needs in the peak period of various field crops as compared to standard grass (Brouwer and Heibloem 1986)

Column 1	Column 2	Column 3	Column 4	Column 5
-30%	-10%	same as standard grass	+ 10%	+20%
citrus	cucumber	carrots	barley	paddy rice
olives	radishes	crucifers (cabbage, cauliflower, broccoli, etc.)	beans	sugarcane
grapes	squash	lettuce	maize	banana
		melons	flax	nuts & fruit trees with cover crop
		onions	small grains	
		peanuts	cotton	
		peppers	tomato	
		spinach	eggplant	
		tea	lentils	
		grass	millet	
		cacao	oats	
		coffee	peas	
		clean cultivated nuts & fruit trees e.g. apples	potatoes	
			safflower	
			sorghum	
			soybeans	
			sugarbeet	
			sunflower	
			tobacco	
			wheat	

The table above indicates five groups of crops. The crops in column 1 need 30 % less water than grass in their peak period. The crops in column 2 need 10 % less water than grass. The crops in column 3 need the same amount of water as grass. The crops in columns 4 and 5 need 10 and 20 % more water than grass in their peak period respectively.

Some examples of the amount of the total crop water needs can be seen from Table 3.

*Table 3: Approximate values of seasonal crop water needs by FAO (Brouwer and Heibloem 1986).*

Crop	Crop water need (mm/total growing period)
Alfalfa	800-1600
Banana	1200-2200
Barley/Oats/Wheat	450-650
Bean	300-500
Cabbage	350-500
Citrus	900-1200
Cotton	700-1300
Maize	500-800
Melon	400-600
Onion	350-550
Peanut	500-700
Pea	350-500
Pepper	600-900
Potato	500-700
Rice (paddy)	450-700
Sorghum/Millet	450-650
Soybean	450-700
Sugarbeet	550-750
Sugarcane	1500-2500
Sunflower	600-1000
Tomato	400-800

The crop type not only has an influence on the daily water need of a fully-grown crop, i.e. the daily peak water need, but the crop type also has an influence on the duration of the total growing season of the crop, and thus on the seasonal water need.

Data on the duration of the total growing season of the various crops grown in an area is best to be obtained locally. The duration of the total growing season has an enormous influence on the seasonal crop water need. There are, for example, some crops with a short growing cycle (e.g. 75 – 90 days) and others with a long growing cycle (e.g. 150 – 220 days). Indicative values or approximate values for the duration of the total growing season for the various field crops are described in the next figure (Table 4). These values are only rough approximations and it is much better to obtain the values locally (Brouwer and Heibloem 1986).

Table 4: Indicative values of the total growing period (Brouwer and Heibloem 1986).

Crop	Total growing period (days)	Crop	Total growing period (days)
Alfalfa	100-365	Millet	105-140
Banana	300-365	Onion green	70-95
Barley/Oats/Wheat	120-150	Onion dry	150-210
Bean green	75-90	Peanut/Groundnut	130-140
Bean dry	95-110	Pea	90-100
Cabbage	120-140	Pepper	120-210
Carrot	100-150	Potato	105-145
Citrus	240-365	Radish	35-45
Cotton	180-195	Rice	90-150
Cucumber	105-130	Sorghum	120-130
Eggplant	130-140	Soybean	135-150
Flax	150-220	Spinach	60-100
Grain/small	150-165	Squash	95-120
Lentil	150-170	Sugarbeet	160-230
Lettuce	75-140	Sugarcane	270-365
Maize sweet	80-110	Sunflower	125-130
Maize grain	125-180	Tobacco	130-160
Melon	120-160	Tomato	135-180

There is a large variety of values not only between crops but also within one crop type. An assumption that can be made is that the growing period for a certain crop is longer for cool climate and shorter when the climate is warm (Brouwer and Heibloem 1986).

#### 1.1.3.2. Irrigation requirements

Pruitt in 1984, mentions that the net irrigation requirements of the crops ( $I_n$ ) are calculated using the field water balance. The variables include crop evapotranspiration ( $ET_{crop}$ ), rainfall ( $P_e$ ), groundwater contribution ( $G_e$ ) and stored soil water at the beginning of each period ( $W_b$ ), or:

$$I_n = ET_{crop} - (P_e + G_e + W_b)$$

losses - gains

All variables are expressed in units of depth of water (mm) and, depending on the accuracy required,  $I_n$  can be determined for seasonal, monthly or 10-day periods. For preliminary planning, monthly data are frequently used. The sum of  $I_n$  for the different crops over the entire irrigated area forms the basis for determining the necessary supply (Pruitt 1984).

According to Pruitt, calculating the amount of water stored in the soil and the amount that can be used from the crop is not easy. FAO introduced some years later a simpler method for calculating the irrigation water need using the field water balance. This method can be seen in Figure 17 where the irrigation water need is defined by the crop water need and the available rainwater.

$$\text{CROP WATER NEED} - \text{AVAILABLE RAIN WATER} = \text{IRRIGATION WATER NEED}$$

Figure 17: Irrigation water need (Brouwer and Heibloem 1986).

#### 1.1.3.3. Effective Rainfall

The water can be supplied to the crops by rainfall (also called precipitation), by irrigation or by a combination of rainfall and irrigation. If the rainfall is sufficient to cover the water needs of the crops, irrigation is not required. If there is no rainfall, all the water that the crops need must be supplied by irrigation. If there is some rainfall, but not enough to cover the water needs of the crops, irrigation water must supplement the rain water in such a way that the rain water and the irrigation water together cover the water needs of the crop. This is often called supplemental irrigation: the irrigation water supplements or adds to the rain water (Brouwer and Heibloem 1986).

Part of the rain water percolates below the root zone of the plants and part of the rain water flows away over the soil surface as run-off. This deep percolation water and run-off water cannot be used by the plants. In other words, part of the rainfall is not effective. The remaining part is stored in the root zone and can be used by the plants. This remaining part is the so-called effective rainfall. The factors which influence which part is effective and which part is not effective, include the climate, the soil texture, the soil structure and the depth of the root zone (Brouwer and Heibloem 1986).

The effective rainfall is the total rainfall minus runoff minus evaporation and minus deep percolation. Only the water retained in the root zone can be used by the plants and represents what is called the effective part of the rainwater. The term effective rainfall is used to define this fraction of the total amount of rainwater useful for meeting the water need of the crops. 2 simple formulas are introduced by FAO to estimate the fraction of the total rainfall which is used effectively. These formulas (Equation 10) can be applied in areas with a maximum slope of 4-5 % (Brouwer and Heibloem 1986).

$$\begin{aligned} P_e &= 0.8 P - 25 \text{ if } P > 75 \text{ mm/month} \\ P_e &= 0.6 P - 10 \text{ if } P < 75 \text{ mm/month} \end{aligned} \quad (10)$$

*Equation 10: Formulas for effective rainfall.*

Where  $P_e$  is the effective rainfall and  $P$  is the seasonal rain precipitation.



## 1.2. Problem statement

Denmark was affected by drought during the summer of 2018 and this highlighted the need for improved tools for irrigation management. Optimize irrigation can be beneficial in both environmental and economic terms. The purpose of this study is to estimate the amount of water that farmers need to irrigate in each area of the field. The study will create a decision support system for irrigation management on a national scale for Denmark. In order to test our approach, one municipality in Denmark will be selected as a pilot area.

## 1.3. Research questions

To estimate the amount of water for irrigation, a set of research questions were outlined in order to help in the development of the support system.

1. What are the most important parameters for developing an irrigation support system?
2. How can an irrigation support system benefit from geographical data/tools as well as remote sensing products?
3. How/which meteorological parameters can be implemented in an optimal irrigation management system?
4. How can we propose an optimal irrigation management system based on remote sensing and geographical data?

## 2. Study Area

Denmark is a Nordic country and it is part of the geographic region called Scandinavia (Figure 18). Denmark is the southernmost of the Scandinavian nations, it is south-west of Sweden and south of Norway, and bordered to the south by Germany (Michael Falch (Danmarks Statistik) 2010).

Denmark consists of a peninsula, Jutland, and islands. The larger islands are Zealand, Funen, Lolland-Falster and 429 other islands between the North Sea and the Baltic Sea, 72 of which are inhabited. The islands are characterized by flat, arable land and sandy coasts, low elevation and a temperate climate. Denmark has a total area of 42,934 km<sup>2</sup> and a population of 5.71 million (2016). The population density of Denmark is 132 per square kilometer (Michael Falch (Danmarks Statistik) 2010).



*Figure 18: Denmark's location.*

### 3. Methodology

The purpose of this project was to estimate the amount of water that farmers need to irrigate in every field. The aim of this chapter is to clarify the procedure followed to reach the desired result. As can be seen from the diagram (Figure 19), six steps took place in order to find the process to the final results. Each step will be commented with more details in the following chapter.

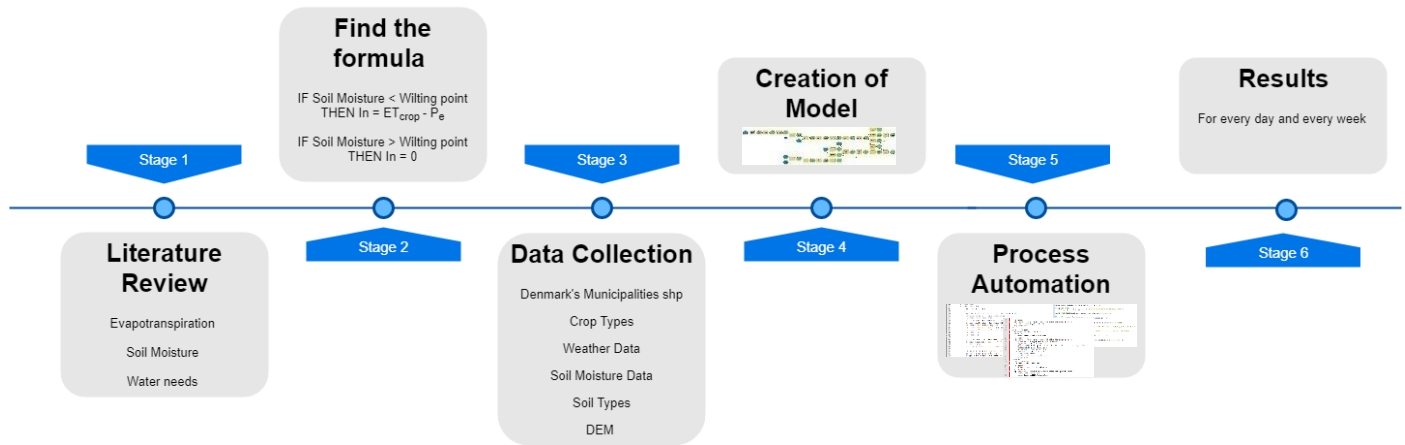


Figure 19: Workflow diagram.

Three methods were used in this project. For answering the first three research questions (see chapter 1.3.) the literature review method was used. For the last research question first, the requirement analysis method and final the prototype design method was used.

The most important step was the literature review stage, which is described with more details in Chapter 1.1. After reading and understanding all the necessary parameters for the water needs of the crops, the next step was to find the formula to calculate the irrigation needs for each crop. The 3<sup>rd</sup> stage was to collect the required data, which was used to deliver the final result. At this stage, lots of complicate procedures took place as some data had to be processed before running the final scripts and other data was used directly from the model. In this step, scripts were also created to download the data. After that, the model was created and soon after the process had to automated. This step made easy to have results for all the growing period of the crops in 2018 and led to the results. The final result consists of information about the amount of water that the farmers need to irrigate the specific date, in a specific area and the amount was the daily or weekly need.

#### 3.1. Stage 1 – Literature Review

As already mentioned above, this step is described in the Chapter of Background. In this stage, definitions such as evapotranspiration, water needs, and soil moisture were clarified. This step was important to understand which parameters can be used to calculate the irrigation requirements (see Chapter 1.1.3.2. Irrigation requirements). According to this Chapter, the irrigation requirements can be estimated calculating the losses of water (ET) and the gains of water (Soil Moisture and Precipitation) for the crops. The following figure (Figure 20) gives an overview of how the above-mentioned variables can be estimated and in some cases from which other parameters can be estimated.

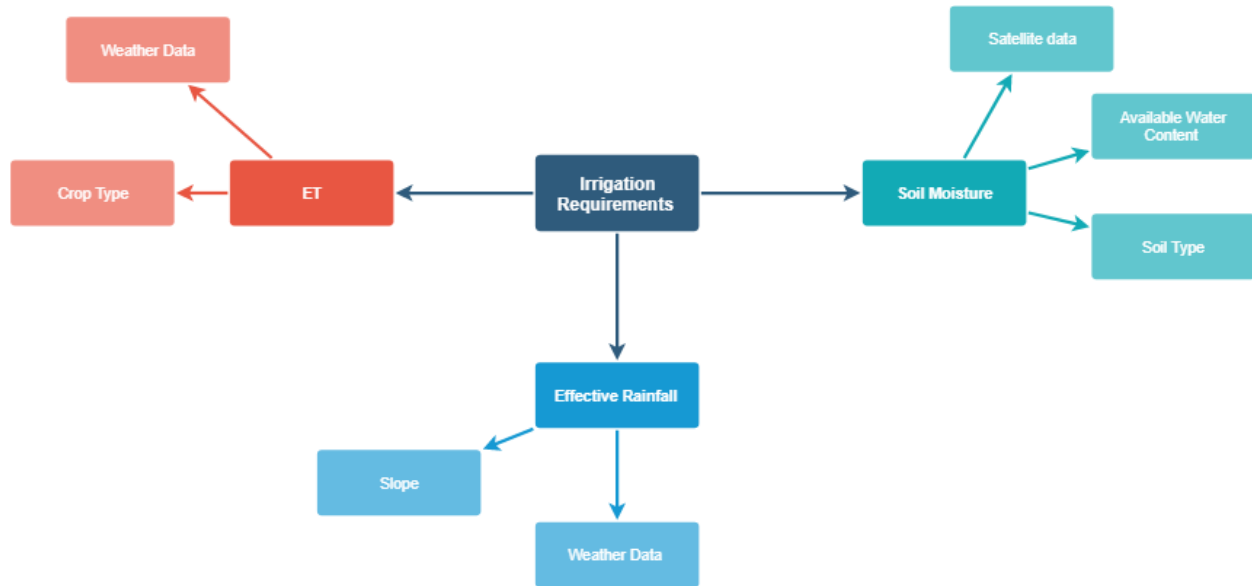


Figure 20: An overview of the variables, which can be used to estimate the irrigation requirements.

Evapotranspiration can be estimated with different ways according to the Chapter 1.1.1. For the needs of this project a straightforward method was used, which is described in Chapter 1.1.3.1. Crop water need is evapotranspiration and can be calculated by using the reference type of crop according to FAO which is the grass. According to the following table (Table 5), in Denmark, which is in Semi-arid zone, grass's average daily ET is 4 – 5 mm, when the mean temperature is less than 15 °C. ET is 6 – 7 mm, when the mean daily temperature is from 15 °C till 25 °C, and 8 – 9 mm when the mean temperature is more than 25 °C.

Table 5: Average daily water need of standard grass during the irrigation season (Brouwer and Heibloem 1986).

Climatic zone	Mean daily temperature		
	low	medium	high
	(less than 15°C)	(15-25°C)	(more than 25°C)
Desert/arid	4-6	7-8	9-10
Semi arid	4-5	6-7	8-9
Sub-humid	3-4	5-6	7-8
Humid	1-2	3-4	5-6

At this point is known how to calculate ET for grass. The ET for each crop can be estimated using the next table (Table 6), where for each type of crop a percentage is added or removed from the ET of the grass. There are crops like carrots that have the same ET rate with grass. Another example is potatoes, which they evapotranspire 10% more than grass. For example, in Denmark, in a day with a mean daily temperature of 16 °C, ET for potatoes will be  $6.5 * 110\% = 7.15$  mm/day. For the same place and day, ET for carrots will be 6.5 mm/day.

Table 6: Crop water needs in the peak period of various field crops as compared to standard grass (Brouwer and Heibloem 1986).

Column 1	Column 2	Column 3	Column 4	Column 5
-30%	-10%	same as standard grass	+ 10%	+20%
citrus	cucumber	carrots	barley	paddy rice
olives	radishes	crucifers (cabbage, cauliflower, broccoli, etc.)	beans	sugarcane
grapes	squash	lettuce	maize	banana
		melons	flax	nuts & fruit trees with cover crop
		onions	small grains	
		peanuts	cotton	
		peppers	tomato	
		spinach	eggplant	
		tea	lentils	
		grass	millet	
		cacao	oats	
		coffee	peas	
		clean cultivated nuts & fruit trees e.g. apples	potatoes	
			safflower	
			sorghum	
			soybeans	
			sugarbeet	
			sunflower	
			tobacco	
			wheat	

For the calculation of ET the mean temperature daily values and the type of the crops have to be known.

Moving forward, Soil Moisture is the next variable which can be used to estimate the irrigation requirements. Soil Moisture satellite data is freely available through several websites on the web. In each pixel exists the information about the soil moisture content. In the following picture (Figure 21) for a specific area, the soil moisture content can be seen on the left. The number of 0.03003 means that 1 m<sup>3</sup> of soil, with a depth of 1 m, and a surface area of 1 m<sup>2</sup> contains 0.03003 m<sup>3</sup> of water. The percentage of the moisture content volume for this specific example is 3.003 %.

