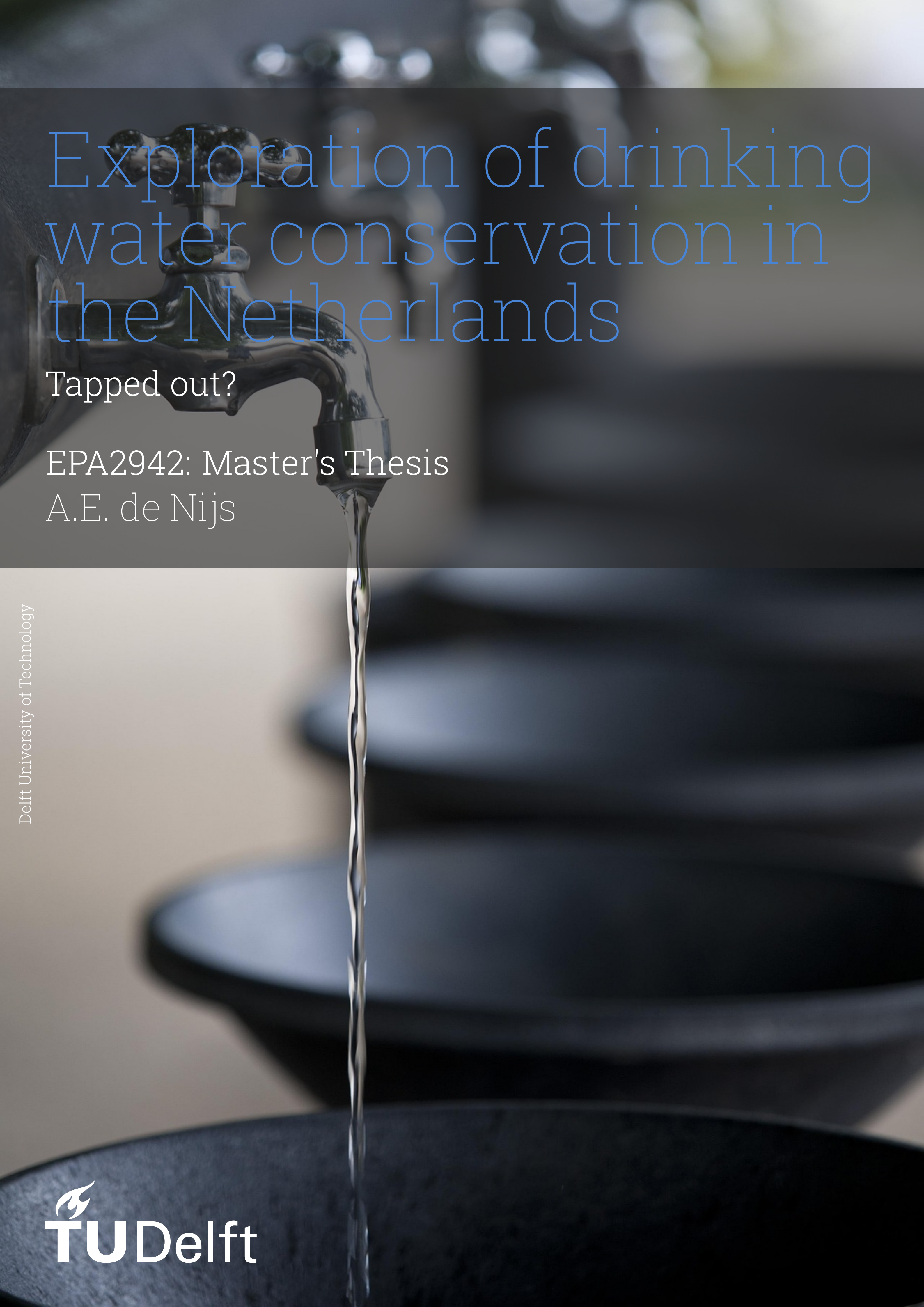


Exploration of drinking water conservation in the Netherlands

Tapped out?

EPA2942: Master's Thesis

A.E. de Nijs

Delft University of Technology

TU Delft

Exploration of drinking water conservation in the Netherlands

Tapped out?

by

A.E. de Nijs

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Friday June 6, 2025 at 15:30.

Student number:	4305906	
Project duration:	November 1, 2024 – June 6, 2025	
Thesis committee:	Dr. L. Scholten,	TU Delft, supervisor
	Dr. A. Ghorbani,	TU Delft

Style: TU Delft Report Style, with modifications by Daan Zwaneveld

TU Delft

Executive Summary

This thesis investigates to what extent and under what conditions it is likely that the Netherlands will meet its national target of reducing the average use of household drinking water from 129 in 2021 to 100 litres per person per day [lpppd] in 2035, as described in the National Plan of Action for Drinking Water Conservation (2024). This plan, published by the Dutch Ministry of Infrastructure and Water Management, was formulated due to growing concerns about anticipated drinking water shortages by 2030. Along with other policy measures, it led the Dutch government to attempt to shift toward demand-management measures. However, the effectiveness of the recently introduced and planned conservation measures remains uncertain, especially in light of various future developments.

This research assesses the potential impact of the proposed household-level policy measures using a model with a bottom-up demand component combined with a higher-level supply component. It applies the methodology of Exploratory System Dynamics Modelling and Analysis. An approach to execute a scenario-based exploration under deep uncertainty, simulating the combined effects of demographic dynamics, increased periods of drought, and economic growth in twelve potential future scenarios. As each of the ten drinking water companies operates within their own demographic and geographical context, they all face their own challenges, and different results are generated.

These results show that in none of the simulated scenarios for any of the drinking water companies, household demand actually falls to the policy target of 100 lpppd by 2035. Instead, demand only decreases to around 120 lpppd in 2035, with slight variations between regions and exogenous factors. However, that decrease persists and reaches 110 lpppd by 2050. Regarding the supply side of the drinking water distribution system, total household and industrial demand actually increases significantly in most scenarios up to 2030 due to population and economic growth. Regional variation further highlights that some water companies may encounter local shortages, even if national demand remains within its supply limits. A key outcome of this research is the demonstration of a policy-focused modelling framework that policy makers and researchers can reuse and extend.

These findings show that the current mix of policy interventions is unlikely to be sufficient to meet the policy objective set for the conservation of drinking water. Nevertheless, demand management is one element of several in drinking water management. The simulations also show that when drinking water companies are successful in developing additional fresh water sources the challenges of increasing drinking water demand could be met.

However, more impactful policy interventions could be made still if decisive action is taken in both the implementation and development of further demand management strategies. There remains a lot to gain by further exploring the limits and opportunities of the Dutch drinking water distribution system to support the implementation of more robust and future-proof policies.

Contents

1 Introduction

Sufficient availability of fresh water is of great concern worldwide. As a result of the increasing effects of climate change, economic development, and population growth, a discrepancy between the supply and demand for drinking water is on the horizon (Mazzoni et al., 2023). In the Netherlands, if no additional action is taken by 2030, it is expected that a yearly shortage of 100 million cubic metres will have to be compensated for (van Leerdam et al., 2023). The country is already experiencing more extreme weather patterns, particularly longer periods of drought that put additional pressure on freshwater sources and increase demand across both households and the economy (Baggelaar et al., 2022).

Each of the ten Dutch drinking water companies is facing their own challenges, ranging from rapid urbanisation to increased levels of salt intrusion on the coast. Looking at the sector as a whole, signs of strain on operational reserves are already becoming evident, casting doubt on the sector's ability to meet future peaks in demand. During recent droughts, they have already been forced to exceed the limits set in their extraction permits or drain their emergency reserves (van Leerdam et al., 2023). However, these limits are considered important remedies to ensure the recovery of the ecosystem and a fair distribution of the available freshwater sources. Historically, drinking water companies have responded to such shortages of clean drinking water by looking for new sources to extract from and optimising their own production processes (Van Alphen et al., 2022). Although that approach remains relevant today, there is now a growing recognition of the need to employ demand management as well. Consequently, current policy plans include not only supply-side measures, but also a more deliberate emphasis on demand reduction (Beleidsnota Drinkwater 2021-2026, 2021). A move which builds upon the advice of the Beleidstafel Droogte (2019) after the drought of 2018.

To implement demand management successfully, changing domestic consumption patterns poses a considerable challenge. Around 70% of the Dutch drinking water is used by households (Baggelaar et al., 2022), an environment in which changes are not easily implemented.

In particular, considering that in an urban environment, water pipes are built to last and that drinking water has been cheap and easily accessible for a long time. As a result, Dutch citizens often have little incentive to adapt their behaviour or invest in water saving technologies. In contrast, household usage has increased over the past decade, in part due to trends such as the popularity of rain showers (Bakker et al., 2022). Policymakers face the challenge of balancing the obligation to provide reliable and affordable drinking water with the approaching limits of the Dutch ecosystem.

To meet this challenge, the Ministry of Infrastructure and Water Management, together with relevant stakeholders in the drinking water sector, has formulated a national plan of action for the conservation of drinking water (National Plan of Action for Drinking Water Conservation, 2024). This plan builds on the strategic direction laid out in the Beleidsnota Drinkwater 2021-2026 (2021) and the Nota Water en Bodem Sturend (2023), which, among other water goals, outline long-term objectives for the national drinking water distribution system (DWDS). The national plan of action is designed to meet two specific targets for the conservation of drinking to be achieved by 2035: (1) reducing the average household use from 129 (in 2021) to 100 litres per person per day and (2) achieving a 20% reduction in water use per unit of economic output (National Plan of Action for Drinking Water Conservation, 2024) (Nota Water en Bodem Sturend, 2023). These goals are the quantified result of the ambitions laid out in the Coalitieakkoord 2021 – 2025: (2021) to make soil and groundwater the guiding factors in spatial planning, in this case specified for drinking water use. Through this political move towards demand management strategy, the stress on the supply in the Dutch DWDS should be decreased, thus accounting for the limits to growth that the ecosystem has been more explicitly showing in the past few years.

However, these goals do not seem to have been substantiated by any extensive evaluations as of yet. Considering that the drinking water system is both dynamic and complex, it is influenced by a range of uncertain future developments. In this context, the decision-making process could benefit from a modelling approach to help evaluate the effectiveness and sensitivity of the chosen policy directions. By simulating various future scenarios, stakeholders may be more able to determine under which conditions the challenge of potential scarcity in the Dutch DWDS could be met and which policy measure could prove the most effective.

This thesis explores the potential of such a modelling approach and focusses on the following.

To what extent and under what conditions are the objectives of the Ministry of Infrastructure and Water Management for drinking water conservation more likely to be met?

The National Plan of Action came into being through the cooperation of a broad coalition of stakeholders; it was a collaboration between different levels of government (national, provincial, and water boards), drinking water companies, and many representatives of relevant interest groups. The existing Dutch water governance system is multilevel, publicly financed, and, compared to many other countries, still relatively decentralised. The responsibilities for water management are shared among the aforementioned governmental bodies (Blankesteijn & Pot, 2024). Each of these parties has a role to play in achieving the chosen objectives for the conservation of drinking water. However, a great deal of uncertainty remains around which actions might prove to be the most effective in meeting the aforementioned objectives, as many of the proposed measures rely on yet-to-be-implemented research and policies. Exploring different future scenarios for the Dutch drinking water system can aid these actors in their current decision-making and support them in moving toward a more resilient and ecosystem-friendly drinking water system. In this manner, the research can account for significant uncertainty - not only in how the system will evolve under external pressures but also in how users will respond to new interventions. Scenario-based modelling offers a means of navigating this uncertainty and supporting robust and future-proof decision-making.

Given that households represent the largest share of drinking water demand in the Netherlands, on average 72,6% in the last 20 years (Centraal Bureau voor de Statistiek, 2023), and that more data and behavioural insights are available at this level (Bakker et al., 2022), this thesis will focus primarily on the use of household water. It takes a bottom-up modelling approach, combining high-resolution data on individual end-uses with broader system-level insights.

Traditionally, drinking water demand modelling had focused on supporting local infrastructure planning and network design (Van den Berg et al., 2022). By building hydrological models, the piping networks could be designed to match the capacity required to meet the needs of their locations. This was often achieved through the analysis of billing data or measurement campaigns, each of which occurred with low time resolutions (Cominola et al., 2015). Recently, the focus of water demand models has shifted to a finer resolution, both from a spatial and a temporal perspective. This change has been made possible by advances in monitoring, such as the increasing spread of pilots with smart drinking water meters (Cominola et al., 2015). These monitoring systems have allowed researchers to even grasp the individual end uses of water at a residential level, from minute to minute. Such higher resolution data allow for more accurate demand forecasting and create the possibility to further train models (Mauro et al., 2021) (Mazzoni et al., 2023).