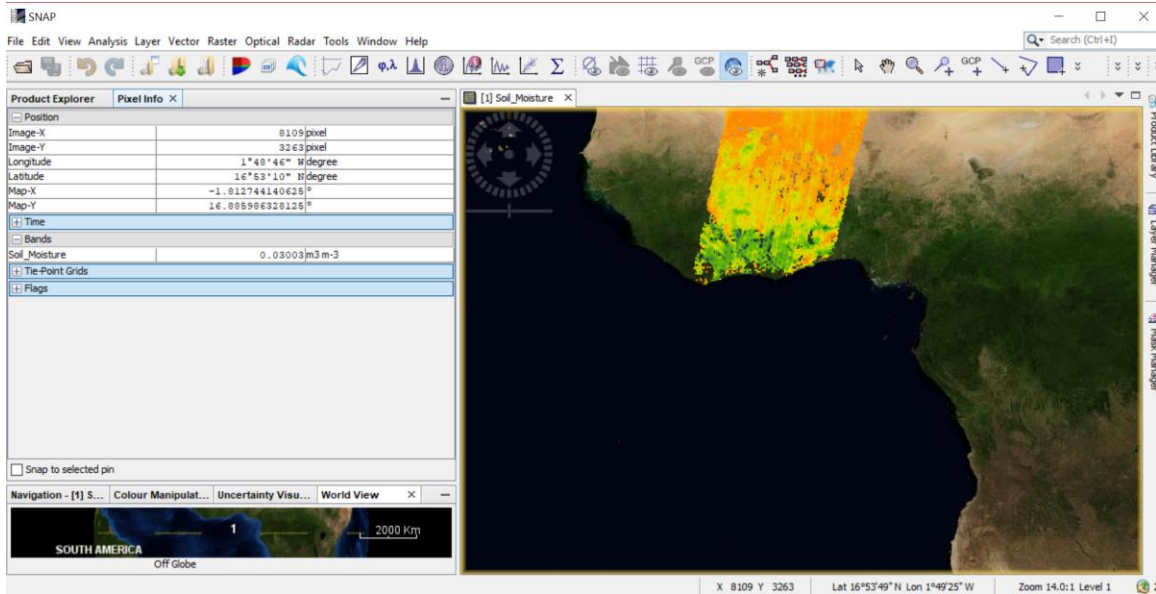


Figure 21: An example of Soil Moisture satellite image.

According to Chapter 1.1.2.1. Available water content soil can be compared to a water reservoir for the plants but the available water for the plants depends greatly on the soil texture and structure. The following table (Table 7) can clarify which percentage of soil moisture in which types of soils can provide water to a plant. Percentages higher than the field capacity number means that the soil is oversaturated, percentages above the number of wilting point means that the soil has no available water for the plants, while percentages in between the two numbers mean that the plants have available water. For example, in an area with sandy soil if the percentage of the moisture content volume is 6% it means that there is available water for the plants, while the same percentage in an area with clay soil, means that there is no available water for the plants.

Table 7: Field Capacity, Wilting Point vs. Soil Texture (Spruceirrigation.com 2019).

Soil Type	Permanent Wilting Point	Field Capacity
Sand	5% VWC	10% VWC
Loamy Sand	5% VWC	12% VWC
Sandy Loam	8% VWC	18% VWC
Sandy Clay Loam	17% VWC	27% VWC
Loam	14% VWC	28% VWC
Sandy Clay	25% VWC	36% VWC
Silt Loam	11% VWC	31% VWC
Silt	6% VWC	30% VWC
Clay Loam	22% VWC	36% VWC
Silty Clay Loam	22% VWC	38% VWC
Silty Clay	27% VWC	41% VWC
Silt	30% VWC	42% VWC

In order to use Soil Moisture as a variable to calculate the irrigation requirements, the percentage of the moisture content volume and the soil type of the area must be known.

Finally, the final variable which can be used to estimate the irrigation requirements is effective rainfall. According to Chapter 1.1.3.3. Effective Rainfall part of the rainfall is not effective; therefore, the plants cannot use all the amount of precipitation. Moreover, FAO introduced 2 simple formulas estimate the fraction of the total rainfall which is used effectively. These formulas can be applied in areas with a maximum slope of 4-5 %. Denmark is a country with a relatively low percentage of slopes therefore, these formulas can be easily used in this project. Using a Digital Elevation Model (DEM) of Denmark, the percentage of the slopes can be estimated. Places with higher values than 5% of the slope can be excluded from the procedure.

As can be seen from the following equation (Equation 11), in order to calculate the effective rainfall, the amount of precipitation from the previous 30 days must be known. In case that the amount of precipitation of the previous month was more than 75 mm the first equation will be used, in another case, the second one will be used.

$$\begin{aligned} P_e &= 0.8 P - 25 \text{ if } P > 75 \text{ mm/month} \\ P_e &= 0.6 P - 10 \text{ if } P < 75 \text{ mm/month} \end{aligned} \quad (11)$$

*Equation 11: Formulas to estimate effective rainfall.*

In order to calculate the irrigation requirements considering also the rainfall, a DEM and precipitation measurements from the area should be known.

### 3.2. Stage 2 – Finding the Formula

The next step was to find the formula in order to find the amount of irrigated water. This has to be estimated considering the evapotranspiration rate of the crop, the amount of the effective precipitation and the soil moisture percentage. The simplest formula can be seen in the following equation (Equation 12)

$$\begin{aligned} \text{IF Soil Moisture} < \text{Wilting point THEN } I_n &= ET_{\text{crop}} - P_e \\ \text{IF Soil Moisture} > \text{Wilting point THEN } I_n &= 0 \end{aligned} \quad (12)$$

*Equation 12: Simplest formula to find the irrigation needs.*

With this formula first a check is made if the soil moisture level is below the wilting point which means that there is no available water for the plants in the soil. If the level is below the wilting point, then the irrigation requirements of the crops ( $I_n$ ) is calculated by estimating the evapotranspiration rate of the specific type of crop minus the effective rainfall ( $P_e$ ). If the soil moisture level is above the wilting point, it means that there is available water for the plants with consequence no irrigation needs. At this point, for checking the soil moisture content if it has available water for the plants, the percentage of the moisture content volume and the soil type is considered.

### 3.3. Stage 3 – Data Collection and Process

Following the 2<sup>nd</sup> stage, was the stage of data collection. At this point, the formula of the estimation of irrigation requirements is known. The next step was to find the data and, in some cases, process the data in order to have it in the desired format. The purpose of this chapter is to clarify all the procedures that

took place from the collection and process of the primary data until the data, that was used from the model.

As mentioned above (4.1. Stage 1 – Literature Review), in order to use Soil Moisture as a variable to calculate the irrigation requirements, the percentage of the moisture content volume and the soil type of the area must be known. For the calculation of ET, the mean temperature daily values and the type of the crops have to be known and considering also the rainfall, a DEM and precipitation measurements from the area should be known.

This chapter will be divided into three subchapters, for better understating of the procedures that took place. The first subchapter describes the process made to obtain the data for soil moisture variable, the following one, refers to the ET and the final refers to the rainfall.

### 3.3.1. Data Collection and Process of Soil Moisture

To obtain the remotely sensed data for soil moisture was proved to be more difficult than expected. There were problems at downloading or processing the data while in some cases the data was not free. For this project, only free available data was used with consequence some providers, could not be used. ESA was one of the candidate organizations for obtaining the data using data from Soil Moisture Ocean Salinity (SMOS) Earth Explorer mission and another candidate was NASA, using SMAP (Soil Moisture Active Passive) data. Even though SMAP data had a better spatial resolution, there were lots of problems with processing the data and because of lack of time, it was decided to use SMOS mission and data from ESA.

The primary objective of the SMOS mission is to provide global observations of soil moisture over land surfaces and ocean salinity over the oceans. SMOS was launched on 2<sup>nd</sup> of November of 2009 and the mission requirements for soil moisture is an accuracy of 4% volumetric soil moisture, spatial resolution of 35-50 km with a revisit time 1-3 days (ESA 2019).

For downloading SMOS data, FileZilla was used. FileZilla is a free software, cross-platform FTP application which after signing in with username and password, users can search and download data from servers. The server is located in smos-diss.eo.esa.int location and the access is free.

There are folders with data for every day, every month and every year. The acquisition of the satellite for Denmark takes place almost every day from 16:00 till 17:00 depending on the day of the month. All images for the period from March till July with start acquisition time from 16:00 till 17:00 were downloaded. There were some images without data and others without Denmark inside the scene. These images were not a problem as the model for finding the irrigation needs can run using them but will not deliver any results. An example of SMOS data can be seen in the following picture (Figure 22). The image is from the 2<sup>nd</sup> of July in 2018.



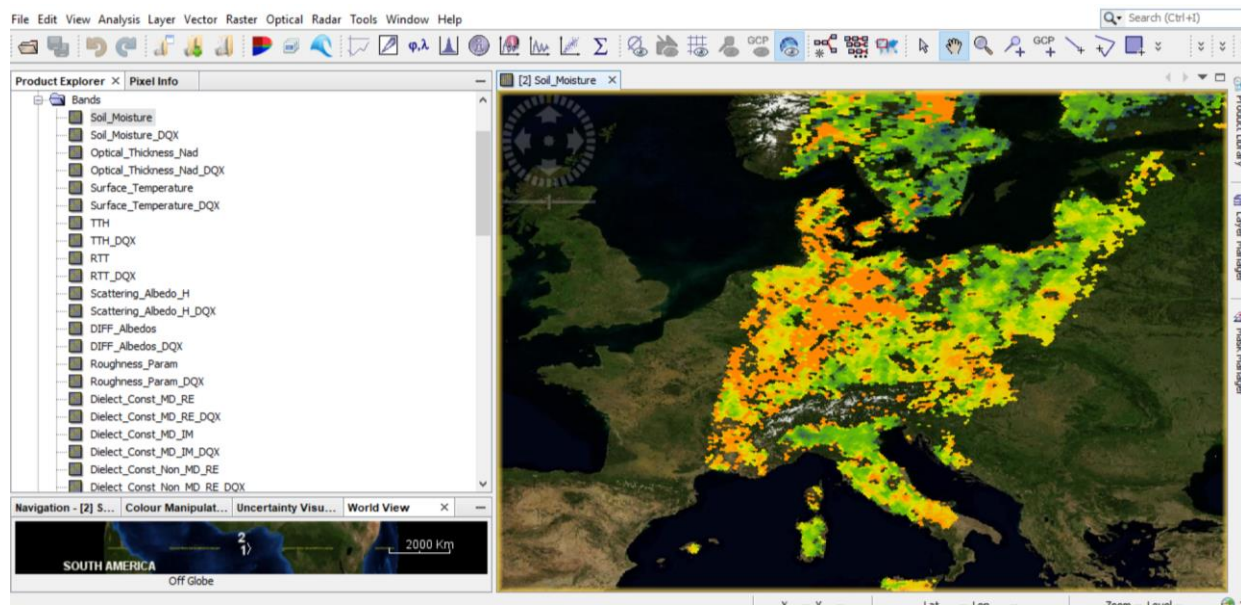


Figure 22: Example of SMOS data.

The next step was to subset the image in order to have only Denmark and extract only the band which consist of information about the soil moisture. This was made by using SNAP (Sentinel Application Platform). SNAP architecture with toolboxes is ideal to process and analyze Earth Observation data and it can be freely downloaded from the ESA website.

The process to subset an image is simple, and it can be seen from the diagram (Figure 23). The first step was to read the image which the subset will take place, then in the subset step the band of interest (in this case the soil moisture band only) and the polygon which consists Denmark (in this case the polygon is 7 54, 13 54, 13 58, 7 58) were defined. The final step was to create the new image and save it as a GeoTIFF file.



Figure 23: Graph for the subset of an image.

The use of Graph Builder in SNAP for the subset is very slow while it cannot be used for automation. Therefore, it was essential to create a .xml file which can make the process quicker and to use this file in the following stage of automation of the process. This .xml file can be run using snap gpt.exe in the command line. The first step was to create the .xml file (Figure 24) where for Read, Subset and Write there was a node with a specific id.

```

1 <graph id="Graph">
2   <version>1.0</version>
3   <node id="Read">
4     <operator>Read</operator>
5     <sources/>
6     <parameters class="com.bc.ceres.binding.dom.XppDomElement">
7       <file>${product_dbl}</file>
8     </parameters>
9   </node>
10  <node id="Subset">
11    <operator>Subset</operator>
12    <sources>
13      <sourceProduct refid="Read"/>
14    </sources>
15    <parameters class="com.bc.ceres.binding.dom.XppDomElement">
16      <sourceBands>Soil_Moisture</sourceBands>
17      <region>0,0,3270,1201</region>
18      <geoRegion>POLYGON ((7 54, 13 54, 13 58, 7 58, 7 54, 7 54))</geoRegion>
19      <subSamplingX>1</subSamplingX>
20      <subSamplingY>1</subSamplingY>
21      <fullSwath>false</fullSwath>
22      <tiePointGridNames/>
23      <copyMetadata>true</copyMetadata>
24    </parameters>
25  </node>
26  <node id="Write">
27    <operator>Write</operator>
28    <sources>
29      <sourceProduct refid="Subset"/>
30    </sources>
31    <parameters class="com.bc.ceres.binding.dom.XppDomElement">
32      <file>${tif_output}</file>
33      <formatName>GeoTIFF</formatName>
34    </parameters>
35  </node>
36  <applicationData id="Presentation">
37    <Description/>
38    <node id="Read">
39      <displayPosition x="37.0" y="134.0"/>
40    </node>
41    <node id="Subset">
42      <displayPosition x="242.0" y="136.0"/>
43    </node>
44    <node id="Write">
45      <displayPosition x="332.0" y="259.0"/>
46    </node>
47  </applicationData>
48 </graph>

```

Figure 24: The .xml file to subset the image.

Inside the node of Read the name of the file with the complete path is defined in line 7. After that, inside the node with the Subset id in line 10, the Soil Moisture band (line 11) and the coordinates of the polygon are defined (line 18). The name of the band is in the node called sourceBands and the polygon in the node with name geoRegion. The final step is to define the name and format of the new file. The format of the new file was GeoTIFF. The .xml file can be run from the command line using SNAP gpt.exe and defining the complete path of the .xml file. The commands can be seen in the following picture (Figure 25).

```

Microsoft Windows [Version 10.0.17134.765]
(c) 2018 Microsoft Corporation. All rights reserved.

C:\Users\Giorgos>cd C:\Program Files\snap\bin

C:\Program Files\snap\bin>gpt C:\Data\final_scripts\Subset_from_whole_orbit_smos.xml

```

Figure 25: Commands to run the subset .xml file.

The second crucial dataset needed to use Soil Moisture as a variable to calculate the irrigation requirements was the soil type of the area. The dataset was downloaded from the European Commission and more specific from the European Soil Data Centre (ESDAC) which is the thematic center for soil-related data in Europe. It was compulsory to ask for permission in order to use this dataset. The request was quickly accepted from ESDAC and the data of soil types in Europe was downloaded and it can be seen in the next picture (Figure 26).

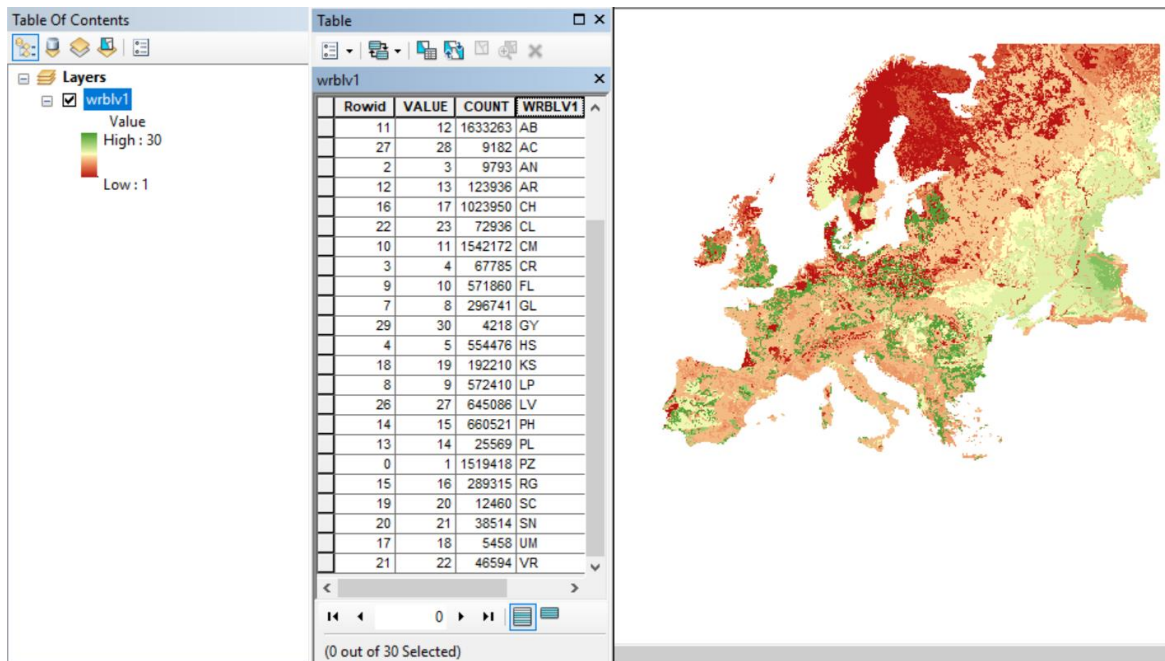


Figure 26: The soil types in Europe (Panagos 2006), (Liedekerke, Jones, and Panagos 2006).

The resolution of the data is 1km<sup>2</sup> and the information refers only in Europe. The next step was to create a new raster file consists information for soil types only for Denmark. Using a shapefile for Denmark which was downloaded from kortforsyningen.dk website and running a mask tool the new raster dataset was produced (Figure 27).

## Soil Types in Denmark

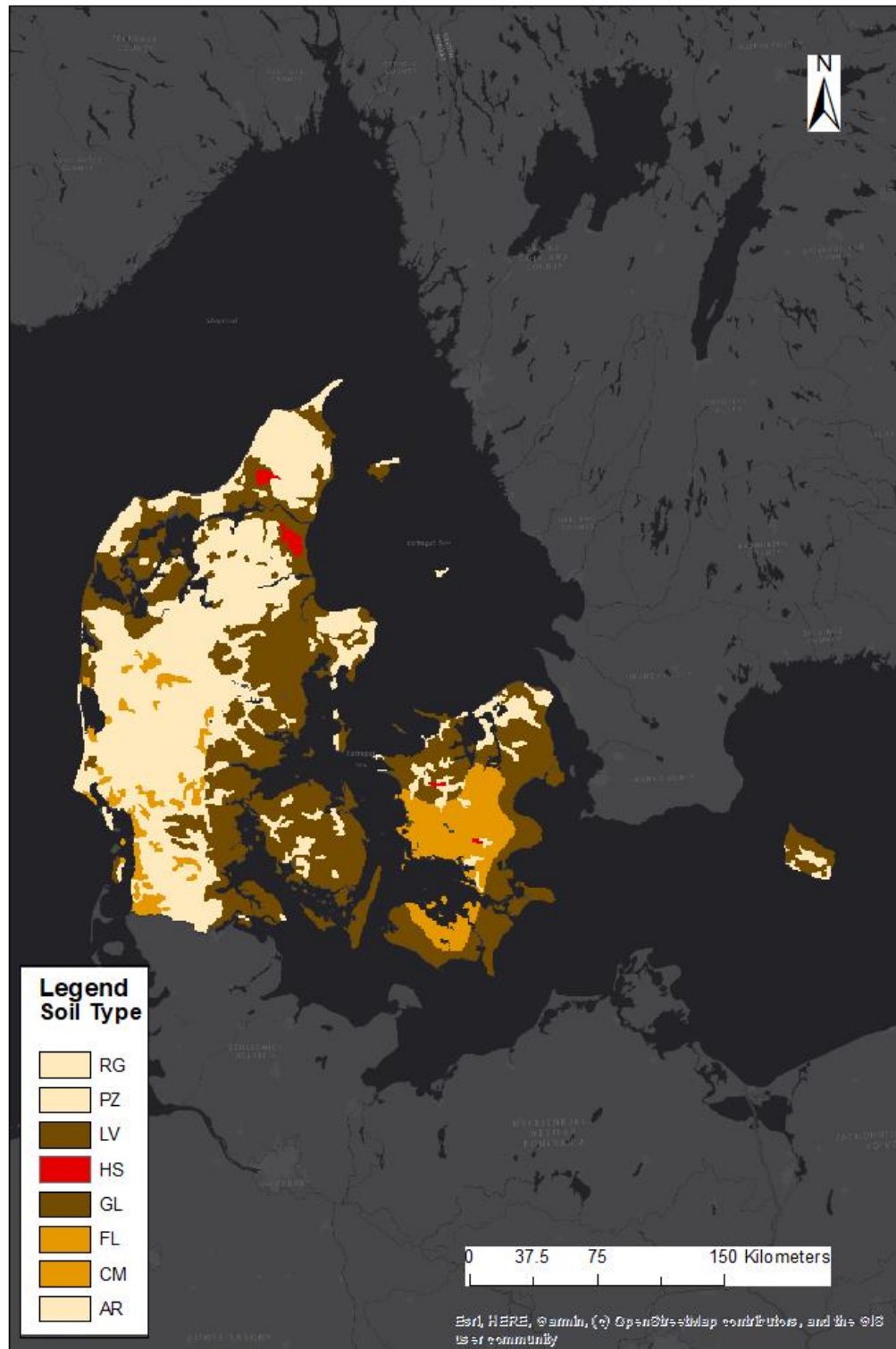


Figure 27: Map of Denmark's soil types.

After having the dataset with the soil types the next step was to categorize the types in clay, loam and sandy soils. This step was necessary for using the table in Figure 28 which presents the Field Capacity and Wilting Point vs. Soil Texture. For finding the categories for each soil type in Denmark, World reference base for soil resources (WRB) from FAO was used. The full name, some more information and the category of the soil type are presented in the next table (Table 8).

*Table 8: Soil type categories in Denmark according to World reference base for soil resources (WRB).*

<b>Soil Type Code</b>	<b>Soil Type Name</b>	<b>Information</b>	<b>Category</b>
<b>RG</b>	Regosol	A Regosol in the World Reference Base for Soil Resources (WRB) is very weakly developed mineral soil in unconsolidated materials (FAO 2015).	Sandy
<b>PZ</b>	Podzol	Podzols are the typical soils of coniferous or boreal forests (FAO 2015).	Sandy
<b>LV</b>	Luvisol	Luvisols are widespread, especially in temperate climates, and are generally fertile. Luvisols are widely used for agriculture (FAO 2015).	Clay
<b>HS</b>	Histosol	The central concept of Histosols is that of soils that are dominantly organic (FAO 2015).	Sandy
<b>GL</b>	Gleysol	Gley is a wetland soil that, unless drained, is saturated with groundwater (FAO 2015).	Clay
<b>FL</b>	Fluvisol	Fluvisol in the World Reference Base for Soil Resources (WRB) is a genetically young soil in alluvial deposits (FAO 2015).	Loam
<b>CM</b>	Cambisol	Cambisol is a soil with a beginning of soil formation. Cambisols are developed in medium and fine-textured materials derived from a wide range of rocks, mostly in alluvial, colluvial and aeolian deposits. Most of these soils make good agricultural land and are intensively used (FAO 2015).	Loam
<b>AR</b>	Arenosol	Arenosol consists basically of unconsolidated sand deposits, often found in shifting sand dunes (FAO 2015).	Sandy

The final step for the soil type dataset was to reclassify the raster file so that each pixel consists the information about the soil type category (Sandy, Clay or Loam) and to reproject the dataset. The value 1 was assigned for sandy soils, 2 for loam type and 3 for clay. The projection used for all datasets was the WGS 84 / UTM zone 32N.

### 3.3.2. Data Collection and Process for ET

As already mentioned above in chapter 3.1. Stage 1 – Literature Review for the calculation of ET the mean temperature daily values and the type of the crops have to be known. The mean temperature daily values were obtained from the archive of weather data from Danish Meteorology Institute (DMI) and the type of the crops was obtained from the Ministry of Environment and Food of Denmark (Miljø og Fødevareministeriet) website.

More specific, for the weather data, the website <https://www.dmi.dk/vejarkiv/> (Figure 28) allows the users to download weather data for Denmark, Greenland and Faroe Islands for every municipality from

2012 till the current date. The data can be yearly, monthly, daily or hourly and it can be downloaded as a diagram or .csv file. Moreover, among temperature and precipitation, there are 8 more measurements which can be downloaded.

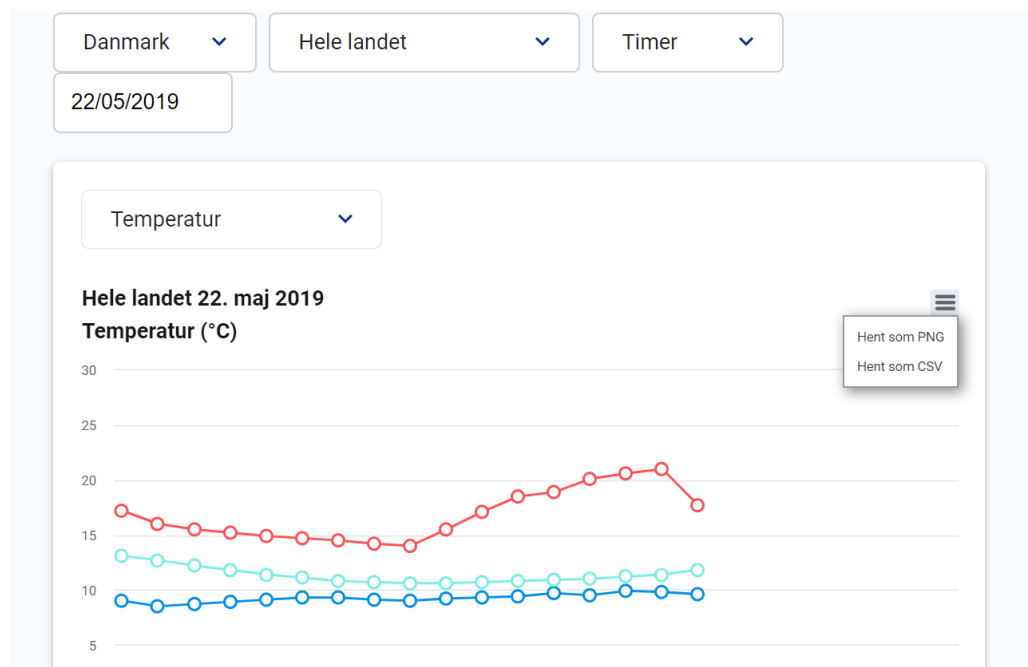


Figure 28: DMI weather data archive website (DMI (Danish Meteorology Institute) 2019).

The next step was to create a script using Java in order to download the data from the website. The script was provided from Sandholt ApS and the data was downloaded. The downloaded data was for Denmark, for all municipalities, for all weather measurements, for every year and every day. The format and the structure of the downloaded data can be seen in the following image (Figure 29). There was a folder for every municipality consisting 10 folders, one for each weather parameter. Inside each weather parameter folder, there were .txt files one for every month of the year. The .txt files which include information for every day of the month were in GeoJSON format.



Municipalities	Weather parameters	Months	GeoJSON files
→	→	→	
<ul style="list-style-type: none"> <li>Aabenraa</li> <li>Aalborg</li> <li>Aarhus</li> <li>Ærø</li> <li>Albertslund</li> <li>Allerød</li> <li>Assens</li> <li>Ballerup</li> <li>Billund</li> <li>Bornholm</li> <li>Brøndby</li> <li>Brønderslev</li> </ul>	<ul style="list-style-type: none"> <li>drought</li> <li>humidity</li> <li>lightning</li> <li>precip</li> <li>pressure</li> <li>snow</li> <li>sunhours</li> <li>temperature</li> <li>wind</li> <li>winddir</li> </ul>	<ul style="list-style-type: none"> <li>2017-10.txt</li> <li>2017-11.txt</li> <li>2017-12.txt</li> <li>2018-01.txt</li> <li>2018-02.txt</li> <li>2018-03.txt</li> <li>2018-04.txt</li> <li>2018-05.txt</li> <li>2018-06.txt</li> <li>2018-07.txt</li> <li>2018-08.txt</li> <li>2018-09.txt</li> <li>2018-10.txt</li> <li>2018-11.txt</li> <li>2018-12.txt</li> <li>2019-01.txt</li> </ul>	<pre>[   {     "parameter": "Dags middeltemperatur",     "unit": "°C",     "area": "Aalborg",     "dataserie": [       {         "time": 1301616000000,         "value": 7.83151912689209,         "normal": null,         "parameterNumber": 101,         "valuesUsed": 0,         "timezone": "Europe/Copenhagen",         "yearString": "2011",         "dateString": "01-04-2011 00:00",         "valueOneDecimal": "7,8",         "dateLocalString": "01-04-2011 02:00",         "yearLocalString": "2011",         "monthString": "Apr",         "monthLocalString": "Apr",         "dayString": "1",         "dayLocalString": "1",         "hourString": "00",         "hourLocalString": "02",         "valueRounded2OneDecimal": 7.8       }     ],     {       "time": 1301702400000,       "value": 12.174365043640137, </pre>

Figure 29: Format and structure of DMI weather data.

The next step, in order to save the weather data, was to create a database. The database software used was MySQL Workbench which is a unified visual tool for database architects and developers (MySQL 2019) and it is an open source software. The structure of the database can be seen in the following UML (Unified Modeling Language) diagram (Figure 30).

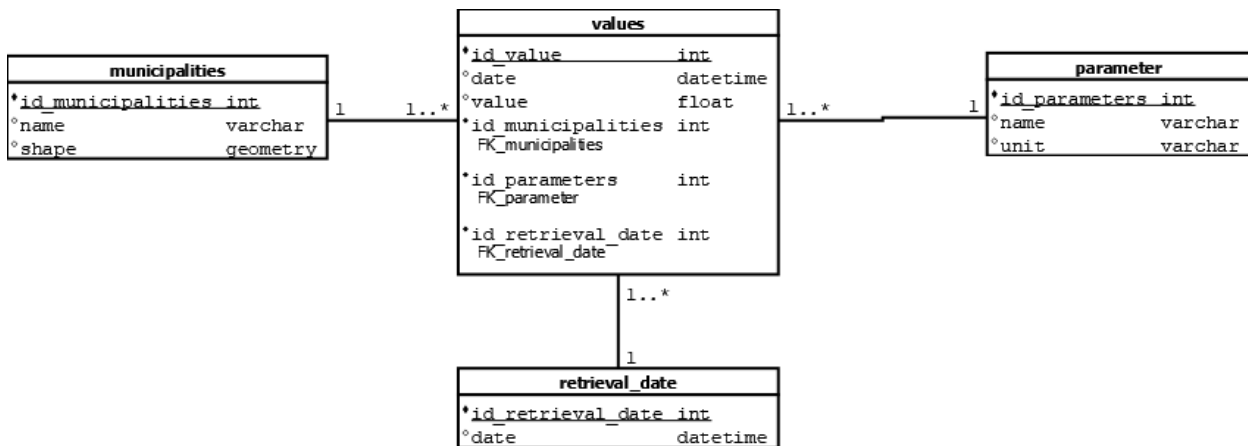


Figure 30: UML diagram of the weather data archive database.

More specifically, the database has four entities. The table of municipalities contains a unique number attribute called `id_municipalities` which is the primary key of the table, the name, and the shape attribute. The shape attribute contains the geometry information of each municipality in Denmark. The parameter table has the attribute `id_parameters`, which is a unique number and primary key. The other attributes of this table are the name and the unit. The retrieval\_date entity has two attributes. The first is the `id_retrieval_date`, which is the primary key of this table and unique number, and the date attribute with datetime format. Lastly, there is the values entity. In this table, the attribute `id_value` is the primary key but there are also foreign keys. The `id_municipalities` attribute refers to municipalities table, the

id\_parameters attribute refers to parameters table and the id\_retrieval\_date attribute refers to retrieval\_date table. The last two attributes are the date, which is datetime type, and the value, which is in float format.

As for the cardinalities of the tables, the relationships are all one to many. More specific, municipalities and value table have a one to many cardinality. This means that every entity in the municipalities table can be associated to one or a lot of occurrences in the values table but every entity in the values table is linked to only one occurrence in the municipalities table. Similarly, parameter and value table have a one to many cardinality. This means that every entity in the parameter table can be associated to one or a lot of occurrences in the values table but every entity in the values table is linked to only one occurrence in the parameter table. Finally, the degree of relationship between retrieval\_date and values table, is one to many. This means that every entity in the retrieval\_date table can be associated to one or more occurrence in the values table and every entity in the values table is linked to only one occurrence in the retrieval\_date table.

The next stage after creating the database was to store the data from the .txt files inside the database. In this step, a script was created from Sandholt ApS using Java. The aim of this script was to read all the GeoJSON files and store the information about the weather parameters inside the database. After running the Java file, the database was populated with the values of all the weather parameters, for all the municipalities, for every day from January of 2012 till March of 2019.

At this point, the mean temperature daily values, which are needed to estimate the ET of the crops, are stored in the database and they can be exported as a table. But not only mean temperature values are needed to calculate ET. The type of crops is also crucial information. For retrieving this data the <http://miljoegis.mim.dk/cbkort?profile=jordbrugsanalyse> website was used. An effort made to download one shapefile consisting the information about the type of the crops in every field in Denmark for 2018. In this shapefile, there were more than 600.000 polygons, which could give difficulties during the process. There was no option to download one shapefile for all Denmark, the shapefile was divided by 4 areas. After that, the 4 areas were combined in one shapefile (Figure 31).



## Fields in Denmark in 2018

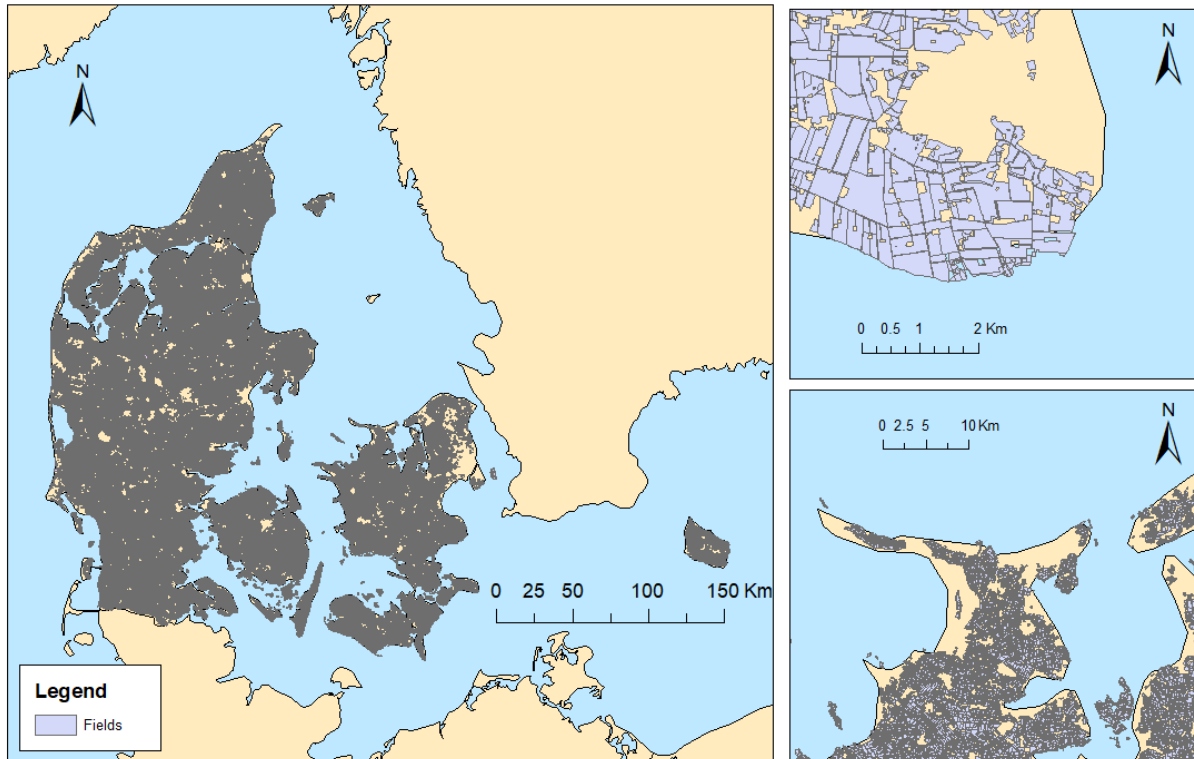


Figure 31: Shapefile of fields in Denmark in 2018 (Miljø, Og, and Fødevareministeriet 2019).