Despite these developments, there is still a lack of integration of behavioural, policy, and system-level insights into cohesive integrated simulation frameworks. Various studies on water management have used system dynamics (SD) modelling, but socio-behavioural factors, especially those at the household level, have often been given little consideration (Phan et al., 2021). This limits the ability of these models to assess the real impact of demand management policies. Fuenfschilling and Truffer (2016), for instance, show in their study of the Australian urban water sector how large-scale investments in desalination strengthened the so-called ”Hydraulic Logic”, thereby side lining more flexible or sustainable options like wastewater recycling. This example shows how flexible, behaviour-oriented methods can be overlooked by institutional and technology path dependencies. Moreover, Koop et al. (2019) argue that water saving behaviour must be approached not merely through technical means, but by engaging the public through social norms, tailored feedback, and persuasive technologies. According to Phan et al. (2021), these behavioural factors are often excluded from existing SD models because they are too difficult to quantify. Similarly, Arnold and Wade (2015) argue that systems thinking remains under-used in modelling efforts due to fragmented frameworks and inconsistent methodologies.

This research aims to draw a connection between a lower resolution system's approach with more detailed end-use data. In this manner, it seeks to link specific demand-side policies directly to the behaviours they aim to influence. With this integrated approach of supply and demand this thesis intends to generate further insight on a lower resolution once again, but also to simulate the individual demand from a bottom-up perspective, allowing further investigation of the different end-uses (Auping, 2018). Consequently, the modelling of the policy measures can be directly linked to these end-uses as well, which will hopefully result in a more accurate representation of their potential impact.

Considering the objective of this thesis, the following sub-questions arise.

1. How is the national drinking water distribution system currently structured in terms of its sources, infrastructure, and managing organisations?
2. Which policies within the National Plan of Action for Drinking Water Conservation are most relevant to reduce drinking water demand?
3. How can the selected policy measures be linked to the known driving factors behind drinking water demand and be integrated into a System Dynamics model?
4. Based on these factors and policies implemented within the model, what behaviour is likely to be exhibited within the drinking water system up to 2050?
5. In what scenarios will the Ministry of Infrastructure meet its objectives and what challenges might arise?

The first three questions are focused on understanding the Dutch DWDS and on identifying the relevant policy instruments. This provides the foundation for system dynamics model by establishing what should be represented and how. The final two questions are aimed at exploring how the system may behave over time and under what scenarios policy targets might be met. Together, they provide a stepwise approach to answering the main research question through both conceptualisation and exploratory analysis.

As a thesis for the Engineering and Policy Analysis (EPA) programme, this research is focussed on a societal challenge. By applying system dynamics modelling to assess the impact of household-level conservation strategies, this research finds its societal relevance by linking technical modelling approaches to practical public policy decisions. In doing so, it demonstrates how evidence-based tools can support the design of robust, demand-side interventions in the face of environmental and societal developments.

2
Methods and Materials

In this chapter, a research outline is provided on how to investigate the questions posed in the introduction. It describes the methodological framework, the experimental setup and the data sources used in this thesis.

In doing so, it provides an answer to the 1st and 2nd sub-questions: *How is the national drinking water distribution system currently structured in terms of its sources, infrastructure, and managing organisations*? & *Which policies within the National Plan of Action for Drinking Water Conservation are most relevant to reduce drinking water demand*?

The chapter also presents an overview of the Dutch drinking water distribution system (DWDS) and it lays the groundwork for the model structure through the application of the XLRM framework. Together, these elements establish the basis for the system dynamics model and enable a structured exploration of policy impacts under conditions of deep uncertainty in the chapters that follow.

2.1. Conceptual and Analytical Approach

To address the dynamic complexity and deep uncertainty inherent in the Dutch drinking water system, this research employs Exploratory System Dynamics Modelling and Analysis (ESDMA). ESDMA is a hybrid approach that combines the methods of System Dynamics (SD) and Exploratory Modelling and Analysis (EMA). A combination that is particularly suitable for the grand societal challenges of future resource availability, such as water scarcity (Kwakkel & Pruyt, 2015). Through this approach, the policy interventions described in the National Plan of Action for Drinking Water Conservation (National Plan of Action for Drinking Water Conservation, 2024) are dynamically tested in a variety of plausible future scenarios. These simulations are carried out on a regional scale, for each drinking water company, and span the period from 2000 to 2050. With this focus, the model can build on earlier research and data on the Dutch drinking water sector before investigating long-term projections of the chosen measures for the different companies.

2.1.1. System Dynamics

System Dynamics (SD) is a modelling technique designed to analyse complex systems that feature feedback loops, delays, and non-linear behaviour (Sterman, 2000). It uses stocks and flows to simulate the accumulation and use of resources over time, which is well suited to simulate the different processes in the DWDS.

In this thesis, SD is used to develop a meta-model comprised of two main interconnected subsystems: a supply sub-model and a demand sub-model. The supply sub-model represents the yearly extraction, production, and distribution of drinking water by water companies. The demand sub-model simulates the daily use of drinking water of households, as well as the policies acting upon that demand.

These model components are conceptualised through the use of causal loop diagrams (CLDs) (Sterman, 2000). The CLDs help to visualise and identify the relationships between the key factors and the policies selected. The structure of the model is iteratively developed, taking lessons from previous SD models which were designed to tackle other challenges of resource scarcity (Van der Linden, 2020) (Kwakkel et al., 2013) (Auping, 2018).

2.1.2. Exploratory Modelling and Analysis

While SD is useful in capturing the structure of the Dutch DWDS and its behaviour over time, EMA addresses the deep uncertainty in the system, as introduced by Lempert et al. (2003):

> Deep uncertainty exists when analysts do not know or the parties to a decision cannot agree on (1) the appropriate models to describe the interactions, (2) the probability distributions to represent uncertainty about key variables and parameters in the models, and/or (3) how to value the desirability of alternative outcomes.

EMA enables the exploration of a wide ensemble of future scenarios, each defined by uncertain external developments, such as population growth, the impact of drought, and economic development. In addition, it also connects it to the internal dynamics of the system and the policies acting on it. As a result, this approach allows for a shift from simulating a single most likely future to a set of futures across a range of plausible conditions. Thus, allowing the investigation of the robustness of the policies, to accurately assess their ability to withstand or survive external shocks (Bankes, 2010).

2.1.3. The XLRM Framework

To provide structure to the exploration of drinking water conservation policies, this research adopts the XLRM framework introduced by Lempert et al. (2003), which disaggregates the system in four dimensions, where

- X (exogenous uncertainties) are the external factors beyond the direct control of policy makers.
- L (policy levers) are the measures available in the national plan of action.
- R (relationships) are represented through the model structure, which simulates the interactions within the Dutch DWDS.
- M (performance metrics) are indicators to evaluate the potential of policies, such as the demand for drinking water per capita.

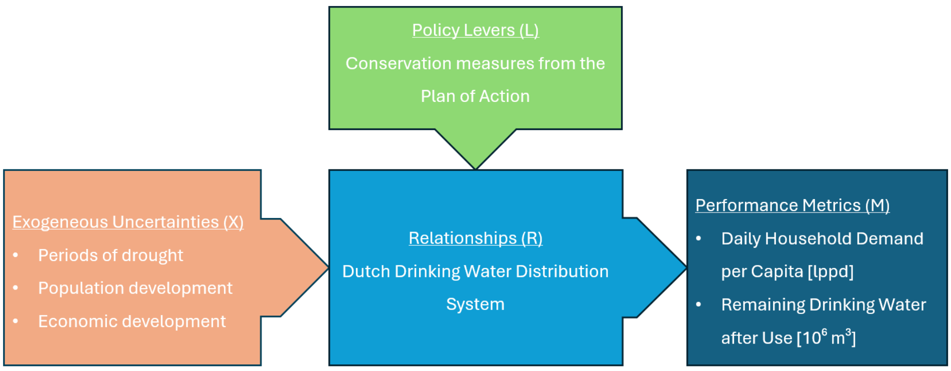


Figure 2.1: The XLRM framework applied to the Dutch DWDS.

By adopting this structural framework for this research, a wide range of possible future scenarios can be explored within a clear defined scope. Consequently, this makes it possible to assess the effectiveness and robustness of the proposed measures under different plausible future conditions.

2.1.4. Data Sources

This section outlines the key data sources used to inform the structure and behaviour of the model. These include baseline values for water demand, population forecasts, and supply chain components. The data underpins the system dynamics model described in the following section.

The model draws on multiple open data sources. The Central Bureau of Statistics (CBS) and Vewin, the national association of water companies, publish periodic reports on the end-use (Bakker et al.,

2022) and the production of drinking water (Vewin, 2022) in the Netherlands. These sources will be used to quantify the yearly flow of drinking water through the Dutch DWDS's and to create a bottom-up perspective on drinking water demand for an average household. The model will also implement other forecasts to account for external factors and stressors, such as population growth (Centraal Bureau voor de Statistiek, 2024) (PBL & CBS, 2022) and climate change scenarios (Royal Netherlands Meteorological Institute, 2024) (Baggelaar et al., 2022). Furthermore, many of the policies chosen are based on previous research on their individual effectiveness (Baltus et al., 2022), these articles will be reviewed to determine the precise bandwidths of their potential impact.

2.1.5. Simulation Setup

The SD model is implemented in Vensim Academic Pro, a widely used software environment that was developed to build and run system dynamics models (Ventana Systems, Inc., 2025). The simulation runs for a time span of 50 years (2000 - 2050) and uses a time step of 0.03125 to capture short-term feedback. The Euler numerical integration method is used for numerical approximation, in line with standard SD modelling practices.

2.1.6. Use of AI

For this thesis Chat GPT-4o was used to support in several tasks:

- Troubleshooting during modelling in Vensim & programming Python.
- Sparring to discuss feedback and to provide insights into potential structure for displaying the findings in the thesis.
- APA style citations through standardised .bib entries for the LaTex bibliography.

2.2. Overview of the Dutch DWDS

With the modelling approach defined, it is essential to provide a clear overview of the real-world system being simulated. The following section introduces a general description of the structure of the national drinking water distribution system, which will form the basis for the design of the model structure. In accordance with sub-question 1, this outline will focus on the different sources within, the infrastructure of, and the managing organisations of the Dutch DWDS.

2.2.1. Drinking Water in the Netherlands

Drinking water is defined in the Drinking Water Act as: Water intended, or partially intended, for drinking, cooking, or the preparation of food, as well as for other domestic purposes, with the exception of heated tap water, which is supplied through piping to consumers or other users (Drinkwaterwet, 2015).
Consequently, there are, of course, strict quality standards associated with this definition. In the Netherlands, there are 10 publicly owned companies that have the responsibility of producing drinking water that complies with those standards. Their shareholders are comprised of the municipalities and provinces to which they distribute their water.

Often these companies operate across provincial or municipal borders, as they are the result of nearly 200 years of developments and consolidations. Moving from the first drinking water company, the Dune Water Association, in the area of Amsterdam in 1854 to over 200 drinking water companies in the 1950s to the 10 that are still in existence today (Van den Berg et al., 2022), as shown in Figure 2.2. The last merger between companies took place in 2006; for clarity's

Figure 2.2: An overview of the ten water companies in the Netherlands (Dumaij & Heezik, 2012).

sake, in the modelling process these predecessors are treated as if they were already a part of the final ten in the year 2000. In this manner the reported data from the past will be aggregated and can be brought in line with the organisational reality of today.

In addition to producing clean water, drinking water companies are also entrusted with distribution of water and maintenance of the piping network required for this task. Drinking water is sold to both private citizens and companies in a wide range of sectors. However, since the water companies are publicly owned, they are not allowed to make a profit on this sale. As a result, the Dutch DWDS functions as a closed-loop system, where the rates are solely based on the costs required to keep their processes up and running.

As each drinking water company faces its own geographic and demographic conditions, the model is designed to be flexible and adaptable across the ten companies by allowing for input variation.

2.2.2. Freshwater sources

Vewin, national association of water companies, distinguishes four main sources of drinking water: groundwater, riverbank filtrate, surface water, and natural dune water (Vewin, 2022). The use of these sources varies significantly by region, as it is highly dependent on the geographical context of their respective drinking water companies, as can be seen in Figure 2.2 and Table 2.1. For example, groundwater is the most common water source in the more landlocked areas of the Netherlands, whereas in the West in the Dutch delta, companies are more dependent on extraction from rivers. In addition, along the coast there are areas where water is taken directly from reservoirs in the dunes, and along rivers the riverbank acts as a filtrate for indirect extraction from those rivers. In addition, what is important to consider is that the extraction of natural dune water involves only natural reservoirs, not the water that is filtrated through the dunes, which happens at a much larger volume.

Table 2.1: Fresh water extraction for drinking water in 2020 [10^6 m^3] (Vewin, 2022). KWR is a subsidiary of PWN & Waternet, it produces drinking water, but does not distribute.

Company	Total	Ground-water	Riverbank filtrate	Natural dune water	Surface water
Brabant Water	206	206	-	-	-
Dunea	82	-	-	-	82
Evides	209	17	-	1	191
Oasen	47	6	41	-	-
PWN	39	5	-	2	31
Vitens	394	381	13	-	-
Waternet	37	-	-	12	25
Waterbedrijf Groningen	47	40	-	-	7
WMD	37	37	-	-	
WML	78	54	25	-	
KWR	158	-	-	-	158
Netherlands	1335	746	79	15	494

2.2.3. Existing infrastructure

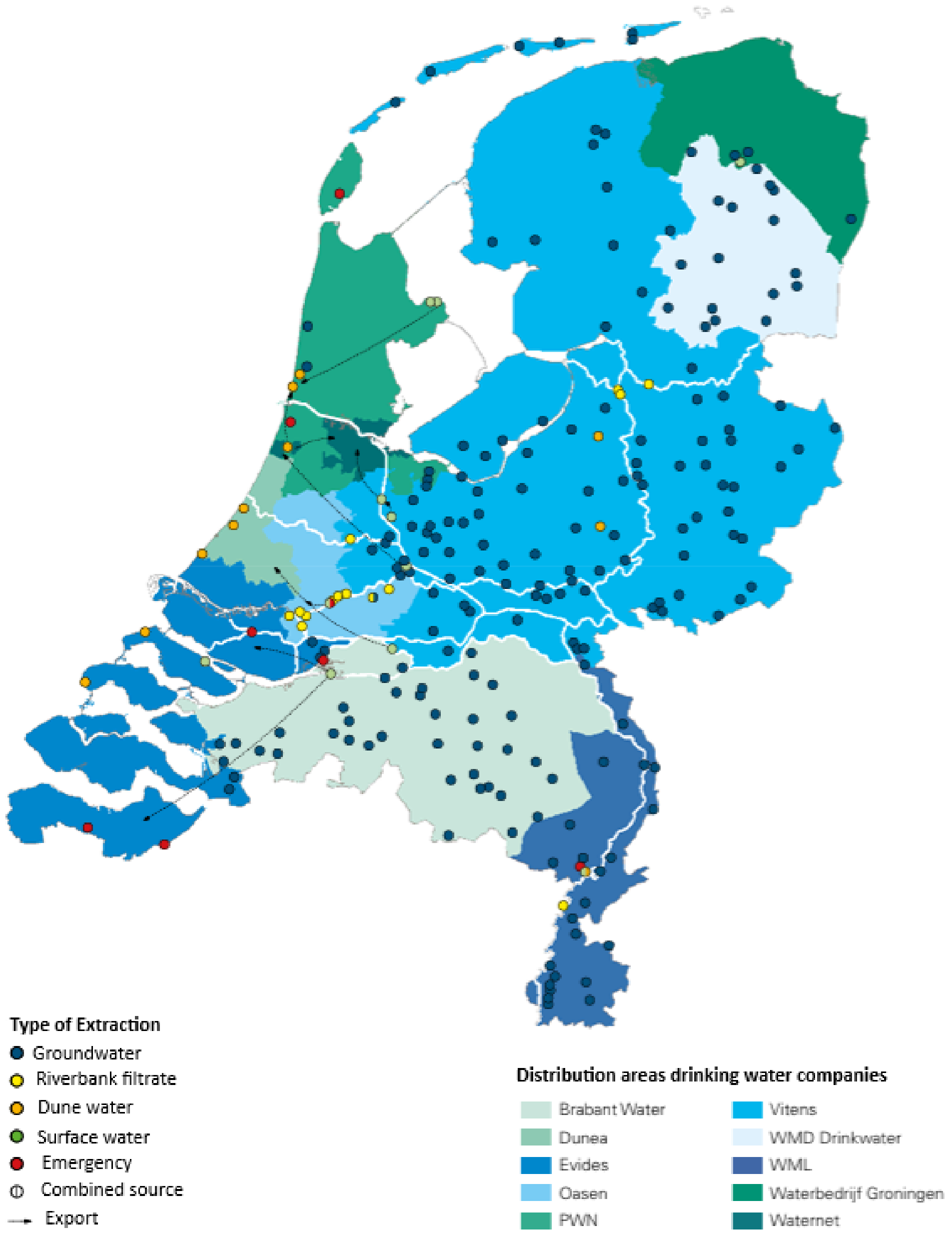


Figure 2.3: An overview of the 225 extraction sources across the Netherlands in 2020, adopted from Vewin (2022)

.

The drinking water companies maintain a network of piping of approximately 120,000 kilometres and manage 23,000 hectares of nature reserves in which their extraction and filtration sites are located (Blankesteijn & Pot, 2024) (Vewin, 2024c). In total, there are 225 extraction locations in the Netherlands, as can be seen in Figure 2.3.

After extraction, the water is either infiltrated into the dunes before purification takes place or transported directly to the purification facilities. Some of the extracted water is purified by other companies, prepared for export, or cleaned to industrial standards (Vewin, 2022). As this would overcomplicate the exploration of drinking water conservation, these dynamics are left out of the scope of this research.

After these purification and production processes have run their course, drinking water is ready for distribution. In the Netherlands, there are approximately 8.7 million clients connected to the DWDS, which includes both household and industrial consumers.

2.2.4. Operating agencies

The relevant stakeholders that are central to the operation of the Dutch DWDS are briefly introduced below, based on the Oxford Encyclopedic Chapter on Water Governance by Blankesteijn and Pot (2024).

- The Ministry of Infrastructure and Water Management is the central governing body responsible for effective water management on a national level, including both water availability and flood protection. It supervises the drinking water sector, develops national policy, and coordinates policy efforts at a regional level. In that capacity, it provides the regulatory framework in which the Dutch DWDS has to operate. In particular, the Drinkwaterwet (2015) and the Drinkwaterbesluit (2011), which define the guidelines and quality standards that the drinking water sector must meet. However, it is also responsible for setting overarching policy goals, such as those investigated in this thesis.
- The Directorate General for Public Works and Water Management (*Rijkswaterstaat*) is the executive agency of the Ministry. With regard to drinking water, they manage a lot of national surface water bodies. Accordingly, they are in charge of extraction permits and deciding on how the available fresh water in those bodies is divided.
- The 12 provinces of the Netherlands are the regional overseers of their own water systems and are in charge of regional planning. They grant permits for the large-scale extraction of groundwater for the production of drinking water and they need to ensure groundwater quality. In this capacity, they set extraction limits that have to ensure a proper balance in the allocation of groundwater between different sectors, from agriculture to households. Quite often, along with municipalities in their borders, they are the shareholders of the publicly owned drinking water companies.
- The water boards are regional water authorities that are responsible for maintaining water levels, water quality, and waste water treatment. They must coordinate with the provinces they reside in and the municipalities within their borders on how to carry out these tasks, as they are not responsible for spatial planning. As a stakeholder in the DWDS, they are focused on the aftermath of drinking water use, as they both collect and treat waste water after it is used.
- As introduced previously, there are ten drinking water companies that are legally obliged to ensure safe and affordable drinking water access. They maintain the infrastructure of the Dutch DWDS and extract, produce, and distribute drinking water. They are the parties who are mainly responsible for the implementation of the proposed conservation measures.

Each of these stakeholders has a central role in the management of the DWDS and the implementation of conservation measures within it. In addition, there is a fair amount of overlap in responsibilities. Therefore, sound collaboration is of great importance.

2.3. Development of the model according to the XLRM framework

Having established the institutional and physical context of the Dutch DWDS, the next step is to capture these elements in the modelling framework. Following the XLRM framework as shown in Figure 2.1 in section 2.1.3, this section expands on the different elements that are central to that framework.

2.3.1. Policy Levers

The policy levers outlined in the National Plan of Action for Drinking Water Conservation (2024) were developed by the Ministry of Infrastructure to respond to the imminent challenge of water scarcity. For this research, a subset of the most relevant and quantifiable policies has been selected from the 24 demand management policies proposed for household use of drinking water.

The selection of drinking water conservation measures for the DWDS model is based on the following three considerations.

- Specificity: Several of the proposed measures are still in the exploratory and research phase. Due to the preliminary nature of these measures, no statements can yet be made on what steps the government will take on those policy directions. This consideration results in the following exclusions:
  - Policy 4: Stimulating the purchase / use of water-saving techniques (subject to the availability

of funds). This policy is excluded from the simulation, since it has not yet specified which measures will be stimulated and through which means.

- Policy 6: Drawing lessons learnt from other policy fields (energy & best practices abroad). This policy is not implemented in the model, as it is very unclear which policies could be derived from the lessons learnt.
- Policy 9: Investigation of the effects and efficacy of pricing/taxing the use of drinking, ground, and surface water. Pricing and taxing are not included because there are many financial mechanisms that could be included. Focussing on a single example or combining all possible measures will strongly impact the effect of such a policy.
- Policy 11: Research of the possibility of the introduction of drinking water restrictions in crisis situations. This policy still lacks clarity with regard to what restrictions might be mandatory and what would constitute a crisis situation. Consequently, the frequency of use and the impact of restrictions are still too difficult to estimate.
- Policy 18: Knowledge sharing on the subject of drinking water conservation within the domain of construction and renovation. Similarly to policy 6, it is not possible to determine what conservation measures might be shared between different actors in the construction sector.
- Policy 20: Exploration of the possibilities for stimulation / subsidisation of drinking water conservation within the domain of construction and conservation. Likewise to policy 4, it is complicated to assess which specific conservation measures might be implemented.

• Direct relevance to drinking water conservation: Some of the measures focus on investigating the impact of conservation on piping or secondary benefits. Since the following measures do not directly influence household drinking water consumption, they fall outside of the scope of this analysis:
  - Policy 7: Investigating the impact of a decrease in drinking water use on the water cycle.
  - Policy 10: Creating an overview of existing pilot projects.
  - Policy 19: The execution of a social cost benefit analysis (SCBA) for future-proof construction, including the conservation of drinking water.

• Compatibility: In cases where multiple policies have the same focus, they are consolidated into a single policy lever. For instance, the introduction of smart water meters is first explored through pilots. Then, the logical next step could be taken to implement them in both existing and newly constructed housing. This approach has resulted in the following combinations:
  - Framing campaigns:
    * Policy 1: Execution of the plan of action for water-conscious behaviour.
    * Policy 2: Execution of regional campaigns.
    * Policy 5: Encouraging the conservation of drinking water in climate adaptation policy.
    * Policy 12: Expanding on the plan of action for water-conscious behaviour.
    * Policy 14: Investigation into certifying 'water-saving products' (e.g. an A-label).
  - Comparative invoices:
    * Policy 3: Direct consumers to the benchmark and give water conservation tips on the invoice.
    * Policy 15: Research of the possibility of including the water treatment tax in the drinking water invoice.
  - Mandatory Smart Water Meter (SWM)
    * Policy 8: Conducting pilots with smart water meter placement.
    * Policy 13: Introduction of smart water meters in existing housing, based on pilot results.
    * Policy 23: Introduction of smart water meters in new housing, based on pilot results.
  - Mandatory rainwater harvesting

    * Policy 16: Follow-up investigations into drinking water conservation (e.g. rainwater harvesting and greywater use) in the built environment, so that ultimately a sound consideration can be made for a possible obligation in the build code.
    * Policy 21: Investigate whether drinking water companies can establish requirements for newly built houses to connect to their networks (national norm NEN6922).
    * Policy 22: In view of the results of the investigations mentioned above, decide on adjusting the build code to make drinking water conservation mandatory.
  - Mandatory re-use of household water
    * Policy 17: Exploration of the possibility of changing the Drinking Water Act with regard to the definition of household water.
    * Policy 24: Examining the possibility of adjusting the national norm for drinking water installations (NEN 1006).

In Table 2.2 a brief overview of the selected combinations of policies is given, which will be expanded further in the following subsections. The factors for the projected reduction in demand are largely based on the work of Koop et al. (2023), since they did an extensive review of the literature of water conservation policy measures and adapted them to the Dutch context.