After that, a very important decision was made. It was decided to use the data for only one municipality as the process for all dataset of the fields for entire Denmark would need more time and computation power. Viborg municipality was chosen as a test area because Sandholt ApS has an experimental site in this area. The next step was to create a new shapefile with all the fields located in Viborg municipality. For every field, there is information about the type of the crop as it can be seen from the following image (Figure 32). The information about the crop type was in Danish and in this selected field “gulerød” means carrot.

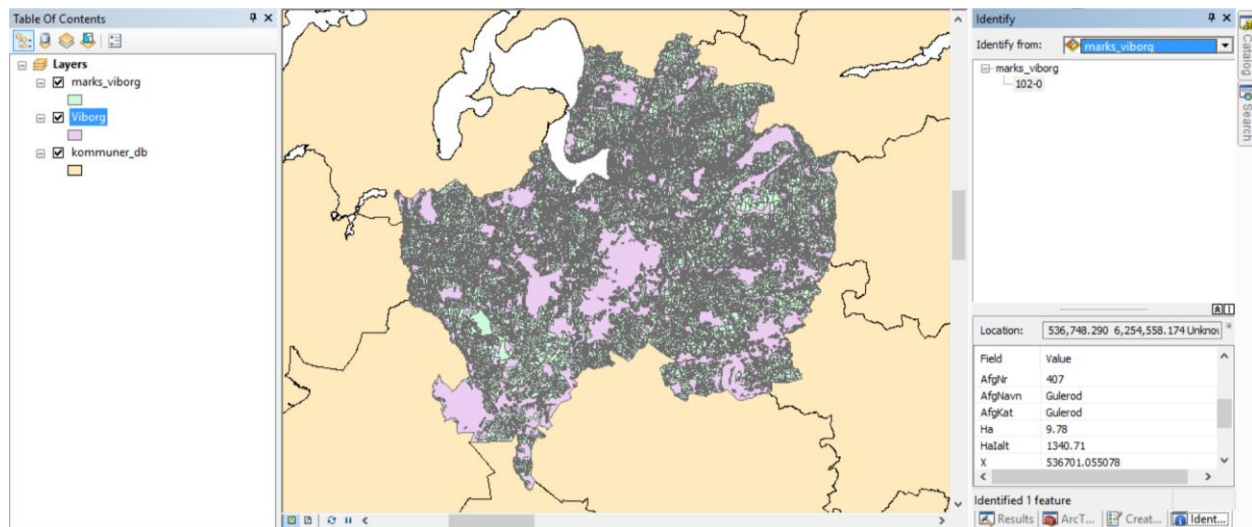


Figure 32: Shapefile presenting marks in Viborg (Miljø, Og, and Fødevarerministeriet 2019).

Thus, the data for the calculation of ET, which is the mean temperature daily values and the type of the crops, is retrieved.

### 3.3.3. Data Collection and Process for rainfall

As already mentioned above in chapter 3.1. Stage 1 – Literature Review for the calculation of the effective rainfall an equation can be used, but the precipitation measurements from the area should be known. Effective rainfall can be estimated with this equation in areas with less than 5% slope. In order to find the slope, a DEM of the area was downloaded. The precipitation values were obtained from the archive of weather data from DMI and the DEM of the area, from the earth explorer of United States Geological Survey website (<https://earthexplorer.usgs.gov/>) using data from the SRTM (Shuttle Radar Topography Mission).

In the previous Chapter (3.3.2. Data Collection and Process for ET) was described the process followed to download weather data from the weather archive of DMI. Precipitation data was already stored inside the database among other parameters. The next step was to extract the weather data, which was needed for the next stages. The data needed was the mean temperature and precipitation values. For this step, an SQL query was created, and it can be seen in the next picture (Figure 33).

```

1 • select * from
2 (select `value`.`date` AS `date`,`value`.`idkommuner` AS `idkommune`,
3 max((case when (`value`.`idparameter` = 101) then `value`.`value` end)) AS `MeanTemperature`,
4 max((case when (`value`.`idparameter` = 601) then `value`.`value` end)) AS `Precipitation`
5 from value group by `value`.`date`,`value`.`idkommuner`) a
6 where date >= '2018-02-01' and date <= '2018-07-31' and idkommune = 93;

```

date	idkommune	MeanTemperature	Precipitation
2018-02-01 00:00:00	93	2.5999999046325684	5.800000190734863
2018-02-02 00:00:00	93	-0.30000001192092896	1
2018-02-03 00:00:00	93	-2.700000047683716	0.20000000298023224
2018-02-04 00:00:00	93	-1.7999999523162842	0
2018-02-05 00:00:00	93	-3.200000047683716	0
2018-02-06 00:00:00	93	-4.400000095367432	0
2018-02-07 00:00:00	93	-0.6000000238418579	0.10000000149011612
2018-02-08 00:00:00	93	0.20000000298023224	0.10000000149011612
2018-02-09 00:00:00	93	-1.2999999523162842	0
2018-02-10 00:00:00	93	-0.8999999761581421	0
2018-02-11 00:00:00	93	1.100000023841858	5.400000095367432
2018-02-12 00:00:00	93	2	1.5
2018-02-13 00:00:00	93	0.800000011920929	0.4000000059604645
2018-02-14 00:00:00	93	0.5	0

Figure 33: SQL query to retrieve weather data measurements from the database.

The SQL query selects from the database the date and then the mean temperature values and precipitation using the id numbers of the parameters. Then, it takes only the measurements from Viborg municipality, which has the 93-id number and the measurements are for the period from February to July of 2018. February's measurements were used only for counting the amount of precipitation of the last month on 1<sup>st</sup> of March. These months were chosen for this study as they are the months consisting the period from seed to harvest for several crops in Denmark such as potatoes, carrots and corn. The following step was to export the table in .csv format.

After that, the next step was to add some more information in the specific file (Figure 9). The first field was the referenced ET for grass. Using Table 1 (Average daily water needs of standard grass during irrigation season (Brouwer and Heibloem 1986)) the column was populated with values 4.5 when the mean temperature value was below 15 °C, 6.5 when the mean daily temperature was from 15 °C till 25 °C, and 9.5 when the mean temperature of the day was more than 25 °C. The above-mentioned values were for every day thus, a second field for the weekly ET was added and populated with the number of the ET per day multiplied by 7. The next four columns refer to rainfall. To use the equation to find the amount of effective rainfall (Equation 11: Formulas for effective rainfall), the precipitation amount of the last 30 days is needed to be known. The mm\_month field is populated, adding all the precipitation values of the last 30 days. Then the effective monthly rainfall amount for each day is estimated using the above-mentioned formula. The weakness of this formula is that it can calculate the effective rainfall for a season of 30 days and not daily. At this point an assumption was made in order to find the weekly and daily amount of effective rainfall, that these values can be estimated by dividing the monthly amount by 4 for the weekly and by 30 for the daily amount of effective rainfall.

Table 9: Final weather data file for Viborg municipality including the ET of grass and effective rainfall.

	OID	date	idkommune	MeanTemper	Precipitat	ET_day	ET_week	mm_month	Ef_rain_m	Ef_rain_w	Ef_rain_d
	28	01-Mar-18	93	-6.2	0.1	4.5	31.5	25.2	5.12	1.28	0.182857
	29	02-Mar-18	93	-3.9	0	4.5	31.5	25.3	5.18	1.295	0.185
	30	03-Mar-18	93	-2.2	0	4.5	31.5	25.3	5.18	1.295	0.185
	31	04-Mar-18	93	-2.2	0	4.5	31.5	25.3	5.18	1.295	0.185
	32	05-Mar-18	93	-0.4	5.8	4.5	31.5	19.5	1.7	0.425	0.060714
	33	06-Mar-18	93	1.2	0.6	4.5	31.5	24.3	4.58	1.145	0.163571

The final step to calculate the effective rainfall after having the data of the amount of precipitation of the last month was to calculate the slope of Viborg municipality. The purpose of this step was to create a raster file, which will be used for masking out all the areas with slopes more than 5%. So, for calculating the slope, the DEM and a tool to create the slop raster file was used. The final file was clipped for having only Viborg municipality and the results can be seen in the following map (Figure 34). The green areas can be used in the calculations for the estimation of irrigation requirements, but red areas should be excluded as they are areas with higher than 5% values of slope. The last step was to reclassify the raster file and assign No Data values in areas with higher than 5% slope so that the file can be used inside the calculations as a clip file for the results.

Slope map of Viborg

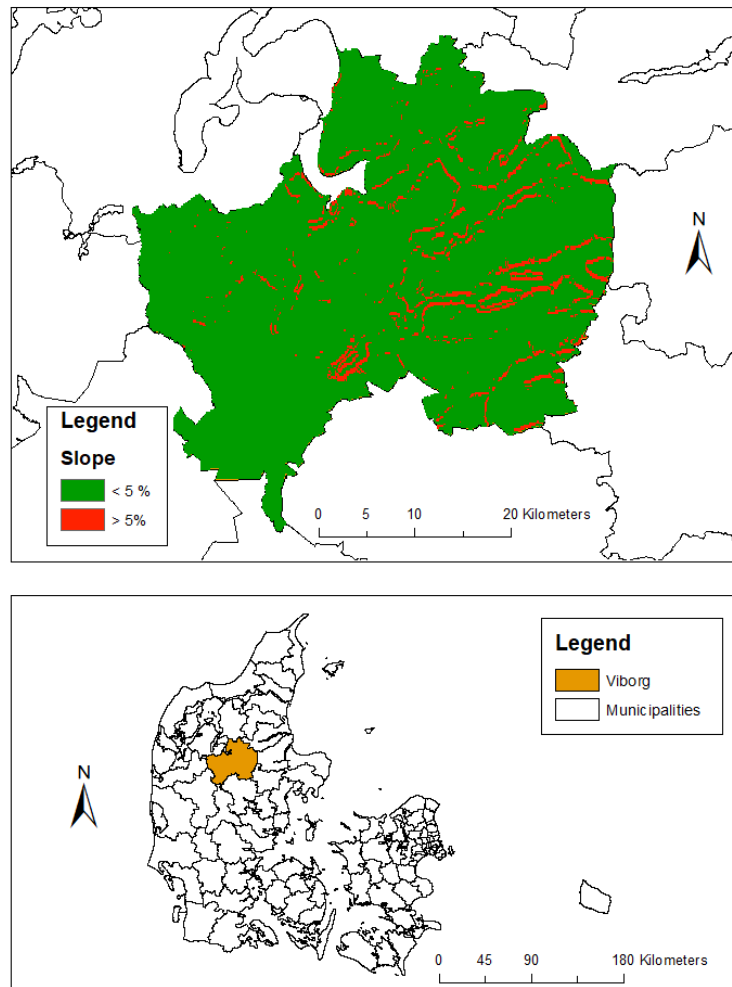


Figure 34: Slope map of Viborg.

### 3.4. Stage 4 – Model creation

In the 3<sup>rd</sup> stage, data was collected and processed. In the next stage, the processed data will be used directly from the model. The entire model is presented using Arc Map model builder in the next picture (Figure 35). For clarifying all the intermediate steps, the model is divided into 5 smaller parts in this Chapter. The model will be commented by parts.

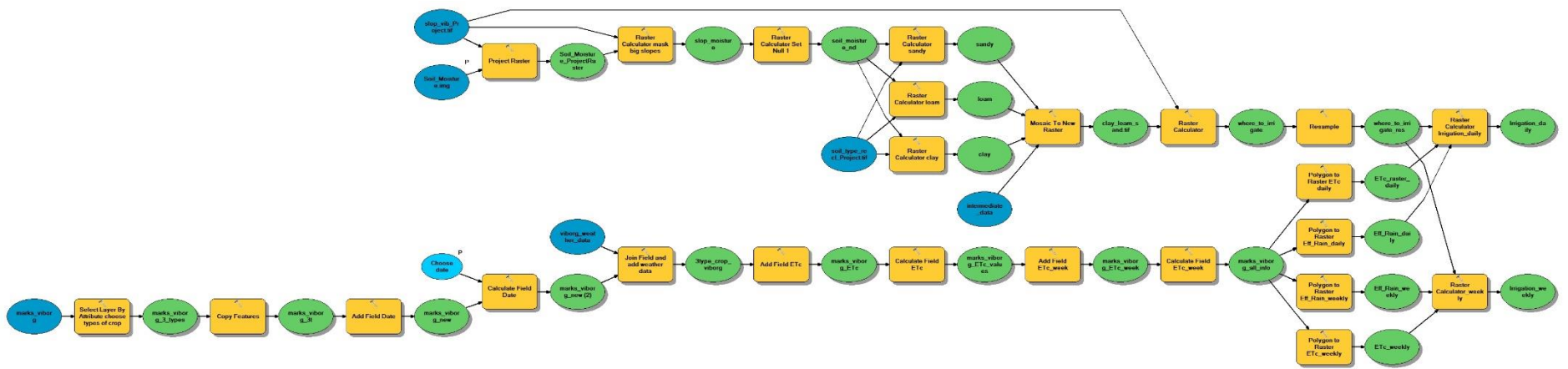


Figure 35: The final model using Arc Map model builder.

In Chapter 3.3.2. Data Collection and Process for ET a shapefile with marks only in Viborg was created. This shapefile contains information, among others, about the type of the crops in each field.

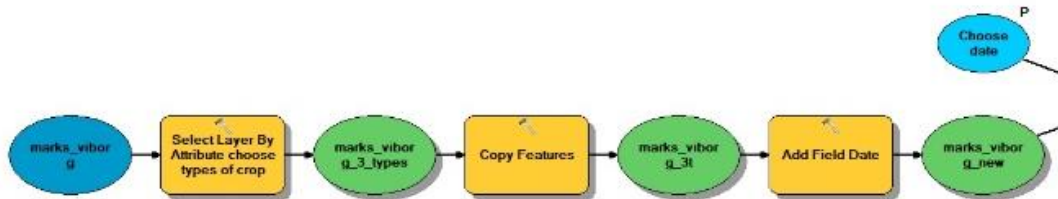


Figure 36: 1<sup>st</sup> part of the model.

The first part of the model (Figure 36) starts with the shapefile of the marks in Viborg. After that, a selection for 3 types of crops was made, choosing potatoes, carrots and corn. These types of crops were chosen as they are the most usually grown crops in the area. Then the selected features were copied and after that, a new field, called date, inside the attribute table was added. At this stage, the first parameter of the model which is the choice of the date was created. The date is depending on the date of the soil moisture image. For example, if the soil moisture image is from 03-06-2018, then the date should be 03-06-2018.

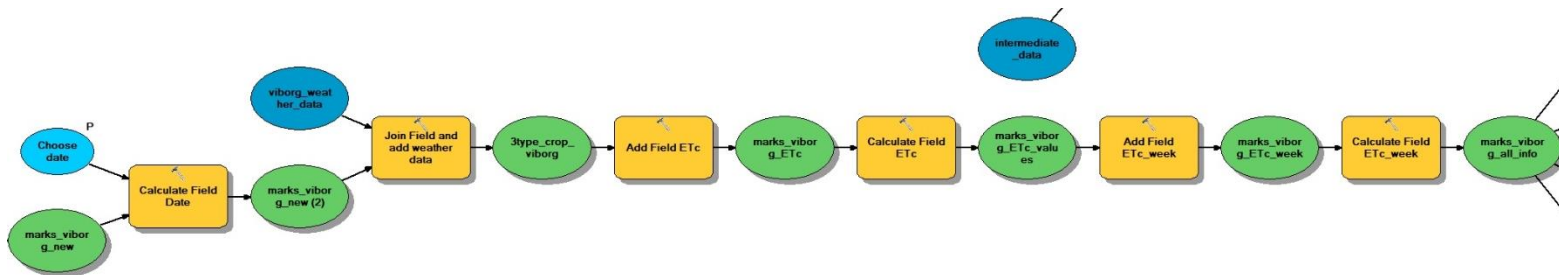


Figure 37: 2<sup>nd</sup> part of the model.

The next step was to populate the date field of the new file of the marks in Viborg with the chosen date (Figure 37). All the records of the attribute table had the same date. After that, the Viborg weather data .csv file was used, and the attribute table of the shapefile joined with the .csv table using the date field. The next step was to add the ETc field and populate the column by using a function (Figure 38).

```
def Reclass(ETc, AfgKat, ET_day ):
    if ( AfgKat == "Majs"):
        return ( ET_day * 1.1)
    elif ( AfgKat == "Kartoffel"):
        return ( ET_day * 1.1)
    elif ( AfgKat == "Gulerod"):
        return ( ET_day * 1)

Reclass(!ETc!, !AfgKat!, !ET_day! )
```

Figure 38: Function for populate ETc field.



This function estimates the ET<sub>c</sub> field. More specifically, if the category of the crop is corn or potato, then the value is the ET<sub>day</sub> (grass's ET) multiplied by 1.1 (10% more), and if the category of the crop is carrot, then the value is the ET<sub>day</sub> multiplied by 1 (same as grass). The next step was to add a field called ET<sub>c</sub>\_week for the weekly evapotranspiration of the crops and populate the field using the ET<sub>c</sub>\_day multiplied by 7.

At this stage of the model, it was known the amount of weekly and daily evapotranspiration for the specific day for each type of crop.