Table 2.2: Overview of policy levers selected for the model

Policy name	Demand reduction	Modelling choices made	Reference
Framing campaigns	2.79%	- The reduction starts in 2022. - Directly impacts the total daily demand for household use, not specific end uses.	Koop et al. (2023)
Comparative invoices	2.64%	- A combination of invoice communication measures. - The reduction starts in 2024. - Directly impacts the total daily demand for household use, not specific end uses.	European Parliament & Council (2020) Koop et al. (2023)
Mandatory Placement of Smart Water Meters (SWM)	4.88% per SWM	- After completion of the pilot a shift is made to SWMs. - The reduction starts in 2027. - Impact depends on the total amount of households connected to the specific DSDW.	Koop et al. (2023)
Mandatory rainwater harvesting	Not fixed	- Policy decision to change build code. - Reduction starts in 2028. - Impact depends on the yearly construction rate of new housing. - Focus on end uses: toilet, washing and outside.	-
Mandatory re-use household water	Not fixed	- Choice to amend water act. - Reduction starts in 2028. - Impact depends on the yearly construction rate of new housing. - Focus on end uses: toilet, washing and outside.	-

Behavioural Policies

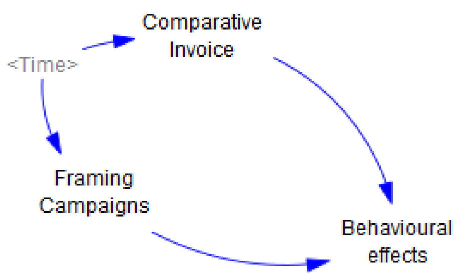


Figure 2.4: The behavioural policies implemented in the model.

Several of the proposed measures are centred around regional and national campaigns to achieve behavioural change at the consumer level. As some of these campaigns were already underway by 2022, shortly after the last end-use survey, it is assumed that this marks the beginning year for these policies. In addition, to maintain their long-term impact, it is assumed that the campaigns are periodically renewed until 2050, as repeated exposure is essential to consolidating behavioural change. It is also presumed that these campaigns are implemented with the most effective understanding of convincing framing language (Koop et al., 2023).

Since January 2023, water companies have been required to provide additional comparative insights on their customers' invoices, as stipulated by the European Parliament & Council (2020). This has been included as a policy measure in the action plan, which makes sense considering research has shown that when demand management communication includes a normative message it is much more effective than purely informational campaigns (Koop et al., 2019). However, it takes time for the message to take hold, and therefore this policy is assumed to only become effective after one year, from 2024 onwards. (Koop et al., 2023).

In Figure 2.4 the combination of the two behavioural policies is shown, which together directly act on the daily household demand per capita in the model. It is assumed that the policies have a broad impact on all uses, as it is too difficult to assess whether they will result in specific water-saving behaviour. Especially since this policy lever contains several types of communication. For example, while some might be tempted shower for a shorter time by one campaign, others are more likely to respond to another and use the dual flush of the toilet more consciously.

Smart Water Meters

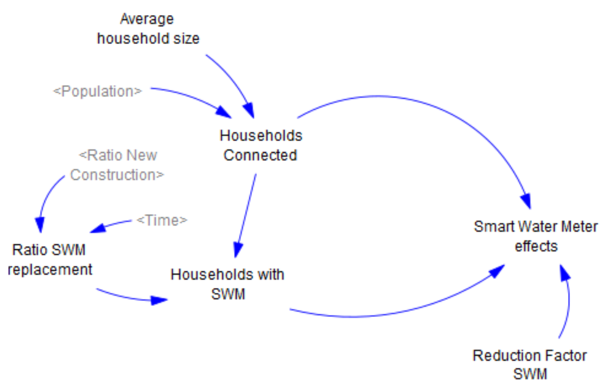


Figure 2.5: The transition to SWMs as implemented in the model.

Smart water meters (SWMs) are advanced measurement devices that digitally track household water consumption in real time and transmit usage data to consumers and drinking water companies (Cominola et al., 2015). Unlike traditional analogue meters, SWMs allow for detailed monitoring of drinking water use, early detection of leaks, and tailored feedback to users, thus encouraging more efficient consumption habits. The Plan of Action includes a phased approach to the implementation of SWMs. First, for the period of 2024-2025, small pilots with SWM's will be carried out by several drinking water companies. Then based on the outcomes of these pilots, a decision will have to be made in 2026 and 2027, regarding the widespread replacement of their customers' analogue meters.

The projected transition to SWMs shown in Figure 2.5 is founded on two factors. The first is the number of households connected to the DWDS of that company, which is determined by dividing the current population by the expected national average household size for that year (Centraal Bureau voor de Statistiek, 2019) (Centraal Bureau voor de Statistiek (CBS), 2023). The second factor is the rate of new housing construction in that area (Centraal Bureau voor de Statistiek, 2025), as this policy is aimed at both existing and new customers of the drinking water companies. The modelling choices made for these two datasets can be seen in Tables A.1 and A.2, and the actual construction rates used in Table 2.3.

Table 2.3: The average yearly construction rate for each drinking water company.

Drinking water company	Average rate [%]
Brabant Water	0.89
Dunea	0.74
Evides	0.71
Oasen	0.74
PWN	0.85
Vitens	0.87
Waterbedrijf Groningen	0.70
Waterleiding Maatschappij Limburg (WML)	0.46
Waterleiding Maatschappij Drenthe (WMD)	0.55
Waternet	0.85

Thereafter, it is assumed that both these practices start in 2027, thereby increasing the ratio of households with an SWM with the combined rate of replacement due to the age of the old analogue meter in existing housing and the placement of SWMs in new housing. On average, a water meter needs to be replaced every 20 years (National Plan of Action for Drinking Water Conservation, 2024), so a yearly replacement rate of 5% is used for existing housing. Finally, the total reduction factor for SWMs is calculated by multiplying the reduction rate in Table 2.2 by the ratio of households with an SWM divided by the total number of households connected to the specific DWDS.

This policy also directly affects the daily household demand per capita, since studies have shown it will incentivise citizens to conserve their drinking water in general (Koop et al., 2023).

Regulatory Policies

In 2024 & 2025, two regulatory measures are being investigated as potential conservation policies. This research serves as a follow-up to the exploration of the potential of rainwater harvesting and reuse of household water by Hermans et al. (2023). This previous report left some unanswered recommendations and questions which need to be followed up before the potential of these construction requirements can be implemented.

Currently, private citizens are already permitted to install rainwater harvesting devices on their properties, ranging from a simple barrel to a large water tank (Hermans et al., 2023). These systems can support uses that do not require the cleanest water, such as watering plants, cleaning a car, or flushing a toilet. However, the policy measure formalised in the National Plan of Action for Drinking Water Conservation (2024) aims to incorporate these rainwater harvesting devices into the building code. Thus, making the inclusion of rainwater harvesting infrastructure mandatory for each new appropriate property built in the Netherlands. For this measure, the assumption is made that it can only be effective for low-rise housing, as it requires private outdoor space to capture the rain successfully.

The second potential regulatory measure is the expansion of the definition of water in the Drinkwaterbesluit (2011). A change that would allow for the reuse of bath and shower water for the same uses as those for rainwater harvesting. In addition, this change will also aim to reintroduce a wider range of potential uses. After detailed research, as mentioned in the National Plan of Action for Drinking Water Conservation (2024), on potential health risks, household water using washing machines are very likely be allowed once again. For this model, it is assumed that this is a measure uniquely suited for new high-rise buildings and therefore acts as an alternative to rainwater harvesting for these construction projects.

As both policies require the adaptation of legislation and a choice in 2026-2027 by a decision-maker, it is assumed that these policies will become effective in 2028 at the earliest. In Figure 2.6 the regulatory component of the model is shown.

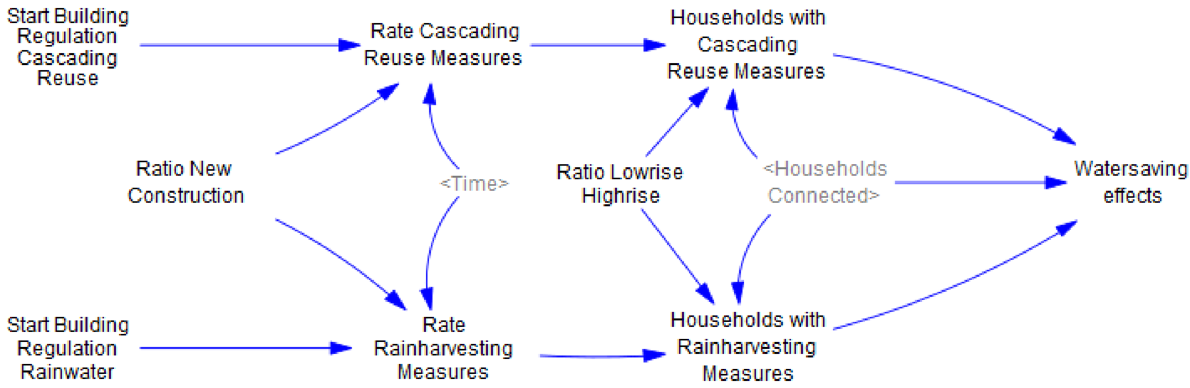


Figure 2.6: The regulatory policies implemented in the model.

Once again, for the calculation of the reduction effects, the ratio for new construction is used, which is shown in Table 2.3. Only now is this rate aggregated from year to year, depending on the start of the policy. In this way, the rate of households with these measures increases linearly after implementation. Then a ratio of two low-rise buildings to one high-rise building is applied to decide which houses receive cascading measures and which houses receive rain-harvesting measures, following the example of Koop et al. (2023). Afterwards, the calculated water-saving effects are connected to their respective end-uses, which are outlined above.

2.3.2. Uncertainties

This study accounts for several exogenous drivers that are not completely predictable, but still have a strong influence on the dynamics of the demand for drinking water. The uncertainties used are based on national scenarios and projections. They are incorporated into the model to reflect a range of plausible future developments.

Economic Growth

Economic growth is the driving factor behind the industrial use of drinking water and is therefore directly related to industrial demand. Although the industrial side of the DWDS is not the main focus of this thesis, it is still important to consider its impact on drinking water availability. An increase in the economic use of water will limit its availability to households. Consequently, a component for industrial demand at a more abstracted level will have to be added to the model to account for that effect.

For the first half of the simulation industrial demand is modelled on the basis of historical data, for that time frame, it is assumed that all drinking water produced for the industrial sector is used for its intended purpose. From 2025 onwards, economic development is approximated on the basis of the prognosis by Baggelaar et al. (2022), the result of which can be seen in Table 2.4. This prognosis is the aggregation of the expected development for different fields of economic activity. The analysis of the dynamics of different sectors is based on the 2015 welfare and environmental scenarios, as developed by CPB and PBL (2015).

Table 2.4: The two potential economic development paths.

Year	Min	Max
2000	1	1
2025	0.967	1.040
2030	0.937	1.083
2035	0.917	1.119
2040	0.897	1.156
2045	0.897	1.149
2050	0.869	1.197

These two plausible economic development paths account for the complete range of industrial sectors to which drinking water companies distribute. Considering the fact that the demand management policies from the National Plan of Action for Drinking Water Conservation (2024) for industry are left out of the scope of this thesis, a more targeted economic impact at specific end uses is not possible.

Population Growth

Population growth is considered to be one of the main drivers behind the growth in national drinking water demand (van Leerdam et al., 2023). However, on a more regional level, its effect can vary considerably. The dynamics of urbanisation will put a strain on drinking water companies companies in strongly urbanised areas, whereas they might relieve pressure on companies outside of those areas.

Drinking water companies do not monitor the dynamics of population growth in their distribution areas. Therefore, for the factor of population growth, an aggregation of Census data and the population prognosis is used (Centraal Bureau voor de Statistiek, 2024) (PBL & CBS, 2022). From 2000 onwards, municipal population data is aggregated on the basis of the distribution areas of each of the drinking water companies (Vewin, 2022) (Vewin, n.d.).

As a result, for each drinking water company, three different trajectories can be identified. An expected population development according to the prognosis and a lower & a higher path according to a 67% forecast interval. The process for which, through Jupyter notebooks, can be seen in Appendix B. In Figure 2.7 the three potential developmental paths for population growth are shown for each drinking water company.