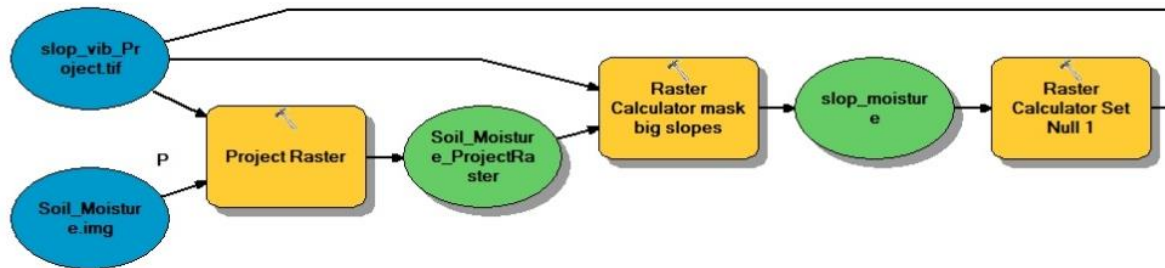


Figure 39: 3<sup>rd</sup> part of the model.

While the above-mentioned process was executed another process was taking place, which refers to the soil moisture information (Figure 39). The soil moisture image first, was projected in WGS 84 / UTM zone 32N projection and then the Viborg slope mask raster file (see 4.3.3. Data Collection and Process for rainfall) was used to mask out all the areas of Viborg, which they have slopes bigger than 5%. Lastly, a Set Null function in raster calculator was used and the result was a raster file with soil moisture information in Viborg municipality having No data values in areas with higher than 5% slope.

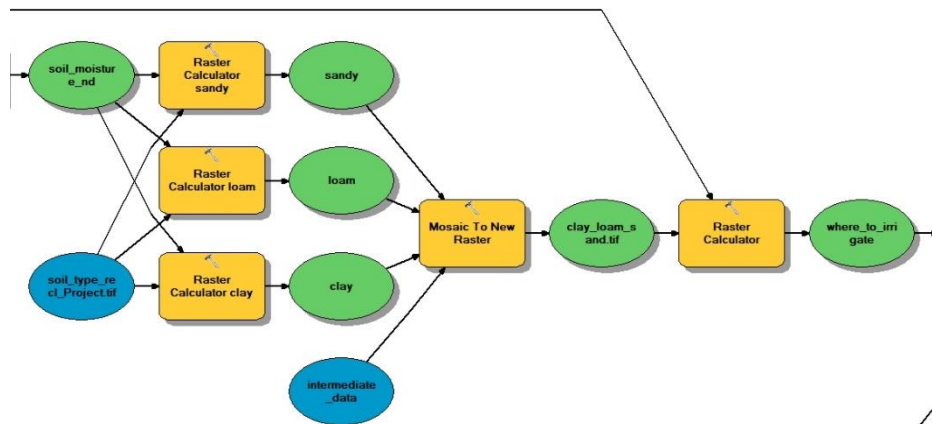


Figure 40: 4<sup>th</sup> part of the model.

After the previous process the masked raster file with the soil moisture information for Viborg was used with the raster file of soil type information (see 4.3.1. Data Collection and Process of Soil Moisture) (Figure 40). The two files were used inside 3 different Set Null functions (Figure 41).



Sand and Dry  
SetNull("soil\_type\_recl\_Resample" != 1, "moist\_slope\_v2" <= 0.05)

Loam and Dry  
SetNull("soil\_type\_recl\_Resample" != 2, "moist\_slope\_v2" <= 0.14)

Clay and Dry  
SetNull("soil\_type\_recl\_Resample" != 3, "moist\_slope\_v2" <= 0.17)

Figure 41: Set Null functions for soil type and moisture.

In the first function, only the areas with sandy soils and soil moisture values with lower values of 0.05 were kept and all the others were characterized as No Data. The second function keeps the values only in areas with loam type of soil and soil moisture less than 0.14, and the third function assign No Data values in all areas except the ones that they have clay type of soil and soil moisture values less than 0.17. The three raster files turned into one and at this point of the model a raster file was created having the values 1 in areas, where there is need of irrigations because the soil moisture values are lower than Wilting Point (see 4.1. Stage 1 – Literature Review) and 0 to all other areas. The last step was to mask out again the areas with high slopes.

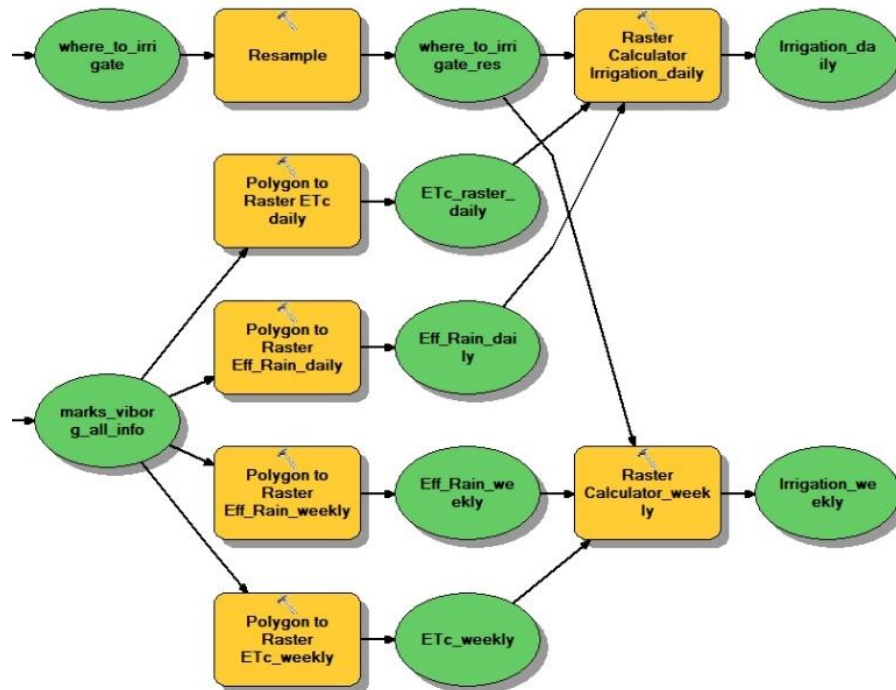


Figure 42: 5<sup>th</sup> part of the model.

The final part of the model (Figure 42) uses one shapefile with the information of marks in Viborg from the 1<sup>st</sup> and 2<sup>nd</sup> part and the raster file created from the 3<sup>rd</sup> and 4<sup>th</sup> part. The raster file consisting the information about the places which they should be irrigated was resampled. The new resolution was 10 x 10 m. The same resolution was used to create 4 different raster files from the shapefile, which consist information about the ETC and effective rainfall. The first raster had information about the daily ETC, the

second about the estimation of the daily effective rainfall, the third about the estimation of the weekly effective rainfall and the last, about the weekly ETc. The final calculation was made using the weekly and daily measurements. The reason why both measurements were used, was that irrigation can happen not necessary every day. So, the final results of the amount of water can be irrigated in one or more events during the week. In the next picture, it can be seen the final calculations (Figure 43) for daily and weekly measurements. The amount of effective rainfall was removed from the ETc amount and the result was multiplied with the raster file which has value 1 to the places which they should be irrigated and 0 to others. The results contain information about the amount of water that should be irrigated for every day and every week for a specific day with a resolution of 100 m<sup>3</sup>.

```
(("%ETc_raster_daily%" - "%Eff_Rain_daily%") * "%where_to_irrigate_res%")
(("%ETc_weekly%" - "%Eff_Rain_weekly%") * "%where_to_irrigate_res%")
```

Figure 43: Calculations of the final raster files.

### 3.5. Stage 5 – Process Automation

In this chapter, will be described how all the processes are automated. All the files of soil moisture images are stored locally, and the purpose of the automation was to run all the processes for each file. To do that three different scripts had to be created. The first was a .xml file, which will deliver the subset of each soil moisture satellite image, the second was a Python script, which will run all the process of the model as mentioned above, and the third, was a Java script, which will run for each file the two previous mentioned scripts defining two parameters.

The .xml file was already presented in the Chapter (3.3.1. Data Collection and Process of Soil Moisture). With this file, a subset of every image in the folder where the soil moisture pictures are stored (Figure 44) will be created in GeoTIFF format.



Name	Date modified	Type	Size
SM_OPER_MIR_SMUDP2_20180731T170600_20180731T175913_650_001_1.HDR	01-Aug-18 5:00 AM	HDR File	25 KB
SM_OPER_MIR_SMUDP2_20180731T170600_20180731T175913_650_001_1.DBL	01-Aug-18 5:00 AM	DBL File	18,294 KB
SM_OPER_MIR_SMUDP2_20180730T174457_20180730T183810_650_001_1.HDR	31-Jul-18 5:16 AM	HDR File	25 KB
SM_OPER_MIR_SMUDP2_20180730T174457_20180730T183810_650_001_1.DBL	31-Jul-18 5:16 AM	DBL File	17,669 KB
SM_OPER_MIR_SMUDP2_20180730T165449_20180730T174807_650_001_1.HDR	31-Jul-18 4:52 AM	HDR File	24 KB
SM_OPER_MIR_SMUDP2_20180730T165449_20180730T174807_650_001_1.DBL	31-Jul-18 4:52 AM	DBL File	9,835 KB

Figure 44: Folder containing SMOS data.

The Python file was created almost from the crutch as model builder in Arc Map, even though has an option to extract the model to a python script, lots of changes had to be done to run the model in python. There is no need to comment each line of the script because the process described in the python script is the same as was described in the previous chapter. The whole code is presented in the appendix and in this chapter will be commented only some parts that differ from the model.

More specific, after the libraries were imported inside the python script the next step was to define all the arguments. There were two arguments that they will be imported from Java in the next step. The first was the date and the second was the path of the soil moisture image. The date was taken from the name of the file of soil moisture. In the above figure, it can be seen how the names of SMOS data files are. At the first part of the code (Figure 45), there were some changes that had to be done referring to

the date format, which was delivered from the Java script. After this process, the date could be read from python.

```
# Script arguments
• Choose_date = sys.argv[1] #"2018-07-02\" # provide a default value if unspecified
• choose_date_2 = "\" "+Choose_date+"\"
• date_with_ = str(Choose_date)
• date_without_ = date_with_.replace("-", "_")
• date_without_and_qu = date_without_.strip('"')
• i = "D"+date_without_and_qu+"V"+(time.strftime("%Y%m%d_%H%M%S"))

• env.workspace = "C:/Data/Trial_model/trial.gdb"
• final_results = "C:/Data/final_results/"

# Marks
• marks_viborg = "marks_viborg"
• marks_viborg_3t = "marks_viborg_3t"+str(i)
• marks_viborg_new = "marks_viborg_new"+str(i)
• marks_viborg_new = marks_viborg_3t
• marks_viborg_new_2_ = "marks_viborg_new_2_"+str(i)
• marks_viborg_new_2_ = marks_viborg_new
• v3type_crop_viborg = marks_viborg_new_2_
• raster_marks = "raster_marks"+str(i)

# Slope
• slop_vib_Project = "slop_vib_Project"

# Soil Moisture
• Soil_Moisture_img=sys.argv[2]#"Soil_Moisture"
• Soil_Moisture_ProjectRaster = "Soil_Moisture_ProjectRaster"+str(i)
• slop_moisture = "slop_moisture"+str(i)
• soil_moisture_nd = "soil_moisture_nd"+str(i)
```

Figure 45: Python code with arguments.

The next part of the code refers to how to retrieve the information about the resolution and projection. Every raster file had to be at the same resolution and have the same projection. Similar, all the shapefiles must have the same projection. The processes to have the same projection and resolution was done using a reference raster file (Figure 46).

```
# SpatialReference
• out_coor_system = arcpy.Describe(slop_vib_Project).spatialReference

# Resolution

# Read cell size in X direction
• cellSizeX = arcpy.Describe(slop_vib_Project).meanCellWidth
# Read cell size in Y direction
• cellSizeY = arcpy.Describe(slop_vib_Project).meanCellHeight
# Resample
• cellSizeXY = u"{} {}".format(cellSizeX, cellSizeY)
```

Figure 46: Python code for the transformation of other files.

The other processes were made as described in the previous chapter using python.

The final step was to create a Java script, which is presented in the appendix. In this chapter, will be explained some parts of the script. After importing the libraries, the next step was to define the folder, where the data for soil moisture is stored (Figure 47). This process is made in line 24. Then with an if function, only files ending with DBL were processed to the next step.

```

23
24     File directory = new File("C:\\Data\\SMOS_data");
25
26     if (directory.isDirectory()) {
27
28         File[] files = directory.listFiles(new FileFilter() {
29             @Override
30             public boolean accept(File pathname) {
31
32                 if (pathname.getAbsolutePath().endsWith(".DBL")) {
33                     return true;
34                 } else {
35                     return false;
36                 }
37             }
38         });
39     }

```

Figure 47: Java code presenting if function.

In the next step, a for loop takes all the files from the previous step one by one (Figure 48). First, the path of the script.properties is defined. In this file, the full path of the input image of the soil moisture was defined and after that, the full path of the GeoTIFF image of the subset was also defined (Figure 49). The String date in line 55 gets the name of the file and the date is extracted as an argument. The date will be used in the python script. In lines 59 and 60 Java execute the .xml file using the gpt.exe file from SNAP, using the full path of DBL image. This information is written inside the script.properties file. The next step was to execute the python script in lines 62 and 63 using python with arguments the date and the GeoTIFF file created from the previous step.

```

40 //For each file
41 for (File f : files) {
42     PrintWriter out = null;
43     try {
44         System.out.println("Processing: " + f.getAbsolutePath());
45         ExecuteCommand ec = new ExecuteCommand();
46         //Write the properties file
47         String propertiesFilePath = "C:\\Data\\final scripts\\script.properties";
48         String outputFolder = f.getParentFile().getAbsolutePath();
49         out = new PrintWriter(propertiesFilePath);
50         out.println("product_dbl=" + f.getAbsolutePath().replace("\\", "\\\""));
51         out.println("tif_output=" + f.getAbsolutePath().replace("DBL", "").replace("\\", "\\\""));
52         out.flush();
53         out.close();
54
55         String date = "" + f.getName().substring(19, 23) +
56             "-" + f.getName().substring(23, 25) + "-" + f.getName().substring(25, 27);
57         System.out.println("Date: " + "\"" + date + "\"");
58         //Execute graph
59         ec.executeCommand("C:\\Program Files\\snap\\bin\\gpt.exe",
60             "C:\\Data\\final scripts\\Subset_from_whole_orbit_smos.xml", "-p", propertiesFilePath);
61
62         ec.executeCommand("C:\\Python27\\ArcGIS10.6\\pythonw.exe",
63             "C:\\Data\\final scripts\\model_thesis.py", date, f.getAbsolutePath().replace("DBL", "TIF"));

```

Figure 48: Java code presenting the for loop.

```

1 product_dbl=C:\\Data\\SMOS_data\\SM_OPER_MIR_SMUDP2_20180730T174457_20180730T183810_650_001_1.DBL
2 tif_output=C:\\Data\\SMOS_data\\SM_OPER_MIR_SMUDP2_20180730T174457_20180730T183810_650_001_1.
3

```

Figure 49: The script.properties file.

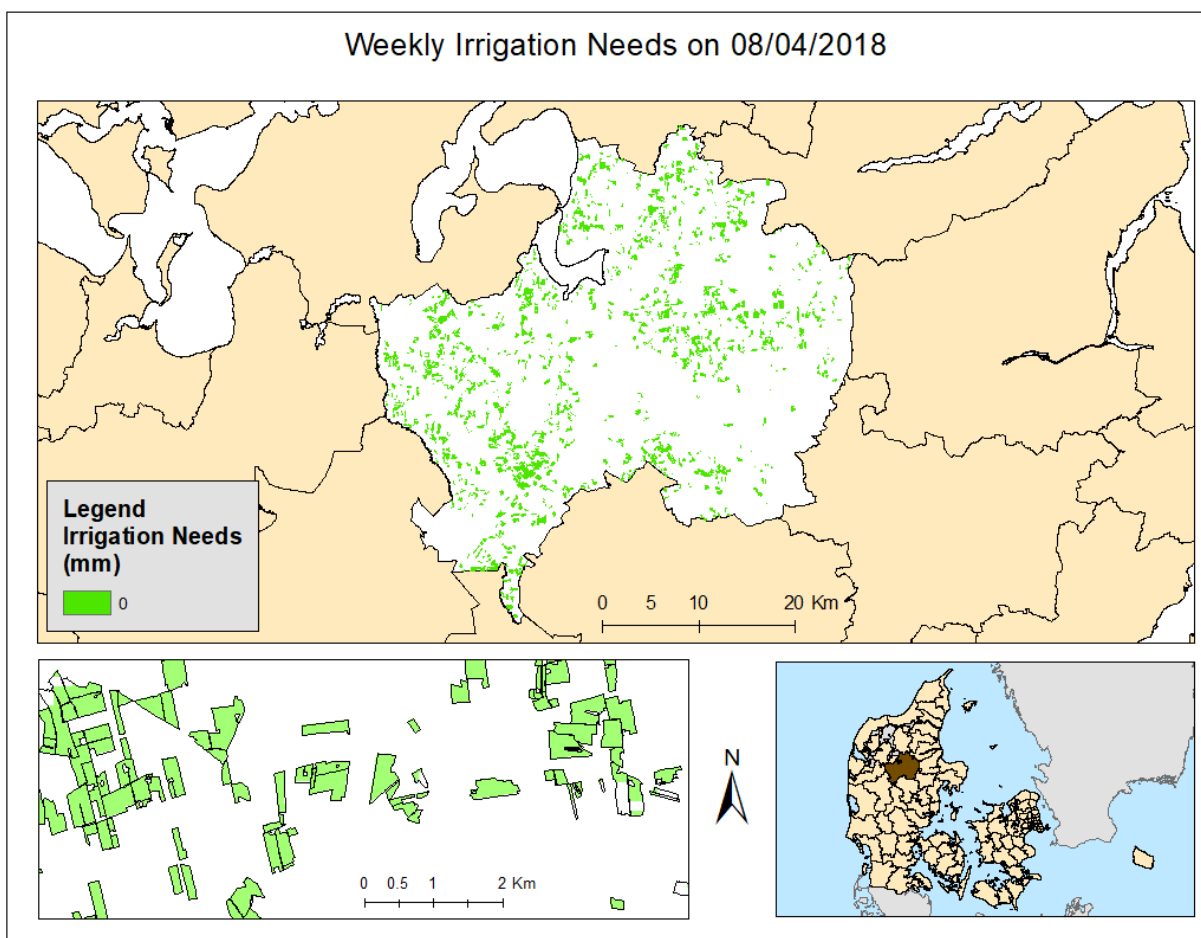
At this point, files with the irrigation needs for every day from March to July of 2018 using the soil moisture images were created and stored.