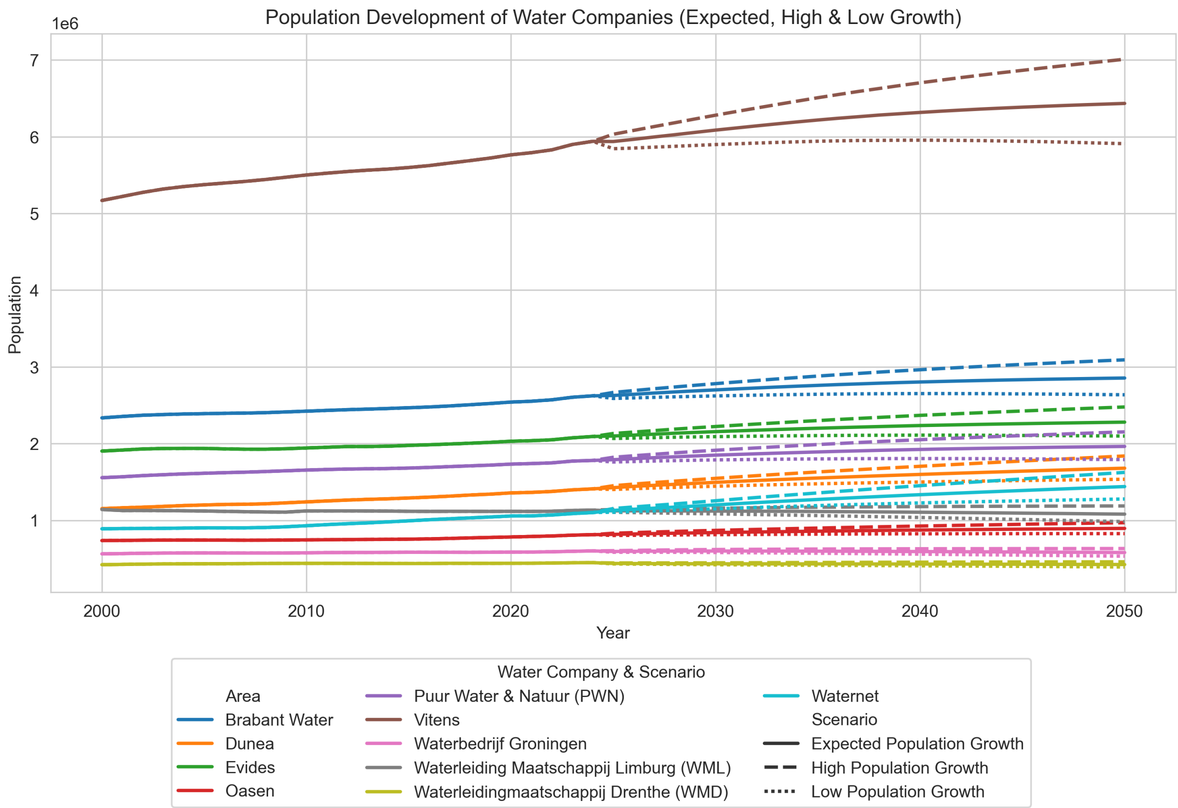


Figure 2.7: The potential population development of the ten drinking water companies.

Drought

The third uncertainty studied is the occurrence of droughts, as periods of drought are more likely to occur in the near future (Beleidstafel Droogte, 2019). For this model, periods of drought have mainly behavioural effects on water demand. Short-term drought events could also cause temporary shortages, as the dried-up supply may not be able meet the increased demand. However, for this thesis, seasonality is not considered. Therefore these short term shortages won't be captured in the model.

The summer of 2018 had a demand increase effect of 3,7%. When this uncertainty is implemented in the model, it is assumed that every summer will be like that one, adding the said percentage to the existing household demand.

2.3.3. Performance metrics

To evaluate the impact of different policy measures and uncertainty scenarios, two key performance metrics are of interest, the Daily Household Demand per Capita [lpppd] and the Annual Amount of Remaining Drinking Water after Use [10^6 m^3 p/y]. These indicators focus on the effectiveness of the policies selected and the system-wide availability of drinking water. Together, they provide a comprehensive picture of how well the drinking water system could respond to future challenges in demand and supply.

The first performance metric has a direct link with the policy targets laid out in the National Plan of Action for Drinking Water Conservation (2024). The second performance metric represents the larger overarching target of having a more robust DWDS. It captures the volume of drinking water that remains available in the DWDS after domestic and industrial use. The metric is calculated by subtracting total household & industrial consumption from the total amount of fresh water extracted by the drinking water company. A consistently low value might indicate a growing discrepancy between demand and supply, which would require stronger demand management policies.

Table 2.5: Overview of the performance metrics of interest in the model.

Outcome	Unit
Daily Household Demand per Capita	[lppd]]
Total drinking water consumed by households	[10^6 m^3 p/y]
Total drinking water consumed by industry	[10^6 m^3 p/y]
Total fresh water extracted by drinking water companies	[10^6 m^3 p/y]

3 Results

Building on the methodological foundation outlined in the previous chapter, this chapter presents the implementation and outcomes of the developed simulation model.

It addresses the 3rd and 4th sub-questions: *How can the selected policy measures be linked to the known driving factors behind drinking water demand and be integrated into a System Dynamics model?* & *Based on these factors and policies implemented within the model, what behaviour is likely to be exhibited within the drinking water system up to 2050?*

The chapter first outlines the structure of the model and the assumptions supporting its design. It then explores how the system behaves under a range of future scenarios.

3.1. Model Structure

The drinking water model developed for this thesis is structured around two main interacting components, the drinking water distribution system and the demand for drinking water. Together, these components simulate the supply-demand dynamics in the Dutch DWDS under different uncertainty scenarios. This section outlines the core assumptions, sub-models, and choices that are essential to the model design. In Figure 3.1 an overview is given on how the main components are related to one another and what elements they contain.

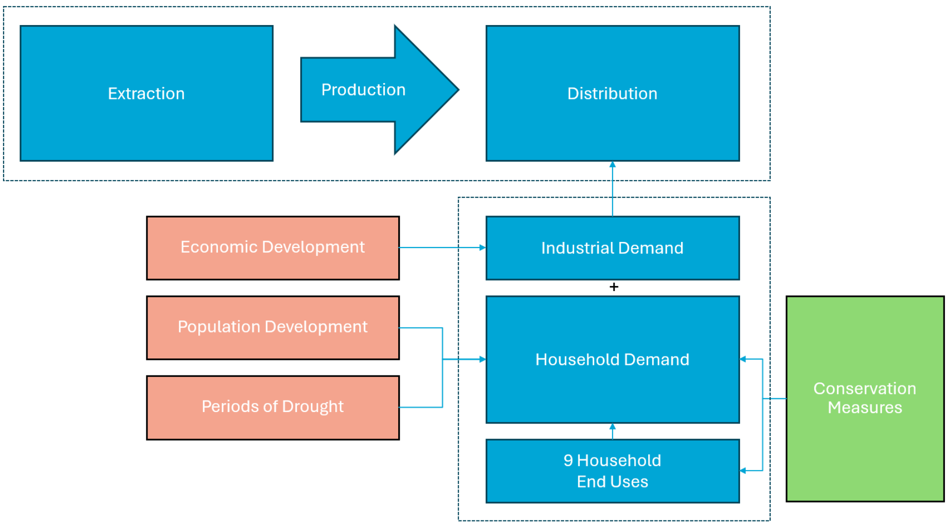


Figure 3.1: Overview of the structure of the supply-demand model and its two interacting components, at the top the DWDS sub-model and at the bottom the drinking water demand sub-model. Acting upon the demand sub-model, in orange the exogenous drivers have been added, as well as the policy levers in green.

3.1.1. Drinking Water Distribution System

The Dutch DWDS and thus the regional DWDS's as well consist of a number of stages: extraction from fresh water sources, production of drinking water through purification, and distribution through piping networks. In this section, an overview of the sub-models that conceptualise each stage of the drinking water supply chain is given.

Extraction

For each of the ten drinking water companies, data was collected from different reports throughout the past two decades. The modelling decisions and sources are shown in Table A.3 and in Figure 3.2. The model builds on these historic data points to obtain an accurate indication of the extraction patterns from 2023 onwards. As stated in Section 2.2.2, there are four types of fresh water sources. This component of the model allows for flexibility in the different methods of extraction by each drinking water company.

In addition, for this component of the model, it is assumed that all the plans formulated in the Action Plan for the availability of fresh water sources are successfully implemented (Vewin, 2025). As a result, investment costs and permitting challenges are excluded from the simulation.

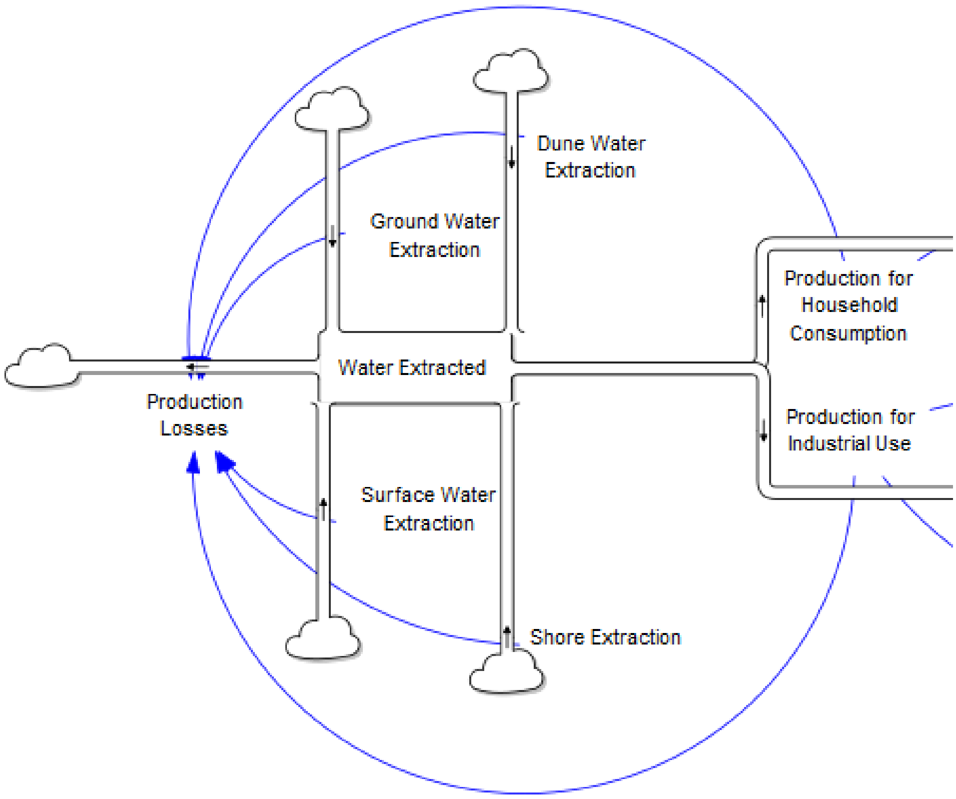


Figure 3.2: The extraction and production component of the model

Production

After extraction, drinking water companies use a wide range of techniques to process water to ensure that it meets the aforementioned quality standards for human consumption. As this research focusses on the balance between fresh water availability and drinking water demand, these processes are not modelled in detail. In Figure 3.2 the three relevant production flows are included in the model. Each water company supplies both civil and industrial consumers. Similarly to the extraction data, the patterns in division between the two groups are abstracted from the historical data of the past two decades. The modelling choices are shown in Table A.4. Water companies keep an operational reserve on hand, experience some leakages, and sell some of their extracted water to other companies. All of these streams are combined in the production loss flow, simply by subtracting the data for production from the data for extraction.

Distribution

In the next stage of the DWDS, drinking water is distributed to customers of the water companies. The division between household consumers and industrial clients is, similarly to the extraction of fresh water, highly reliant on the local context. Some areas simply have more economic activity or are more urbanised than other regions. In Figure 3.3 the distribution component of the model is shown, where the two supply streams are linked to the demand side of the drinking water system.

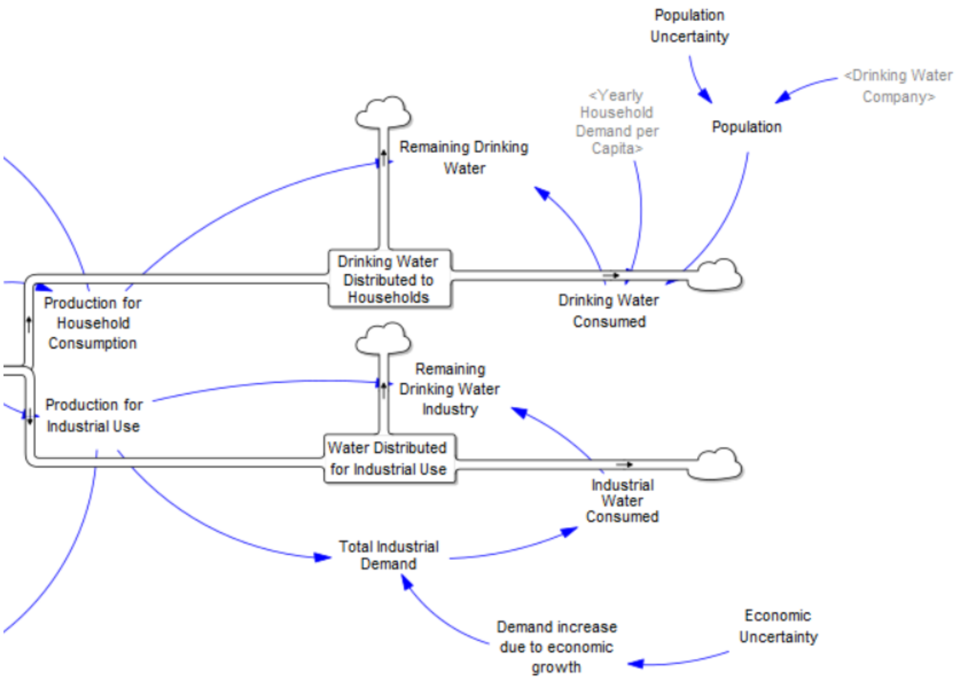


Figure 3.3: The distribution component of the model

The quantity of water distributed to households is based on production data as laid out in Section 3.1.1, but now it is coupled with the bottom-up demand per capita from the other side of the DWDS model. As a result, it needs to be multiplied with the yearly population data of that specific water company's distribution area. After demand subtraction, the remaining drinking water is also monitored, since this is an important outcome of the model. Especially when considering that if there is a supply shortage, this flow will show negative values, which would be a clear indicator of imbalance within the system.

For the amount of drinking water used by the industrial sector, a simpler approach is taken. Since the focus is on household conservation measures in the National Plan of Action, it is assumed that for the first half of the simulation, the industrial demand is equal to the amount of drinking water produced for that sector. However, the remaining flow of drinking water is still included, as it is essential to account for the economic impact in the second half of the simulation, as introduced in Section 2.3.2. Once again, this flow will display a potential discrepancy between water availability and demand. In scenarios with economic growth, it will show a shortage and in times of economic contraction a surplus.

3.1.2. Household Demand

The demand side of the model adopts a bottom-up, end-use approach, breaking down total household consumption into its underlying end-uses. In this section, an overview of how the demand for drinking water for households is modelled is presented. The design of this model is based on the end use surveys conducted by the Central Bureau of Statistics, where they have identified three driving factors behind different end uses of drinking water by households (Bakker et al., 2022), which are shown in Figure 3.4.

The degree of presence represents the percentage of Dutch households that own a device that matches that specific end-use. The frequency of use stands for how often that device is used per day. Similarly, the capacity per use signifies how many litres of water is required for a single use of that device. By

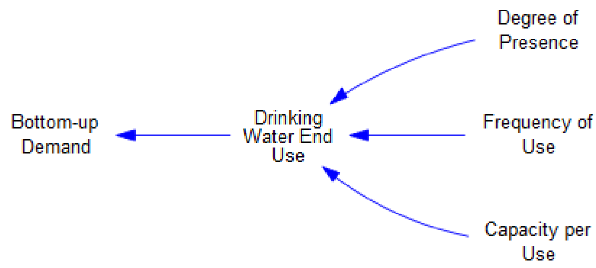


Figure 3.4: The driving factors behind drinking water end uses.

multiplying these three factors, the demand for drinking water per capita per day can be determined for the specific end use in question.

End use

For the selection of individual end uses, the CBS report by Bakker et al. (2022) is used as a basis for the bottom-up design of the demand for drinking water. Over time, there have been a few adaptations to the factors investigated in the surveys, but it is assumed that the most recent report matches best the policy directions laid out in the National Plan of Action for Drinking Water Conservation (2024). Nine main end uses are included in the model, based on their significance in overall consumption and alignment with policy focus areas. In Figure 3.5 the model component for the nine end uses is shown.

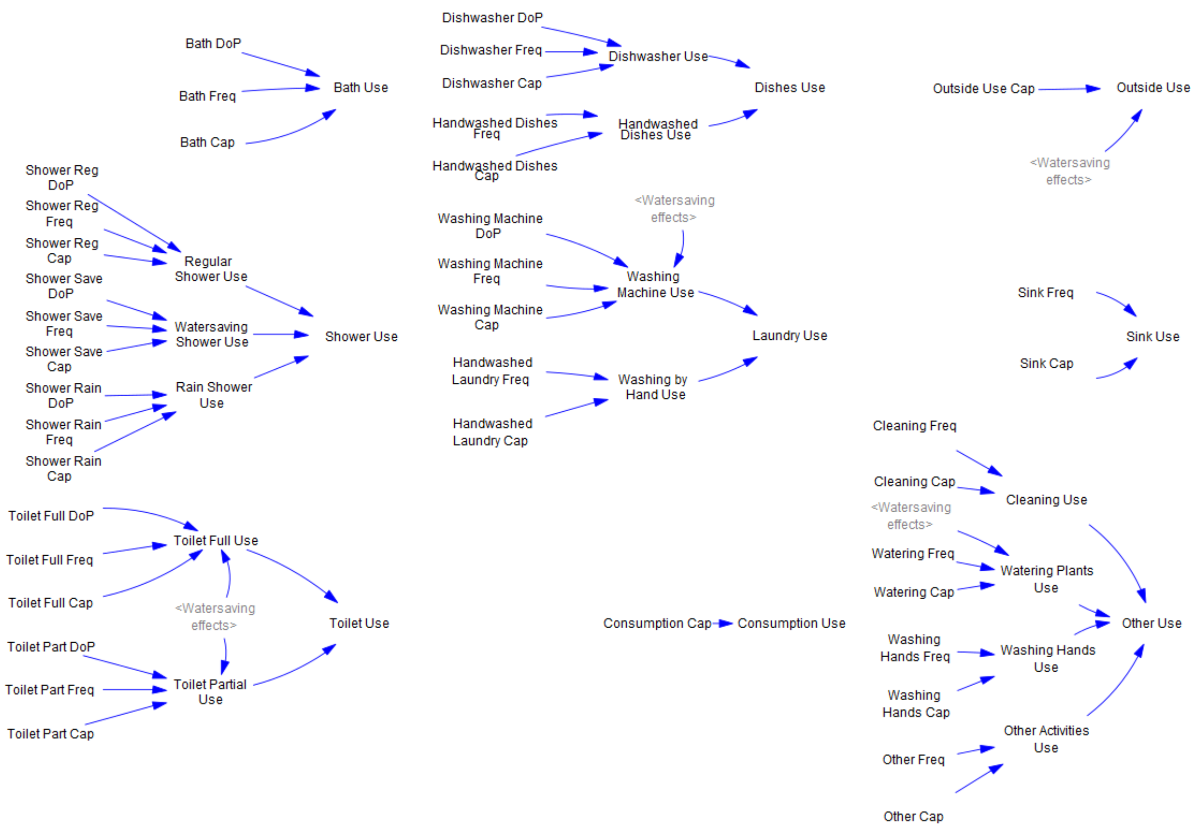


Figure 3.5: The end uses component of the model.

For each of the end uses, the relevant factors are included, since not all are needed for every single one. For example, for the use of drinking water for consumption, it is assumed that the degree of presence is 100%. As a result, only the daily capacity required for that use is included. Additionally, some end use components are more extensive, since they represent a category, like shower use. The modelling decisions with regard to the data used for each of the end use factors are expanded upon further in Table A.5.

Notably, in this section the water-saving effects of the framing campaigns and the comparative invoices take hold on the relevant end uses as well. As the years progress, there are more and more houses equipped with either cascading household water or rainwater harvesting measures, which decreases the Degree of Presence (DoP) of water using devices for specific end uses. Currently, water reuse is only allowed for flushing toilets and watering plants (both in- and outside). Nonetheless, it can be assumed that Dutch legislation will follow the example of Belgium and also include washing machines.

Overall Demand

Afterwards these different demands for their respective end uses are combined into an aggregated demand for household use for drinking water. To be able to pair with the DWDS side of the model this demand is transformed from a daily interval to a yearly one and from litres per person per day to $10^6 m^3$ per year. This aggregation can be seen in Figure 3.6.

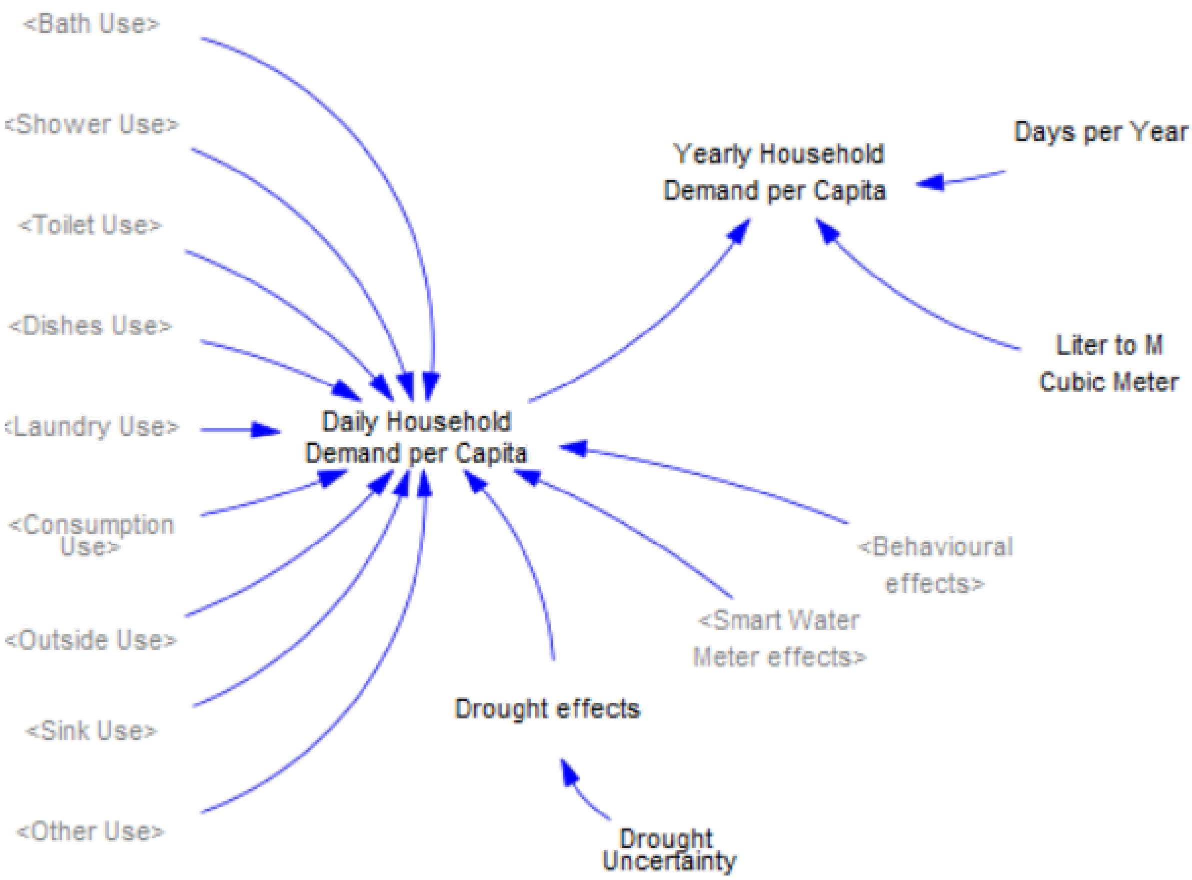


Figure 3.6: The overall demand component of the model.

In this component, the behavioural and SWM effects of their respective policy levers are also introduced. They directly impact daily household demand as they stimulate an overall decrease in the use of drinking water. Similarly, the uncertainty of drought is included in this component as well. Since the drought factor, based on the analysis by Baggelaar et al. (2022) on the drought of 2018, represents an overarching increase in drinking water use for that year, that exogenous driver takes hold here too.

3.2. Overview of Scenarios

The scenarios are based on exogenous demand drivers as introduced in Section 2.3.2. The future development of the drinking water system is quite uncertain; by exploring different possible scenarios, the effectiveness of water conservation measures can be fairly assessed. In Table 3.1 an overview is given of the 12 scenarios that are simulated for each of the water companies.

Table 3.1: Overview of the 12 scenarios selected for the model

#	Population Development	Economical Development	Increased Drought like 2018
A	High	Increased	Yes
B	High	Decreased	Yes
C	High	Increased	No
D	High	Decreased	No
E	Expected	Increased	Yes
F	Expected	Decreased	Yes
G	Expected	Increased	No
H	Expected	Decreased	No
I	Low	Increased	Yes
J	Low	Decreased	Yes
K	Low	Increased	No
L	Low	Decreased	No

3.3. The household demand for drinking water and total household consumption

As can be seen in Figure 3.7 the development of demand per capita appears to be developing differently per company. However, the main and first observation to be drawn from this graph is the inability of the Action Plan to decrease the demand for drinking water per capita to the intended level as was established in the National Plan of Action for Drinking Water Conservation (2024). For all drinking water companies, the drinking water demand per capita appears to hover around 120 litres per person per day by 2035, 20 litres higher than intended.