## 4. Results

This Chapter presents the results of the project. There are more than 60 Irrigation Needs maps from the period from March to July. There is no need to present all the maps in this Chapter. Therefore, six were chosen to be presented and commented with details in order to understand what the model can do.

The units used in the maps are referring to the amount of the water that farmers should irrigate in a specific area on their field depending on the type of the crop, the amount of water stored in the soil and the precipitation of the last days. The units are in mm. In this point is important to clarify what the unit refers to. 1 mm expresses the amount of water need of crops. As one hectare has a surface of 10,000 m<sup>2</sup> and 1 mm is equal to 0.001 m, water need of 1 mm of water, corresponds to 10 m<sup>3</sup> of water per hectare. Accordingly, 1mm of water needs means that farmers should irrigate 1 m<sup>3</sup> (1,000 Liter) per 1,000 m<sup>2</sup> or 0.1 m<sup>3</sup> (100 Liter) per 100 m<sup>2</sup>, etc.

Every map has the same format. The first big map presents the amount of water that farmers need to irrigate in the potatoes, carrots and corn fields in Viborg. The smaller map on the left presents the results from a smaller area and on the right, the map presents the location of Viborg municipality in Denmark, which is the test area of this project.

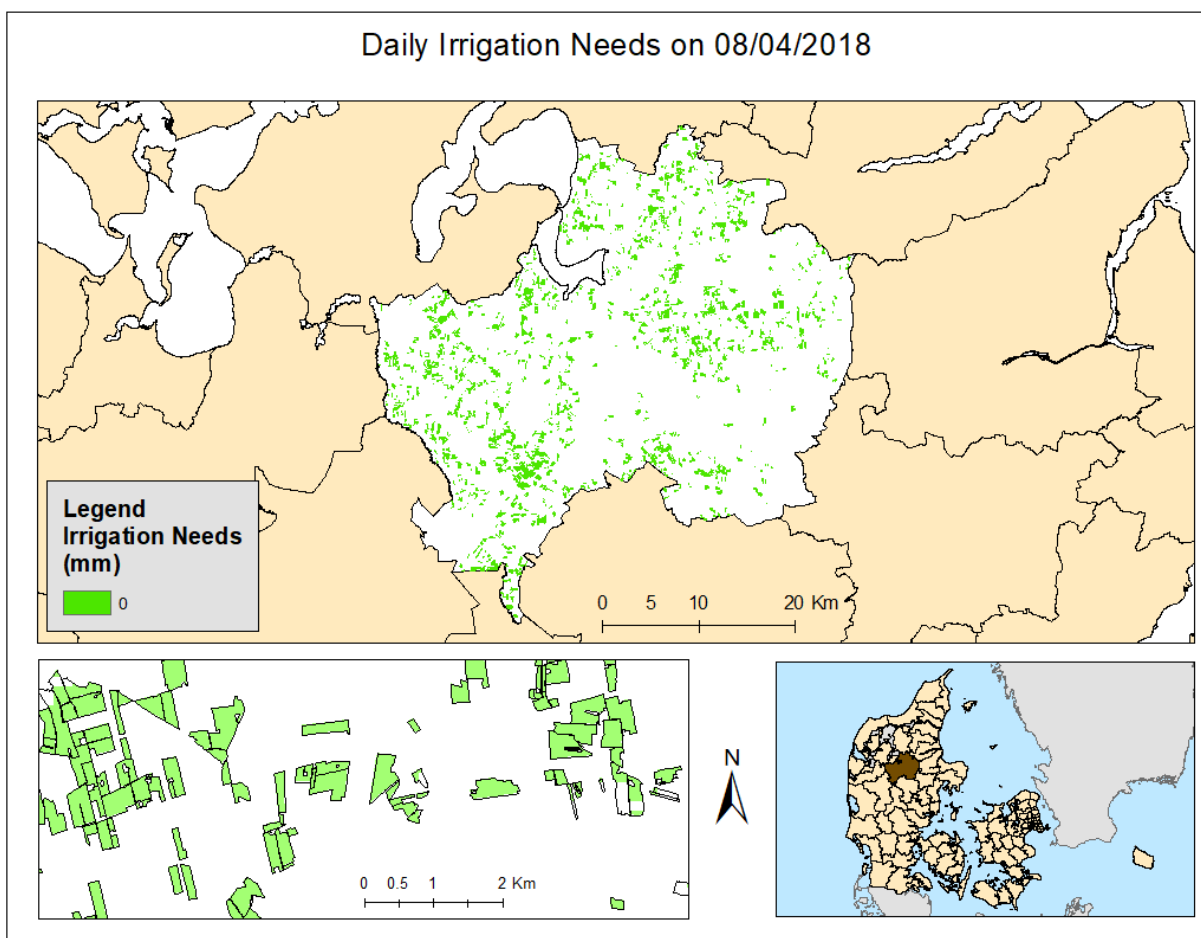


*Figure 50: Weekly Irrigation needs on 08/04/2018.*

The first map (Figure 50) presents the amount of water that farmers should irrigate in their fields on the 8<sup>th</sup> of April of 2018. The amount was calculated by the weekly needs. As it can be seen from the map, according to soil moisture, precipitation and evapotranspiration of the crops for the specific period there is no need for irrigation. Looking at the smaller map on the left, it can be seen that some fields or parts of fields are presented with white color. White color inside a field means that there was no soil moisture data of the area or the field is located in an area with more than 5% slope, with consequence the area is excluded from the calculations because the amount of effective rainfall could not be estimated. In the following table (Table 10) it can be seen the amount of water needs for different units and area size for this period.

*Table 10: Amount of water that farmers should irrigate on 08/04/2018.*

mm	m <sup>3</sup> of water per hectare	m <sup>3</sup> per 1,000 m <sup>2</sup>	Liter per 100 m <sup>2</sup>
0	0	0	0



*Figure 51: Daily Irrigation needs on 08/04/2018.*

The second map (Figure 51) presents the amount of water that farmers should irrigate in their fields on the 8<sup>th</sup> of April of 2018. As it was expected, observing the weekly needs of the specific day, according to soil moisture, precipitation and evapotranspiration of the crops for the specific day there is no need of irrigation. The following table (Table 11) presents the amount of water that farmers should irrigate in different units and scale for this day.

*Table 11: Amount of water that farmers should irrigate on 08/04/2018.*

mm	m <sup>3</sup> of water per hectare	m <sup>3</sup> per 1,000 m <sup>2</sup>	Liter per 100 m <sup>2</sup>
0	0	0	0



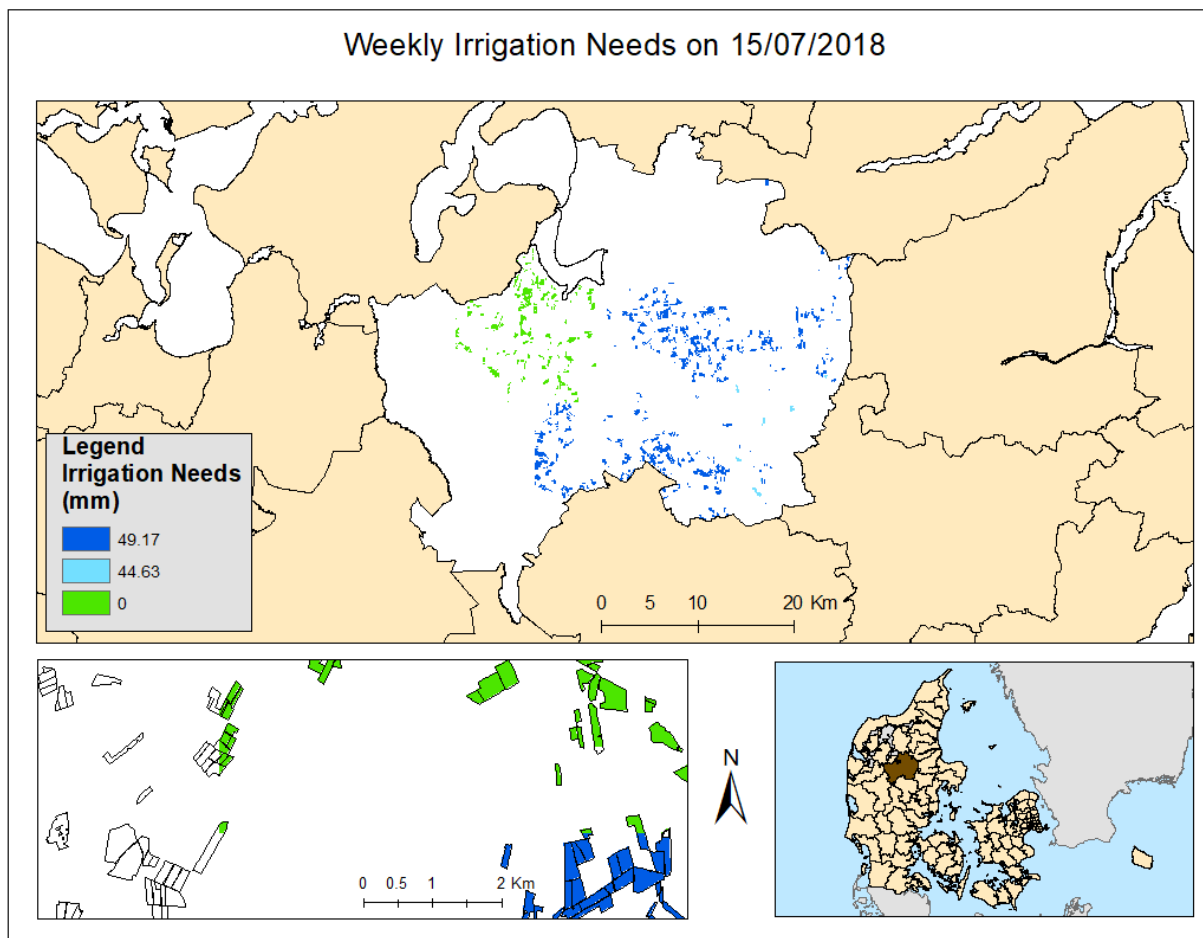
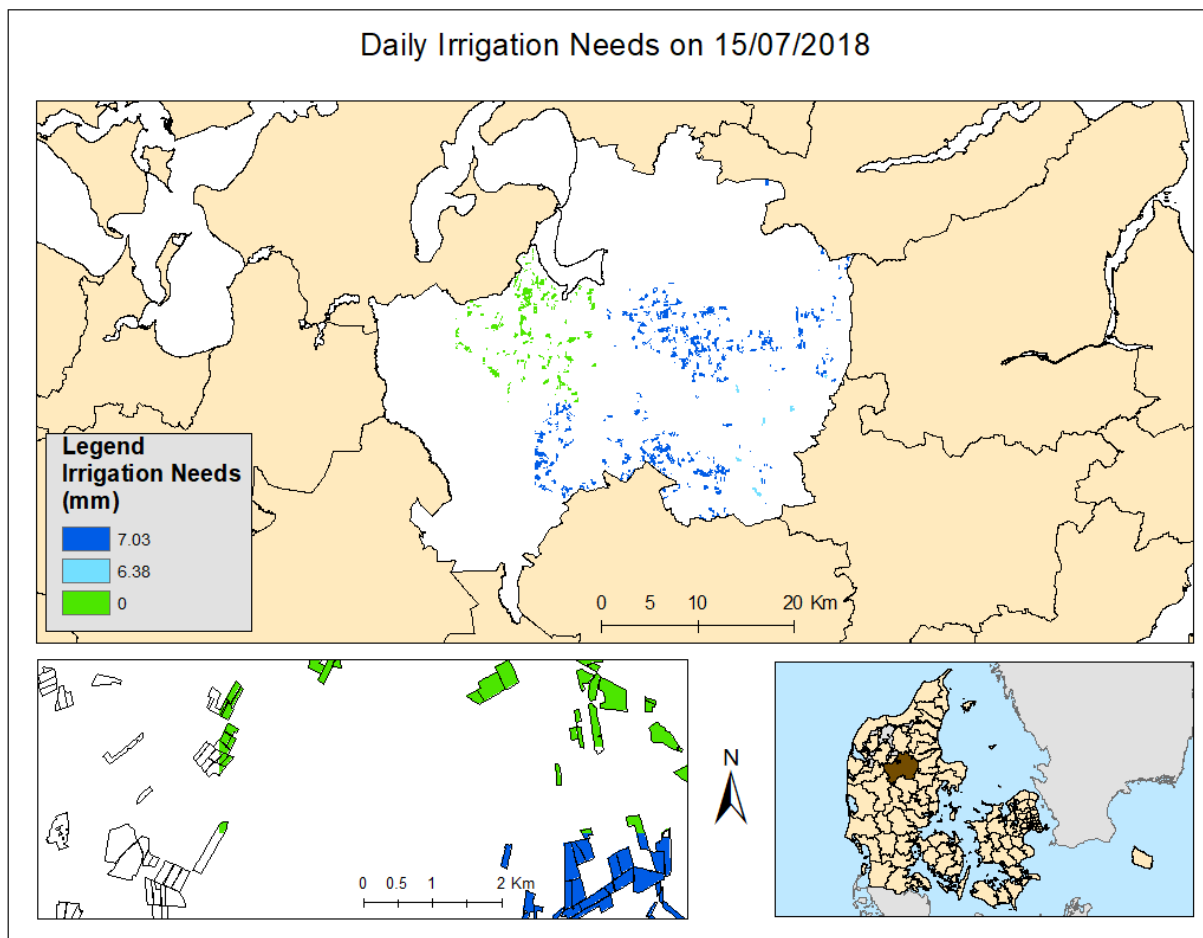


Figure 52: Weekly Irrigation needs on 15/07/2018.

The weekly needs for irrigation on the 15<sup>th</sup> of July of 2018 are presented in the map above (Figure 52). Fields and part of fields with white colors mark areas with no data values. There is no weekly need of irrigation on the most of the fields on the northwest part of Viborg, while in on the southeast part, crops were missing water and irrigation had to be applied. Areas with crops needing more water are presented with dark blue color and needs for less water are presented with a light blue color. 491.70 m<sup>3</sup> of water per hectare should be irrigated in most of the areas during the week, while in light blue areas the amount is 446.30 m<sup>3</sup> of water per hectare. In the following table (Table 12) it can be seen the amount of water needs for different units and area size for this period.

Table 12: Amount of water that farmers should irrigate on 15/07/2018.

mm	m <sup>3</sup> of water per hectare	m <sup>3</sup> per 1,000 m <sup>2</sup>	Liter per 100 m <sup>2</sup>
0	0	0	0
44.63	446.30	44.63	4,463
49.17	491.70	49.17	4,917



*Figure 53: Daily Irrigation needs on 15/07/2018.*

The map above (Figure 53) presents the results of the amount of water that farmers should irrigate on the 15<sup>th</sup> of July of 2018. There was no data for soil moisture for the west and north part of Viborg municipality and this is the reason why there are no results for the mentioned areas. Looking closer to a randomly taken area (small map on the left), it can be seen that there is no need for irrigation in some fields as the soil moisture and type of soil can provide water to the plants for the specific period. While in some other areas there is a need for irrigation as the soil moisture could not provide water to the plants and effective rainfall was not enough to cover the water needs of the crops. In areas presented with dark blue colors the amount of water that the farmers should irrigate on the 15<sup>th</sup> of July is 70.30 m<sup>3</sup> of water per hectare, and 63.80 m<sup>3</sup> of water per hectare for the light blue areas. The following table (Table 13) presents the amount of water that farmers should irrigate in different units and scale for this day.