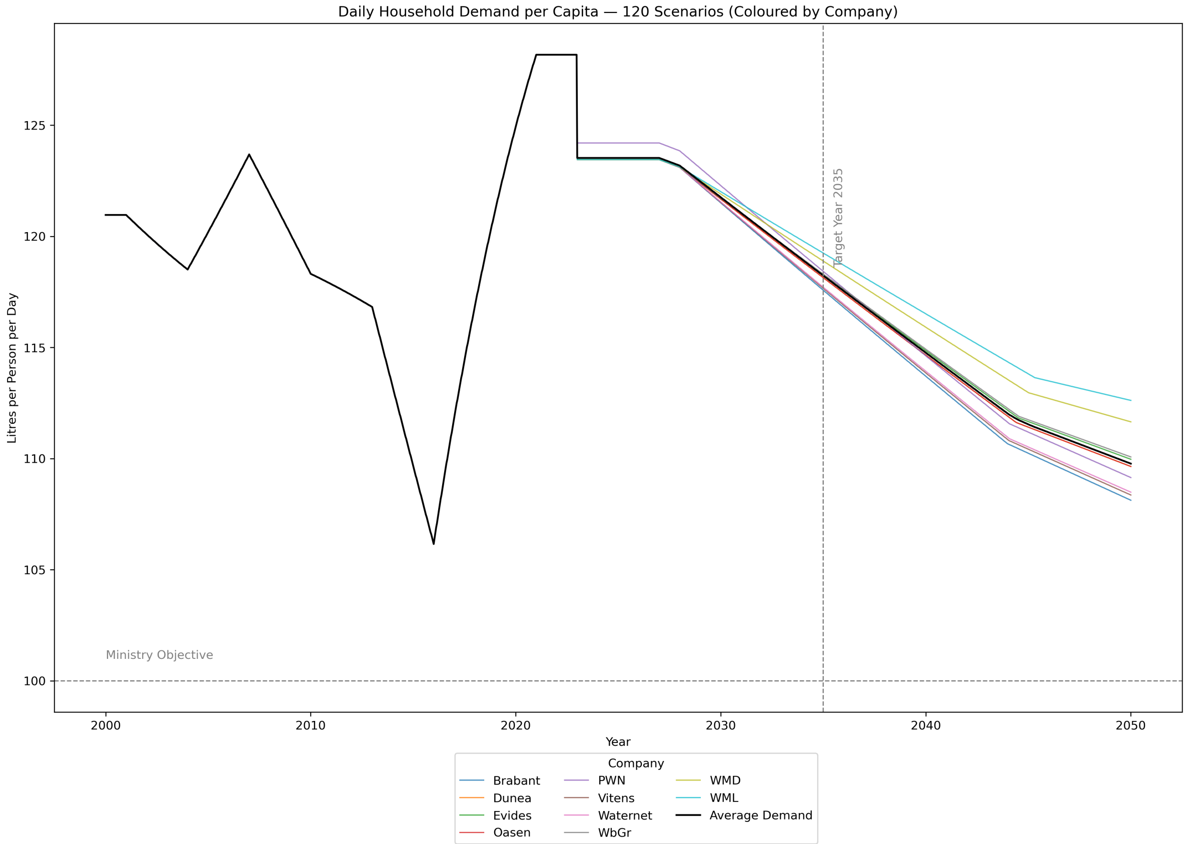


Figure 3.7: The daily household demand per capita - all 120 scenarios, grouped by the company.

A couple other observations can be made from the figure as well:

- There is a clear drop in demand per capita in 2023, when the two behavioural policies have been implemented and start to have an impact. This matches with the flat percentage the policies are modelled as.
- From 2027 onwards, there is a steady gradual decrease once the other policies are introduced as well.
- WMD and WML appear to experience a slower decrease in drinking water demand. In all probability, this can be attributed to a lower construction rate for those regions. Consequently, this will result in fewer newly constructed houses with water-saving features. This explanation coincides with the steepest increase in Brabant, which has the highest average new construction rate.
- After 20 years, the replacement of SWMs (5% per year) is completed and the decrease in demand becomes more gradual as a result.

In Figure 3.8 the two extreme scenarios A & L are compared with the average demand per capita development. This creates a clear image of how the high population development and the increase in drought compare to their direct counter part. The difference in economic uncertainty is of no concern here, as it only affects industrial demand.

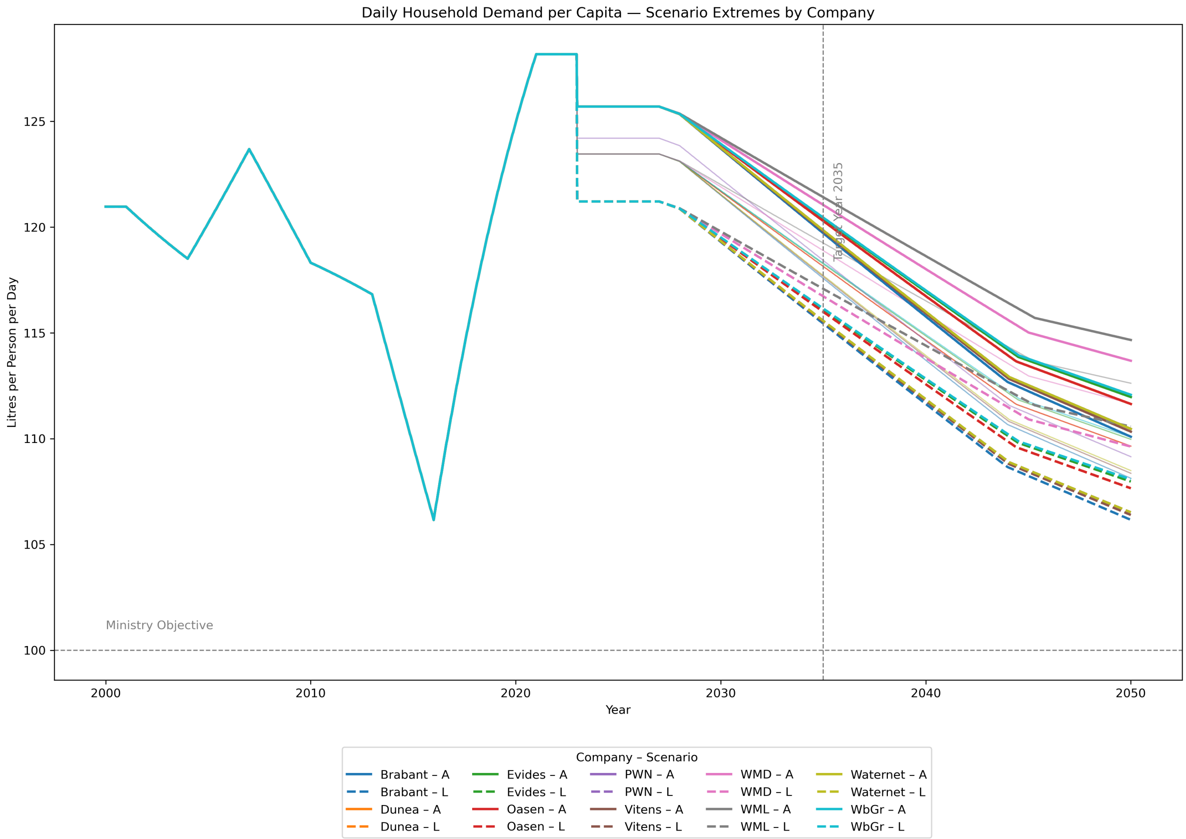


Figure 3.8: The daily household demand per capita - the two extreme scenarios (A & L), grouped by company.

Some additional observations can be made here as well.

- The gradients of the decrease in demand for scenario A seem to be steeper than the gradients in scenario L. As it is by a different amount per company, this cannot be solely attributed to an increase in drought, as that is modelled as a flat percentile impact on the household demand.
- In addition, companies with greater population growth, such as Waternet and Vitens seem to decrease more quickly than companies with a more limited increase in growth such as WML and WMD.

When considering the average total annual household consumption per scenario, as shown in Figure 3.9, for each company, other elements stand out:

- Most companies appear to experience a stabilisation in their total household consumption. The conservation policies, on average, seem to be able to compensate for any uncertain developments.
- However, the more urbanised and rapidly growing regions of Dunea and Waternet still seem to show a growth in total consumption.

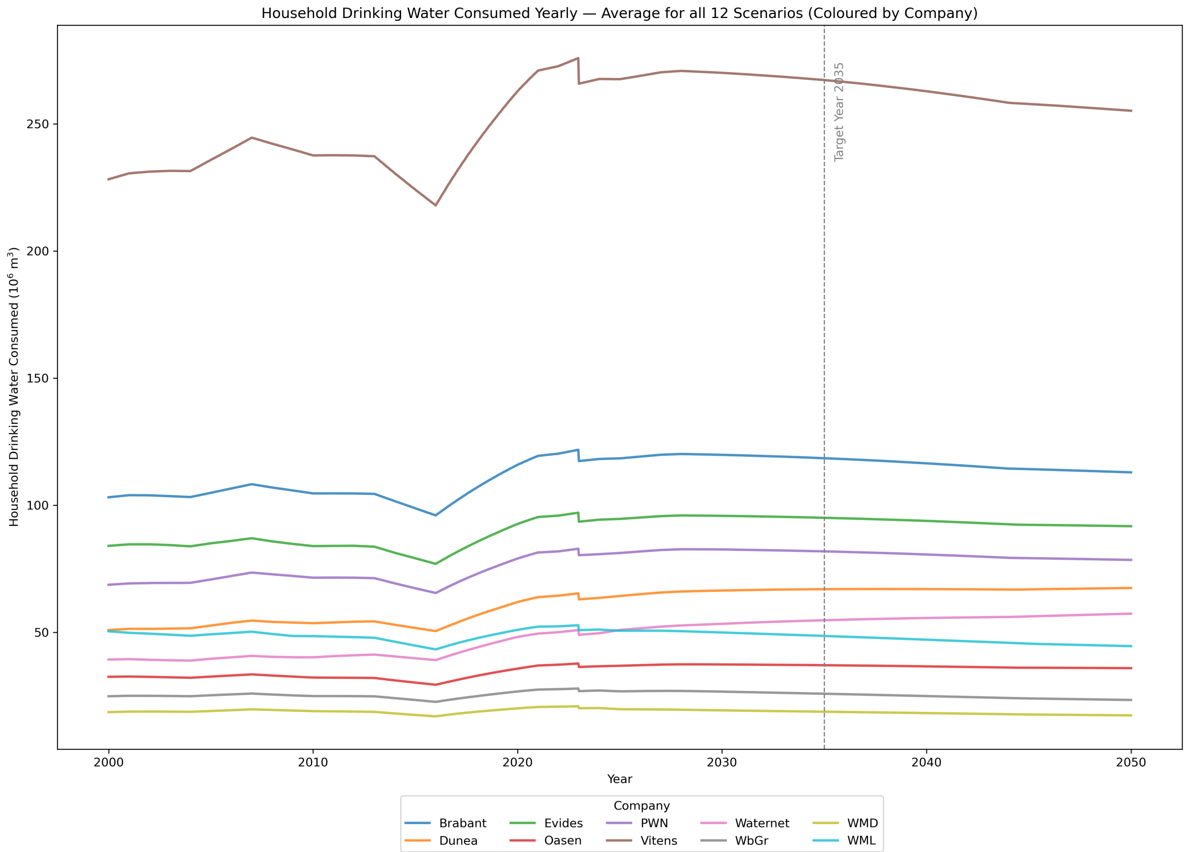


Figure 3.9: The yearly household consumption - average for all 12 scenarios, grouped by the company.

When the two extreme scenarios A & L are taking into consideration once again, as seen in Figure 3.10, the impact of size becomes more pronounced than before. For the smaller regions, there seems to be very little difference in consumption between the two scenarios. Meanwhile, as the total consumption increases from company to company, the differences between the two extreme scenarios become more pronounced. In all likelihood, this is due to the multiplying effect of the population size.