*Table 13: Amount of water that farmers should irrigate on 15/07/2018.*

mm	m <sup>3</sup> of water per hectare	m <sup>3</sup> per 1,000 m <sup>2</sup>	Liter per 100 m <sup>2</sup>
0	0	0	0
6.38	63.80	6.38	638
7.03	70.30	7.03	703

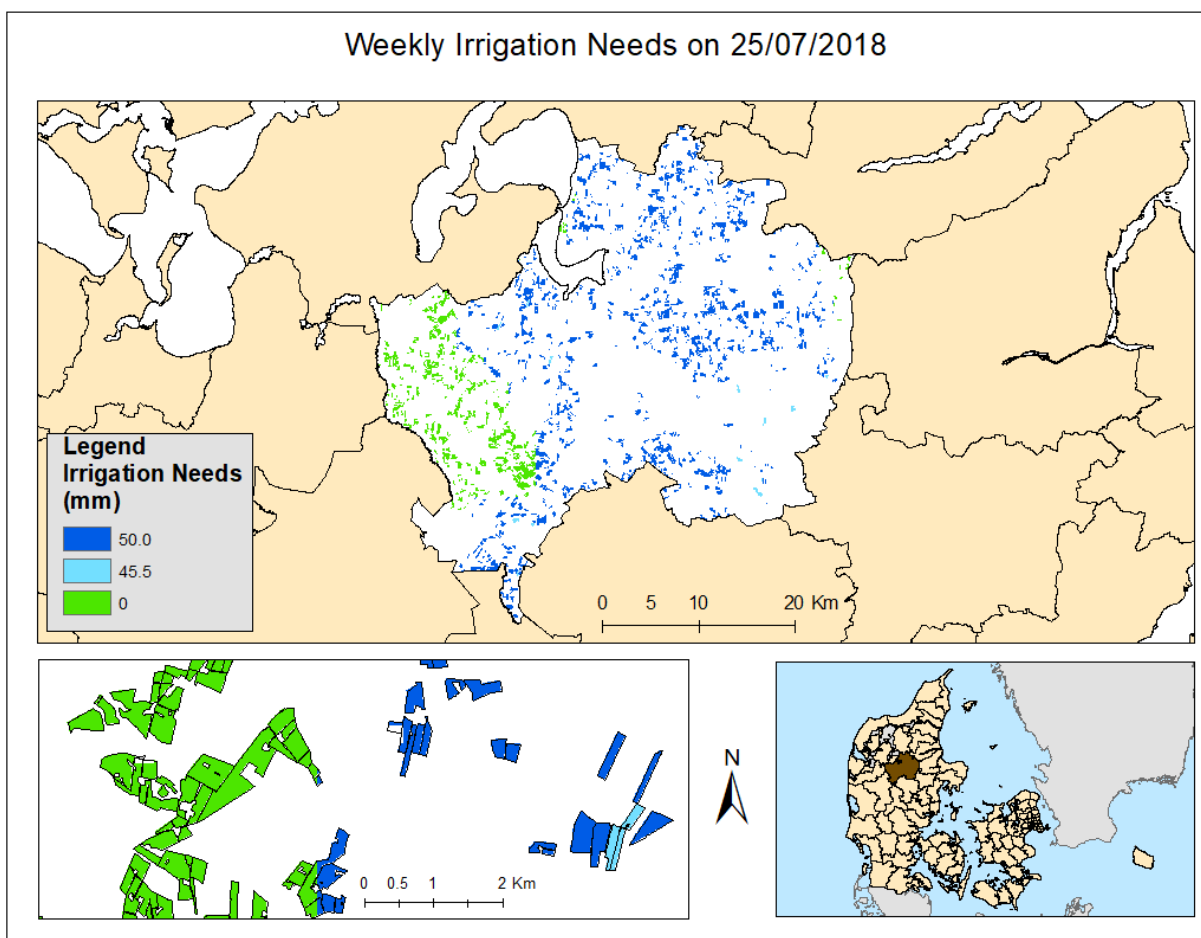


Figure 54: Weekly Irrigation needs on 25/07/2018.

The weekly needs for irrigation on the 25<sup>th</sup> of July of 2018 are presented in the map above (Figure 54). There is no weekly need for irrigation on the fields that are located on the west Viborg, but there was a need for irrigation on the rest areas of the municipality. Areas with crops needing more water are presented with dark blue color and needs for less water are presented with a light blue color.  $\text{m}^3$  of water per hectare should be irrigated in most of the areas during the week, while in light blue areas the amount is  $455 \text{ m}^3$  of water per hectare. In the following table (Table 14) it can be seen the amount of water needs for different units and area size for this period.

Table 14: Amount of water that farmers should irrigate on 25/07/2018.

mm	$\text{m}^3$ of water per hectare	$\text{m}^3$ per $1,000 \text{ m}^2$	Liter per $100 \text{ m}^2$
0	0	0	0
45.50	455.00	45.50	4,550
50.00	500.00	50.00	5,000

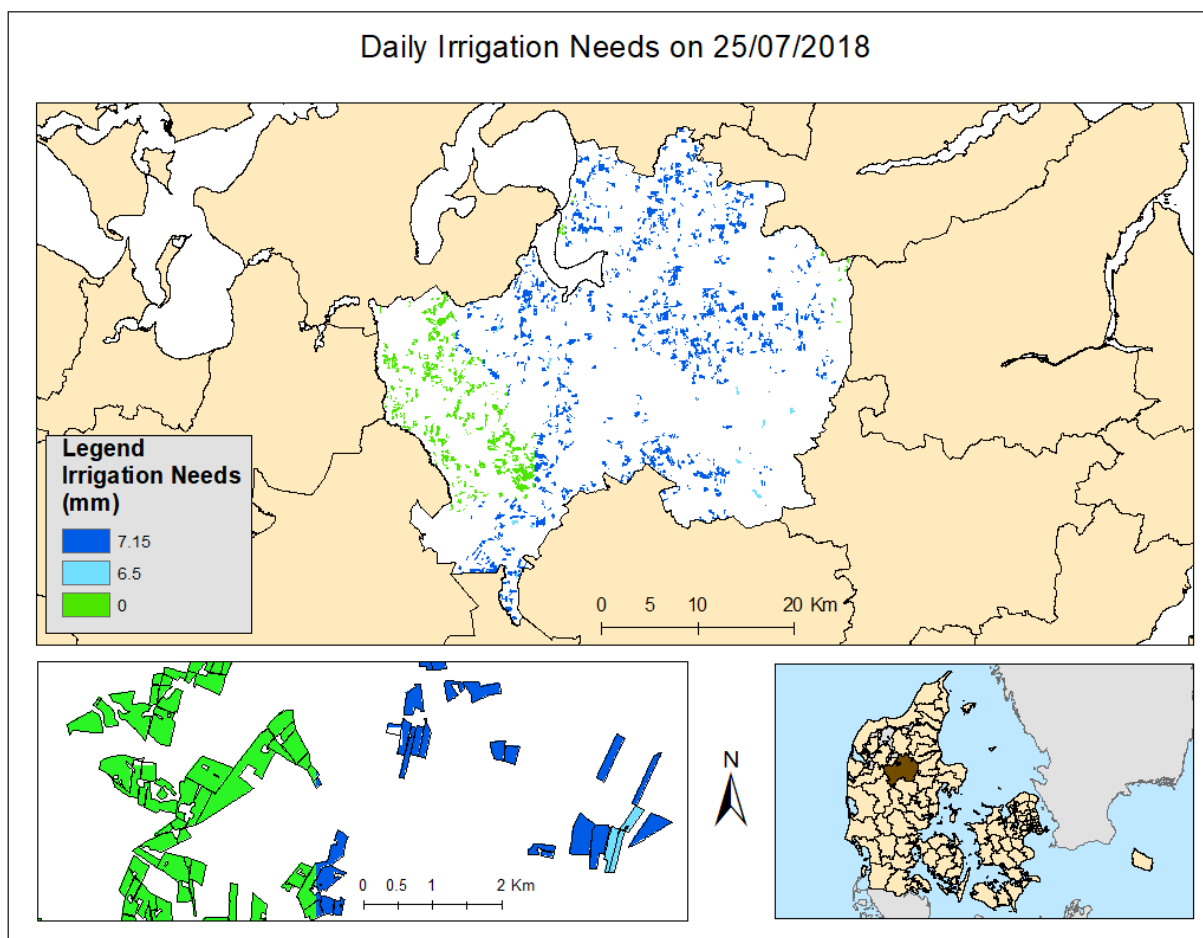


Figure 55: Daily Irrigation needs on 25/07/2018.

The map above (Figure 55) presents the results of the amount of water that farmers should irrigate on the 25<sup>th</sup> of July of 2018. Looking closer to a specific area (small map on the left), it can be seen that there is no need for irrigation in some fields as the soil moisture and type of soil can provide water to the plants for the specific day. Some other areas need irrigation as the soil moisture could not provide water to the crops and effective rainfall was not enough to cover the water needs of the plants. In areas presented with dark blue colors the amount of water that the farmers should irrigate on the 25<sup>th</sup> of July was 71.50 m<sup>3</sup> of water per hectare, and 65 m<sup>3</sup> of water per hectare for the light blue areas. The part of the field which is presented with white color is located in areas with higher than 5% slope. The following table (Table 15) presents the amount of water that farmers should irrigate in different units and scale for this day.

Table 15: Amount of water that farmers should irrigate on 25/07/2018.

mm	m <sup>3</sup> of water per hectare	m <sup>3</sup> per 1,000 m <sup>2</sup>	Liter per 100 m <sup>2</sup>
0	0	0	0
6.50	65.00	6.50	650
7.15	71.50	7.15	715

## 5. Discussion

Nowadays using remote sensing products is not an advantage for the few. Having satellite images in the past could be expensive and difficult but all that changed. Anyone can have access to remote sensing products and in lots of cases, these products are free.

The European Space Agency (ESA) which is Europe's gateway to space provides free data to the public. It is an international organization with 22 Member States (from the EU) while there are also countries that have cooperation agreements with ESA, associate members and countries like Canada who take part in some projects under a cooperation agreement. ESA's mission is to find out more about Earth, its immediate space environment, our Solar System and the Universe and to develop satellite-based technologies and services promoting European industries. The primary goal of ESA is to ensure that the investments in space deliver benefits to the citizens of Europe and the world (ESA (European Space Agency) 2018b). All people of the world can benefit from the freely available data that ESA provides and with this data scientists and decision-makers can have the information they need to understand and protect our environment. The continuous views of our planet which are supplied by satellites have many applications. Among others, they are monitoring the air, seas, and land and they can supply agencies with data when disasters strike (ESA (European Space Agency) 2018a).

Of course, there are still some products that people should pay or there are some that people must have the appropriate equipment to get them. Moreover, online courses and free software for the processing of satellite images can help people who are not specialist to create their own products. All the above-mentioned drive to the conclusion that the last years remote sensing scientists tried to transmit their knowledge to the public, encouraging individual people and companies to use more remote sensing products.

FAO has many examples that can calculate irrigation needs, without any calculations about the soil moisture. Other organizations or private companies are offering applications about irrigation needs using satellite images, but they do not calculate the phenology of the crops. To have a fully covered irrigation management decision support system lots of parameters should be calculated.

In the results, it is clear that remote sensing products and weather data can provide information about the crop water needs. Soil moisture information on the past could only be retrieved by measurements on the ground. Farmers on the past were digging holes to the ground so that they could understand if their field needed water or not. There are also instruments that can measure soil moisture but, in some cases, the instruments were and still are expensive. New technologies are used nowadays to create cheaper products for soil moisture measurements, but cheaper instruments cannot offer very accurate measurements.

In the results, the amount of water that farmers should irrigate in specific areas is changing. The amount is depending on the type of crop. For example, potatoes and corn need more water than carrots and due to this, the values of the irrigation needs were deferent in some areas. The amount of rainfall also affects the results, leading to fewer water needs than the amount of the total evapotranspiration of an area. Areas in the results with white color indicate areas without measurements of soil moisture or areas with high slopes. In some cases, the mentioned areas where inside one field, with the result the farmer could know how much water should be irrigated in only a part of his field. This is a problem as the farmers need

information for the whole area that they are interested in and they need this information for every day. There is also a pattern in the results. Areas in the west of Viborg often do not need irrigation. This is due to the soil type which can hold water inside easier than the other types of soil on the east part of Viborg. So, in some cases, the soil moisture had the same values in a broad area, but the area had to be irrigated due to the soil type.

As for the future development of the project the first that could be easily implemented is to use deferent soil moisture information. The model is very flexible to changes. Soil moisture information with better resolution and better temporal resolution can lead to more accurate results. The other part that can be changed is to find another formula for calculating the effective rainfall. There were some assumptions made to calculate the effective rainfall. The amount of the rainfall was estimated by calculating all the measurements of the past month. For more accurate results may be another formula should be used. Ideally, will be very interesting if the values of the prediction of the next day's rainfall can be implemented in the model. There is no need for irrigating the amount of water that will fall from the sky and can be used effectively form the plants. In an improved version of the model also the phenology of the plants can be taken into account while the crop development can be used also to calculate the irrigation needs. Plants need more water after seed and less before harvest, but this is not applied in all types of crops. Moreover, the evapotranspiration rate of the area can also be calculated directly from satellite images and in this case, it would be interesting to use also this information inside the support system.

The model can be sensitive if basic changes occur. The output of the model could be deferent if the input type of satellite image of soil moisture changes. The resolution of the image is very important and in the present study high spatial resolution images are not used. In case that high spatial resolution images are used as input in the model then there could be different values of irrigation water amount. Different values could also derive from the model in case that the rainfall was calculated using interpolation methods or using different methods to calculate effective rainfall.

## 6. Conclusion

This project is an example of how geoinformatics and remote sensing products can offer the possibility of better management of water resources and reduced environmental impact of agricultural production. The output can benefit Danish farmers, and the export potential to other regions of the World is large. The decision support system created can provide information about water needs. The users can have information about crop water requirements to optimize irrigation.

Calculating the amount of irrigation of the crops is not an easy process. There are lots of parameters that someone can take into account. In this study, the model created checks the amount of the water exists in the soil and then decides if the area should have water from irrigation or not. After the first decision made automated, from the model, which takes into account the soil types of the areas with combination the soil moisture, the second decision is made. The model calculates the amount of the water that should be irrigated in each area on the field, calculating the ET and effective rainfall of the last days. Using weather data, remotely sensed data the model was created, and it is flexible to use and modify. This project was created using mostly open source software like QGIS and SNAP while Sentinel data and weather from DMI is the original inputs for the model.

Accurate and freely available soil moisture satellite data is difficult to obtain, while cloud coverage makes difficult to have data whenever is needed. Nowadays, remote sensing can give a solution by estimating the soil moisture from satellites. The last years an effort is made to produce more accurate and useful information about the soil moisture. There are methods that can use radar data from the satellite in order to estimate soil moisture. This means that the images are not limited to clear sky observations, which is very important as in lots of areas of the world clear sky days are not so many during a year. The Mediterranean region, for example, is a region with good coverage from satellites while clear sky observations can happen often because of the climate. On the other hand, Denmark is located in an area in which there are more observations, but the skies are often covered by clouds. Areas such as Senegal or India have a season which the skies are covered all the time by clouds and another season with clear skies. In some cases, the ground data or radar data from the satellites can be the solution in order to have measurements during all year.

In Denmark, the output for the irrigation management can be used for increasing yields and preserve the amount of nitrogen in the soil. For other regions of the world is very important also to save water. Areas in the Mediterranean which have drought periods and in some cases no water to irrigate, need an irrigation management system. Other areas like in Africa have limited water availability for long periods. Systems like that can help to better management of the available water.

## 7. References

- Allen, Richard G. 2006. "Evaporation Modeling: Potential."  
<https://doi.org/10.1002/0470848944.hsa044>.
- BME ABÉT. 2008. "Lysimeters from Toka Settlement in September 2008." 2008. <http://enfo.agt.bme.hu/drupal/en/node/1929>.
- Brouwer, C., A. Goffeau, and M. Heibloem. 1985. *Irrigation Water Management: Training Manual No. 1 - Introduction to Irrigation*. Provisiona. Rome: FAO - Food and Agriculture Organization of the United Nations.  
<http://www.fao.org/3/r4082e/r4082e00.htm#Contents>.
- Brouwer, C., and M Heibloem. 1986. "Irrigation Water Management: Irrigation Water Needs, Training Manual No. 3 ( Chapter 3 & Chapter 4 ), Food and Agriculture Organisation (FAO)." In *Irrigation Water Management: Irrigation Water Needs, Training Manual No. 3*. Rome, Italy. <http://www.fao.org/3/S2022E/s2022e00.htm#Contents>.
- DMI (Danish Meteorology Institute). 2019. "Vejrarkiv." 2019.  
<https://www.dmi.dk/vejrarkiv/>.
- ESA. 2019. "SMOS Mission Summary - ESA Earth Observation Missions - Earth Online - ESA." 2019.  
<https://earth.esa.int/web/guest/missions/esa-eo-missions/smos/mission-summary>.
- ESA (European Space Agency). 2018a. "Missions - Earth Online - ESA." 2018. <https://earth.esa.int/web/guest/missions>.
- ESA (European Space Agency). 2018b. "What Is ESA? / Welcome to ESA / About Us / ESA." 2018.  
[http://www.esa.int/About\\_Us/Welcome\\_to\\_ESA/What\\_is\\_ESA](http://www.esa.int/About_Us/Welcome_to_ESA/What_is_ESA).
- FAO. 2015. *World Reference Base for Soil Resources 2014 International Soil Classification System*.
- FAO (Food and Agriculture Organization). 2014. "FAO at World Water



- Week 2014 Why Water and Energy Matter for Agriculture?" 2014.  
<http://www.fao.org/land-water/news-archive/news-detail/en/c/267274/>.
- Ghebreamlak, Araya Z, Haruya Tanakamaru, Akio Tada, Bashir M Ahmed Adam, and Khalid A E Elamin. 2018. "Satellite-Based Mapping of Cultivated Area in Gash Delta Spate Irrigation System , Sudan," 1–14.  
<https://doi.org/10.3390/rs10020186>.
- Herrero, J., and M. A. Casterad. 1999. "Using Satellite and Other Data to Estimate the Annual Water Demand of an Irrigation District."  
*Environmental Monitoring and Assessment* 55 (2): 305–317.
- Hillel, Daniel., Arthur W. Warrick, Ralph S. Baker, and Cynthia. Rosenzweig. 1998. *Environmental Soil Physics*. Academic Press.
- Liedekerke, Marc Van, Arwyn Jones, and Panos Panagos. 2006. "ESDBv2 Raster Library - a Set of Rasters Derived from the European Soil Database Distribution v2.0 (Published by the European Commission and the European Soil Bureau Network, CD-ROM, EUR 19945 EN)."
- Michael Falch (Danmarks Statistik). 2010. "Danmark i Tal."
- Miljø, Og, and Fødevareministeriet. 2019. "Jordbrugsanalyser." 2019.  
<http://miljoegis.mim.dk/cbkort?profile=jordbrugsanalyse>.
- MySQL. 2019. "MySQL Workbench." 2019.  
<https://www.mysql.com/products/workbench/>.
- Nieto, Héctor (IRTA), Radoslaw (DHI) Guzinski, Inge (SANDHOLT APS) Sandholt, Georgios (SANDHOLT APS) Karamitilios, Mads Rasmussen Olander, (DHI), Joaquim (IRTA) Bellvert, et al. 2019. "SENTINELS FOR EVAPOTRANSPIRATION [SEN-ET]."
- Panagos, Panos. 2006. "The European Soil Database" GEO: conne (5 (7)): 32–33.
- Penman, H.L. 1963. *Vegetation and Hydrology, Technical Communication No. 53*. Commonwealth Bureau of Soils: Harpenden, p. 125.

- Pruitt, Doorenbos; 1984. *Crop Water Requirements Crop Water Requirements*.
- Rich, Paul M, and Jing Chen. 1995. "Calculation of Leaf Area Index and Other Canopy Indices From Gap Fraction ;," no. c.
- Richard G. Allen, Luis S. Pereira, Dirk Raes, and Martin Smith. 1998. *Crop Evapotranspiration - Guidelines for Computing Crop Water Requirements - FAO Irrigation and Drainage Paper 56*. Rome, Italy: FAO - Food and Agriculture Organization of the United Nations. <http://www.fao.org/3/X0490E/x0490e00.htm#Contents>.
- Sandholt, Inge, L Rasmussen, and Jens Andersen. 2002. "A Simple Interpretation of TS\_NDVI Space for Assessment of Surface Moisture Status." *Remote Sensing of Environment* 79: 213–24. [https://doi.org/10.1016/S0034-4257\(01\)00274-7](https://doi.org/10.1016/S0034-4257(01)00274-7).
- Santos, C., I. J. Lorite, M. Tasumi, R. G. Allen, and E. Fereres. 2008. "Integrating Satellite-Based Evapotranspiration with Simulation Models for Irrigation Management at the Scheme Level." *Irrigation Science* 26 (3): 277–288.
- Spruceirrigation.com. 2019. "What Is My Target Moisture Level?" 2019. <http://support.spruceirrigation.com/knowledge-base/what-is-my-target-moisture-level/>.
- Thornthwaite, C.W. 1948. *An Approach toward a Rational Classification of Climate*. Geographical Review, 38, 55.

## 8. Appendix

### Java Code

```
1. package thesisexecuter;
2.
3. import java.io.File;
4. import java.io.FileFilter;
5. import java.io.FileNotFoundException;
6. import java.io.PrintWriter;
7. import java.util.logging.Level;
8. import java.util.logging.Logger;
9.
10. /**
11.  *
12.  * @author Giorgos
13.  */
14. public class ThesisExecuter {
15.
16.     /**
17.      * @param args the command line arguments
18.      */
19.     public static void main(String[] args) {
20.         // TODO code application logic here
21.
22.         System.out.println(" Start");
23.
24.         File directory = new File("C:\\Data\\SMOS_data");
25.
26.         if (directory.isDirectory()) {
27.
28.             File[] files = directory.listFiles(new FileFilter() {
29.                 @Override
30.                 public boolean accept(File pathname) {
31.
32.                     if (pathname.getAbsolutePath().endsWith("DBL")) {
33.                         return true;
34.                     } else {
35.                         return false;
36.                     }
37.                 }
38.             });
39.
40.             //For each file
41.             for (File f : files) {
42.                 PrintWriter out = null;
43.                 try {
44.                     System.out.println("Processing: " +
45. f.getAbsolutePath());
46.                     ExecuteCommand ec = new ExecuteCommand();
47.                     //Write the properties file
48.                     String propertiesFilePath = "C:\\Data\\final
49. scripts\\script.properties";
50.                     String outputFolder =
51. f.getParentFile().getAbsolutePath();
52.                     out = new PrintWriter(propertiesFilePath);
```

```

50.         out.println("product_dbl=" +
f.getAbsolutePath().replace("\\", "\\\\"));
51.         out.println("tif_output=" +
f.getAbsolutePath().replace("DBL", "").replace("\\", "\\\\"));
52.         out.flush();
53.         out.close();
54.
55.         String date = "" + f.getName().substring(19, 23) +
"-" + f.getName().substring(23, 25) + "-" + f.getName().substring(25,
27);
56.         System.out.println("Date: " + "\"" + date + "\"");
57.         //Execute graph
58.         //ec.executeCommand("C:\\Program
Files\\snap\\bin\\gpt.exe", "C:\\Data\\final
scripts\\Subset_from_whole_orbit_smos.xml", "-p", propertiesFilePath);
59.
60.         ec.executeCommand("C:\\Python27\\ArcGIS10.6\\pythonw.exe",
"C:\\Data\\final scripts\\model_thesis.py", date,
f.getAbsolutePath().replace("DBL", "TIF"));
61.         } catch (FileNotFoundException ex) {
62.             Logger.getLogger(ThesisExecuterbackup.class.getName()).log(Level.SEVERE
, null, ex);
63.         } finally {
64.             out.close();
65.         }
66.     }
67.
68. }
69.
70. }
71.
72. }
73.

```

## XML Code

```
1. <graph id="Graph">
2.   <version>1.0</version>
3.   <node id="Read">
4.     <operator>Read</operator>
5.     <sources/>
6.     <parameters class="com.bc.ceres.binding.dom.XppDomElement">
7.       <file>${product_dbl}</file>
8.     </parameters>
9.   </node>
10.  <node id="Subset">
11.    <operator>Subset</operator>
12.    <sources>
13.      <sourceProduct refid="Read"/>
14.    </sources>
15.    <parameters class="com.bc.ceres.binding.dom.XppDomElement">
16.      <sourceBands>Soil_Moisture</sourceBands>
17.      <region>0,0,3270,1201</region>
18.      <geoRegion>POLYGON ((7 54, 13 54, 13 58, 7 58, 7 54, 7
19.      54))</geoRegion>
20.      <subSamplingX>1</subSamplingX>
21.      <subSamplingY>1</subSamplingY>
22.      <fullSwath>>false</fullSwath>
23.      <tiePointGridNames/>
24.      <copyMetadata>>true</copyMetadata>
25.    </parameters>
26.  </node>
27.  <node id="Write">
28.    <operator>Write</operator>
29.    <sources>
30.      <sourceProduct refid="Subset"/>
31.    </sources>
32.    <parameters class="com.bc.ceres.binding.dom.XppDomElement">
33.      <file>${tif_output}</file>
34.      <formatName>GeoTIFF</formatName>
35.    </parameters>
36.  </node>
37.  <applicationData id="Presentation">
38.    <Description/>
39.    <node id="Read">
40.      <displayPosition x="37.0" y="134.0"/>
41.    </node>
42.    <node id="Subset">
43.      <displayPosition x="242.0" y="136.0"/>
44.    </node>
45.    <node id="Write">
46.      <displayPosition x="332.0" y="259.0"/>
47.    </node>
48.  </applicationData>
49. </graph>
```

## Python Code

```
1. #-----
   #-----
2. # Name:      model_thesis.py
3. # Purpose:
4. #
5. # Author:     Giorgos
6. #
7. # Created:    29-04-2019
8. # Copyright:  (c) Giorgos 2019
9. # Licence:    <your licence>
10. # Import arcpy module
11. import arcpy
12. from arcpy import env
13. from arcpy.sa import *
14. arcpy.CheckOutExtension('Spatial')
15.
16. # Script arguments
17.
18. Choose_date = sys.argv[1] #"\"2018-07-02\"" # provide a default value
   if unspecified
19. choose_date_2 = "\" "+Choose_date+"\""
20. date_with_ = str(Choose_date)
21. date_without_ = date_with_.replace("-", "_")
22. date_without_and_qu = date_without_.strip('"\'')
23. i = "D" + date_without_and_qu + "V" + (time.strftime("%Y%m%d_%H%M%S"))
24.
25. env.workspace = "C:/Data/Trial_model/trial.gdb"
26. final_results = "C:/Data/final_results/"
27.
28. # Marks
29. marks_viborg = "marks_viborg"
30. marks_viborg_3t = "marks_viborg_3t" + str(i)
31. marks_viborg_new = "marks_viborg_new" + str(i)
32. marks_viborg_new = marks_viborg_3t
33. marks_viborg_new_2_ = "marks_viborg_new_2_" + str(i)
34. marks_viborg_new_2_ = marks_viborg_new
35. v3type_crop_viborg = marks_viborg_new_2_
36. raster_marks = "raster_marks" + str(i)
37.
38. # Slope
39. slop_vib_Project = "slop_vib_Project"
40.
41. # Soil Moisture
42. Soil_Moisture_img = sys.argv[2] #"Soil_Moisture"
43. Soil_Moisture_ProjectRaster = "Soil_Moisture_ProjectRaster" + str(i)
44. slop_moisture = "slop_moisture" + str(i)
45. soil_moisture_nd = "soil_moisture_nd" + str(i)
46. soil_type_recl_Project = "soil_type_recl_Project"
47. clay = "moist_type_clay" + str(i)
48. sandy = "moist_type_sandy" + str(i)
49. loam = "moist_type_loam" + str(i)
50. where_to_irrigate = "where_to_irrigate" + str(i)
51. soil_moisture_nd_value1 = "soil_moisture_nd_value1" + str(i)
52. soil_moisture_nd_value1_p = "soil_moisture_nd_value1_p" + str(i)
```

```

53.
54. # Weather data
55. viborg_weather_data = "viborg_weather_data"
56.
57. # ETc and Effective rainfall
58. marks_viborg_ETc = v3type_crop_viborg
59. marks_viborg_ETc_values = marks_viborg_ETc
60. marks_viborg_ETc_week = marks_viborg_ETc_values
61. marks_viborg_all_info = marks_viborg_ETc_week
62. ETc_weekly = "ETc_weekly"+str(i)
63. Eff_Rain_weekly = "Eff_Rain_weekly"+str(i)
64. ETc_raster_daily = "ETc_raster_daily"+str(i)
65. Eff_Rain_daily = "Eff_Rain_daily"+str(i)
66.
67. # Irrigation
68. Irrigation_weekly_n = "Irrig_weekly_n"+str(i)
69. Irrigation_daily_n = "Irrig_daily_n"+str(i)
70. Irrigation_weekly = "I_weekly"+str(i)
71. Irrigation_daily = "I_daily"+str(i)
72. Irr_weekly= final_results+i+"_w.tif"
73. Irr_daily= final_results+i+"_d.tif"
74.
75. # SpatialReference
76. out_coord_system = arcpy.Describe(slop_vib_Project).spatialReference
77.
78. # Resolution
79.
80. # Read cell size in X direction
81. cellSizeX = arcpy.Describe(slop_vib_Project).meanCellWidth
82. # Read cell size in Y direction
83. cellSizeY = arcpy.Describe(slop_vib_Project).meanCellHeight
84. # Resample
85. cellSizeXY = u"{} {}".format(cellSizeX, cellSizeY)
86.
87. ##### START
88. #####
89. # Make a layer from the feature class
90. arcpy.MakeFeatureLayer_management(marks_viborg, "lyr")
91.
92. # Select 3 types of crops
93. arcpy.SelectLayerByAttribute_management("lyr", "NEW_SELECTION",
    "\"AfgKat\" = 'Gulerod' OR \"AfgKat\" = 'Kartoffel' OR \"AfgKat\" =
    'Majs'")
94.
95. # Process: Copy Features with 3 types of crops
96. arcpy.CopyFeatures_management("lyr", marks_viborg_3t, "", "0", "0",
    "0")
97.
98. # Process: Add Field Date
99. arcpy.AddField_management(marks_viborg_new, "date", "DATE", "", "",
    "", "", "NULLABLE", "NON_REQUIRED", "")
100.
101. # Process: Calculate Field Date
102. arcpy.CalculateField_management(marks_viborg_new, "date",
    choose_date_2, "PYTHON_9.3")
103.

```

```

104. # Process: Join Field and add weather data
105. arcpy.JoinField_management(marks_viborg_new__2_, "date",
    viborg_weather_data, "date", "")
106.
107. # Process: Add Field Etc
108. arcpy.AddField_management(v3type_crop_viborg, "ETc", "DOUBLE", "", "",
    "", "", "NULLABLE", "NON_REQUIRED", "")
109.
110. # Process: Calculate Field Etc
111. arcpy.CalculateField_management(marks_viborg_ETc, "ETc",
    "Calculate_ETc(!ETc!, !AfgKat!, !ET_day! )", "PYTHON_9.3",
112. "def Calculate_ETc(ETc, AfgKat, ET_day):\n if ( AfgKat ==
    \"Majs\"):\n         return ( ET_day * 1.1)\n elif ( AfgKat ==
    \"Kartoffel\"):\n         return ( ET_day * 1.1)\n elif ( AfgKat ==
    \"Gulerod\"):\n         return ( ET_day * 1)")
113.
114. # Process: Add Field ETc week
115. arcpy.AddField_management(marks_viborg_ETc_values, "ETc_week",
    "DOUBLE", "", "", "", "", "NULLABLE", "NON_REQUIRED", "")
116.
117. # Process: Calculate Field ETc_week
118. arcpy.CalculateField_management(marks_viborg_ETc_week, "ETc_week",
    "[ETc]*7", "VB", "")
119.
120. # Process: Polygon to Raster ETc weekly
121. arcpy.PolygonToRaster_conversion(marks_viborg_all_info, "ETc_week",
    ETc_weekly, "CELL_CENTER", "NONE", "10")
122.
123. # Process: Polygon to Raster Eff_Rain_weekly
124. arcpy.PolygonToRaster_conversion(marks_viborg_all_info, "Ef_rain_w",
    Eff_Rain_weekly, "CELL_CENTER", "NONE", "10")
125.
126. # Process: Polygon to Raster ETc daily
127. arcpy.PolygonToRaster_conversion(marks_viborg_all_info, "ETc",
    ETc_raster_daily, "CELL_CENTER", "NONE", "10")
128.
129. # Process: Polygon to Raster Eff_Rain_daily
130. arcpy.PolygonToRaster_conversion(marks_viborg_all_info, "Ef_rain_d",
    Eff_Rain_daily, "CELL_CENTER", "NONE", "10")
131.
132. # Process: Project Raster Soil Moisture from Satellite Image
133. arcpy.ProjectRaster_management(Soil_Moisture_img,
    Soil_Moisture_ProjectRaster, out_coord_system, "NEAREST", cellSizeXY)
134.
135. # Process: Raster Calculator mask big slopes for soil moisture
136.
137. slop_moisture_ = Raster(Soil_Moisture_ProjectRaster) *
    Raster(slop_vib_Project)
138. slop_moisture_.save(slop_moisture)
139. # Execute SetNull
140. outSetNull = SetNull(slop_moisture, slop_moisture, "VALUE < 0")
141. # Save the output
142. outSetNull.save(soil_moisture_nd)
143.
144. # Process: Raster Calculator for Sandy, Clay and Loam soil type
145. inRaster1 = Raster(soil_type_recl_Project)
146. inRaster2 = Raster(soil_moisture_nd)

```



```

147.
148. # Process: Raster Calculator sandy
149. out1 = Con(inRaster1 != 1, 0, 1)
150. outCon = Con(((out1 == 1) & (inRaster2 <= 0.05)), out1)
151. outCon.save(sandy)
152.
153. # Process: Raster Calculator clay
154. out1 = Con(inRaster1 != 3, 0, 1)
155. outCon = Con(((out1 == 1) & (inRaster2 <= 0.17)), out1)
156. outCon.save(clay)
157.
158. # Process: Raster Calculator loam
159. out1 = Con(inRaster1 != 2, 0, 1)
160. outCon = Con(((out1 == 1) & (inRaster2 <= 0.08)), out1)
161. outCon.save(loam)
162.
163. # Process: Mosaic To New Raster Sandy, Clay and loam and find where to
    irrigate
164. arcpy.MosaicToNewRaster_management((sandy,loam,clay), env.workspace,
    where_to_irrigate, out_coor_system, "8_BIT_SIGNED", "10", "1", "LAST",
    "FIRST")
165.
166. # Create a raster with all marks with 0 value
167. raster_marks_ = ((Raster(ETc_raster_daily)) * 0)
168. raster_marks_.save(raster_marks)
169.
170. # Process: Raster Calculator_weekly to find the weekly ammount of
    irrigation
171. Irrigation_weekly_n_ = (((Raster(ETc_weekly)) -
    (Raster(Eff_Rain_weekly))) * where_to_irrigate)
172. Irrigation_weekly_n_.save(Irrigation_weekly_n)
173.
174. # Process: Raster Calculator Irrigation_daily to find the daily
    ammount of irrigation
175. Irrigation_daily_n_ = (((Raster(ETc_raster_daily)) -
    (Raster(Eff_Rain_daily))) * where_to_irrigate)
176. Irrigation_daily_n_.save(Irrigation_daily_n)
177.
178. # Process: Mosaic To New Raster for final result Irrigation_daily
179. arcpy.MosaicToNewRaster_management((Irrigation_daily_n,raster_marks),
    env.workspace, Irrigation_daily, out_coor_system, "32_BIT_FLOAT", "10",
    "1", "MAXIMUM", "FIRST")
180.
181. # Process: Mosaic To New Raster for final result Irrigation_weekly
182. arcpy.MosaicToNewRaster_management((Irrigation_weekly_n,raster_marks),
    env.workspace, Irrigation_weekly, out_coor_system, "32_BIT_FLOAT",
    "10", "1", "MAXIMUM", "FIRST")
183.
184. # Create a raster as a mask for soil moisture. has value 1 to areas
    with measurements and no data to no data
185. soil_moisture_nd_value1_ = ((Raster(soil_moisture_nd)) /
    (Raster(soil_moisture_nd)))
186. soil_moisture_nd_value1_.save(soil_moisture_nd_value1)
187.
188. # SpatialReference
189. out_coor_system =
    arcpy.Describe(soil_moisture_nd_value1).spatialReference

```

```

190.
191. arcpy.ProjectRaster_management(Irrigation_weekly, Irrigation_weekly_p,
    out_coor_system, "NEAREST", "10")
192. arcpy.ProjectRaster_management(soil_moisture_nd_value1,
    soil_moisture_nd_value1_p, out_coor_system, "NEAREST", "10")
193.
194. # Create a raster for final result Irrigation_weekly with values
    irrigation, 0 and no data
195. Irr_weekly_ = ((Raster(Irrigation_weekly_p)) *
    (Raster(soil_moisture_nd_value1_p)))
196. Irr_weekly_.save(Irr_weekly)
197.
198. # Create a raster for final result Irrigation_daily with values
    irrigation, 0 and no data
199. Irr_daily_ = ((Raster(Irrigation_daily)) *
    (Raster(soil_moisture_nd_value1_p)))
200. Irr_daily_.save(Irr_daily)

```