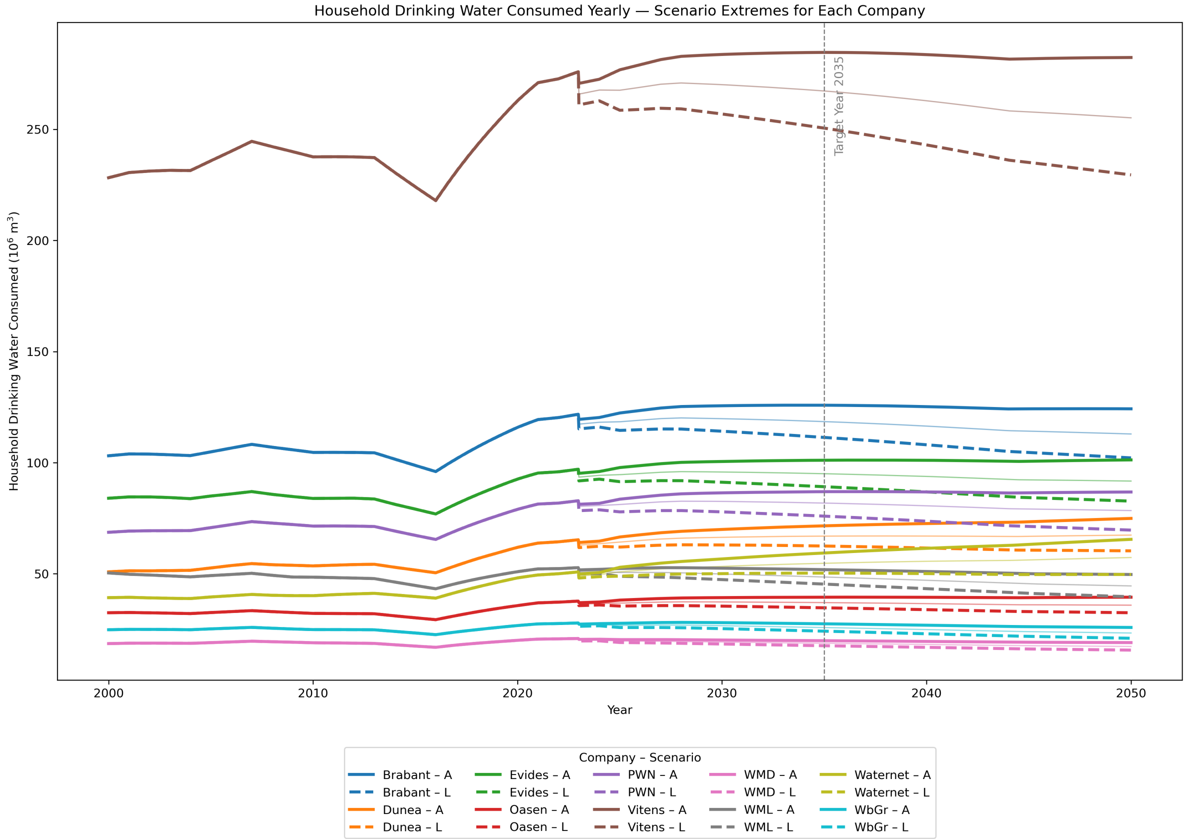


Figure 3.10: The yearly household consumption - the two extreme scenarios (A & L), grouped by company.

3.4. Policy Sensitivity

To investigate the validity of the model output, it is important to see how it responds to a variation in input factors. To this end, a sensitivity analysis is performed to test the assumptions underlying the policy interventions. To achieve this, 2000 simulations are carried out with uniform distributions of 8 different policy parameters, which are outlined in Table 3.2.

Table 3.2: Overview of the 8 policy parameters selected for the sensitivity analysis of the model

Policy parameter	Min Value	Max Value	Source
End Campaign	2022	2050	-
Start Building Regulation Cascading	2027	2050	-
Start Building Regulation Rainwater	2027	2050	-
Start SWM Placement	2027	2050	-
Reduction Factor Campaigns	0	0.0582	(Katz et al., 2018)
Reduction Factor Invoices	0.0098	0.034	(Landon et al., 2018), (Bernedo et al., 2014)
Reduction Factor SWM	0.035	0.089	(Tom et al., 2011), (Tiefenbeck et al., 2018)
Replacement Rate SWM	0	0.1	-

By alternating between different times for the end of the campaigns, the assumption is tested that all campaigns and invoice measures are run indefinitely. Similarly, the starting date for the other three policy interventions is varied up to 2050. The different values for the reduction factors are based on the modifications of the minimum and maximum values of the different policies as determined by Koop et al. (2023).

In Figure 3.11 the results for the 2000 simulations for one of the drinking water companies, in this case Brabant Water, in scenario A are shown. Brabant Water is selected as an example as it is one of the largest drinking water companies, as well as a strong population development. As established in the previous section, demographic development is one of the main driving factors of demand. Therefore, Brabant Water is a strong candidate to highlight the sensitivities of the model design. Scenario A is chosen for a similar reason, because it has the most pronounced effect on the drinking water consumed by households.

These results are compared to a regular run of scenario A and a run of scenario A where no policies are implemented at all.

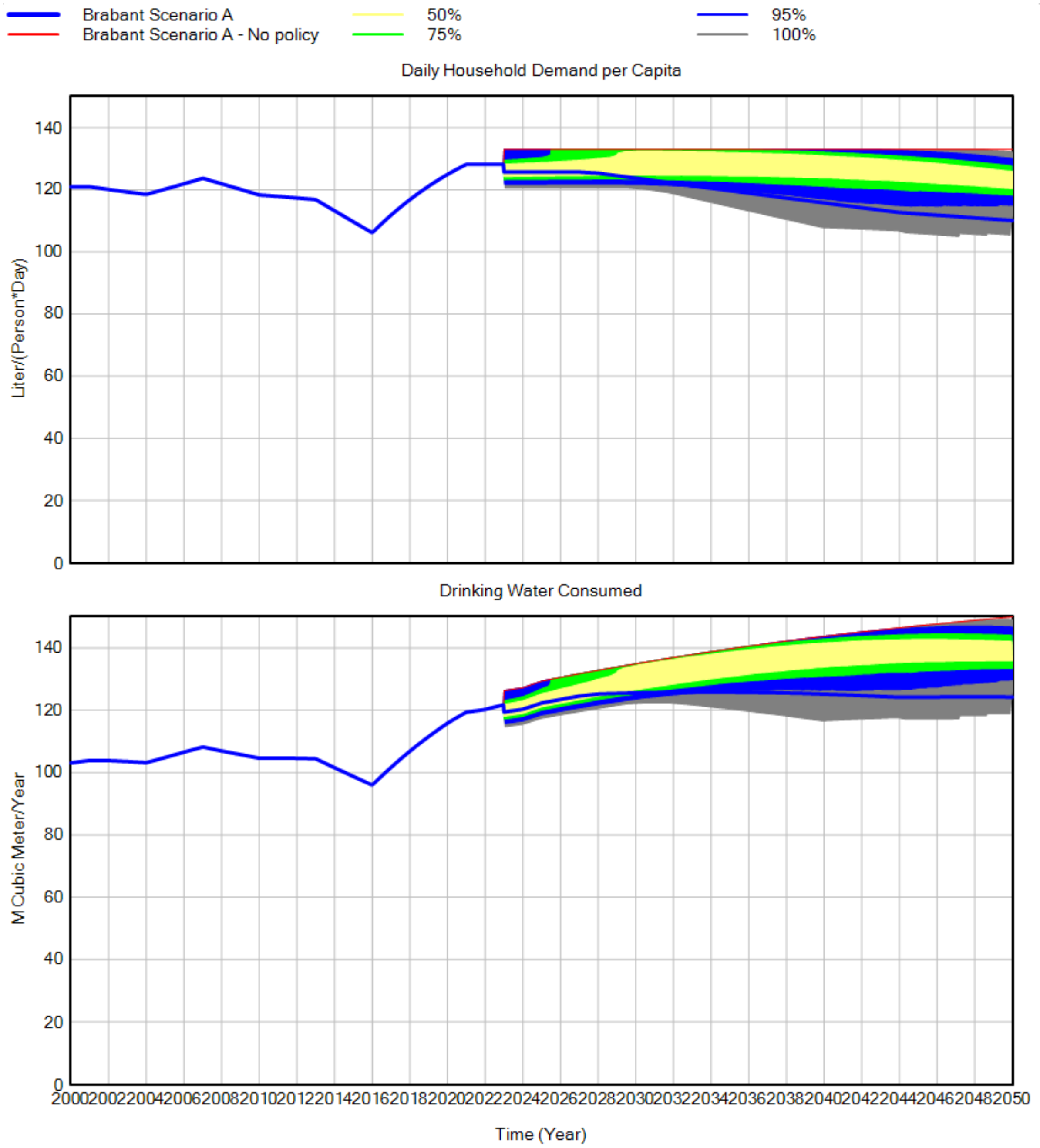


Figure 3.11: The policy sensitivity analysis for Brabant Water, carried out for scenario A.

Several elements stand out in Figure 3.11:

- The regular run of scenario A appears at the lower end of both outcomes. This is unsurprising, since in this run it is assumed that the policies are implemented as quickly as possible, in line with the planning of the national Plan of Action.
- Likewise, the run with no policies implemented shows up on the upper range of the outcomes. It shows that if no action is taken, the consumption of drinking water will continue to increase.
- Both outcomes show a range of $\pm$ 10% for these different simulations.

3.5. The system impact of drinking water consumption

As can be seen in Figure 3.12 on a national level, the consumption of drinking water by households appears to be mainly impacted by population growth, as the scenarios with the greatest population growth seem to lead to the largest increases in consumption. As a benchmark for the national annual total of drinking water consumed by households, the production value of 800 10^6 m^3 was chosen. This value represents the average amount that the study by the van Leerdam et al. (2023) compared with and which is supported by the yearly distribution data (Centraal Bureau voor de Statistiek, 2023).

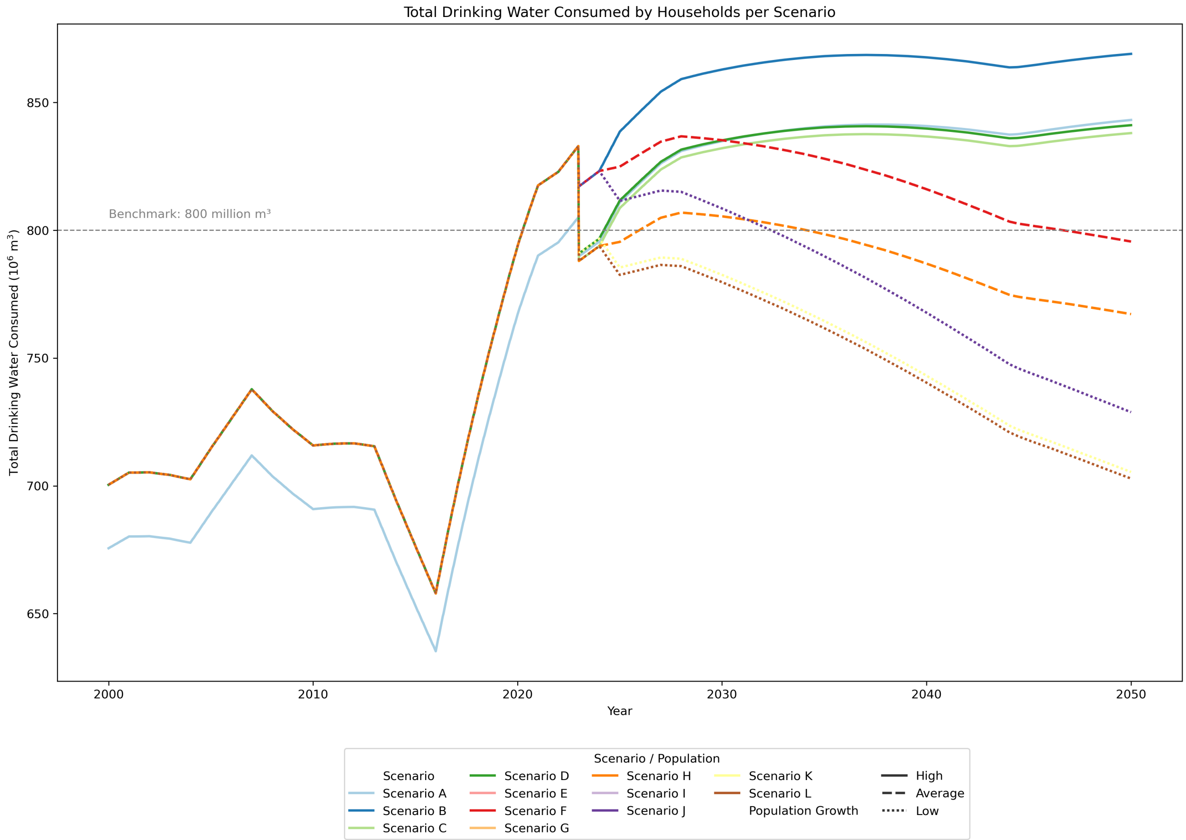


Figure 3.12: The total consumption by households on a national level, grouped by scenario. The lines are full, dashed or dotted depending on the population growth they represent.

- The scenarios with the highest population growth remain above the benchmark. This suggests that in those scenarios additional action should be taken on top of the modelled policies, as the probability for shortages increases drastically in these scenarios.
- The scenarios without additional drought periods appear to score consistently lower than their counterparts.
- The impact of aggregated population data appears to be felt the strongest in 2024. The split between the three paths is quite rough.

Looking at Figure 3.13, the potential influence of economic growth becomes apparent as well once industrial consumption is taken into account. In this instance, the benchmark of 1200 10^6 m^3 was chosen, as that represents the extraction limit that the study by the van Leerdam et al. (2023) based its conclusions on with regard to the potential shortages by 2030.

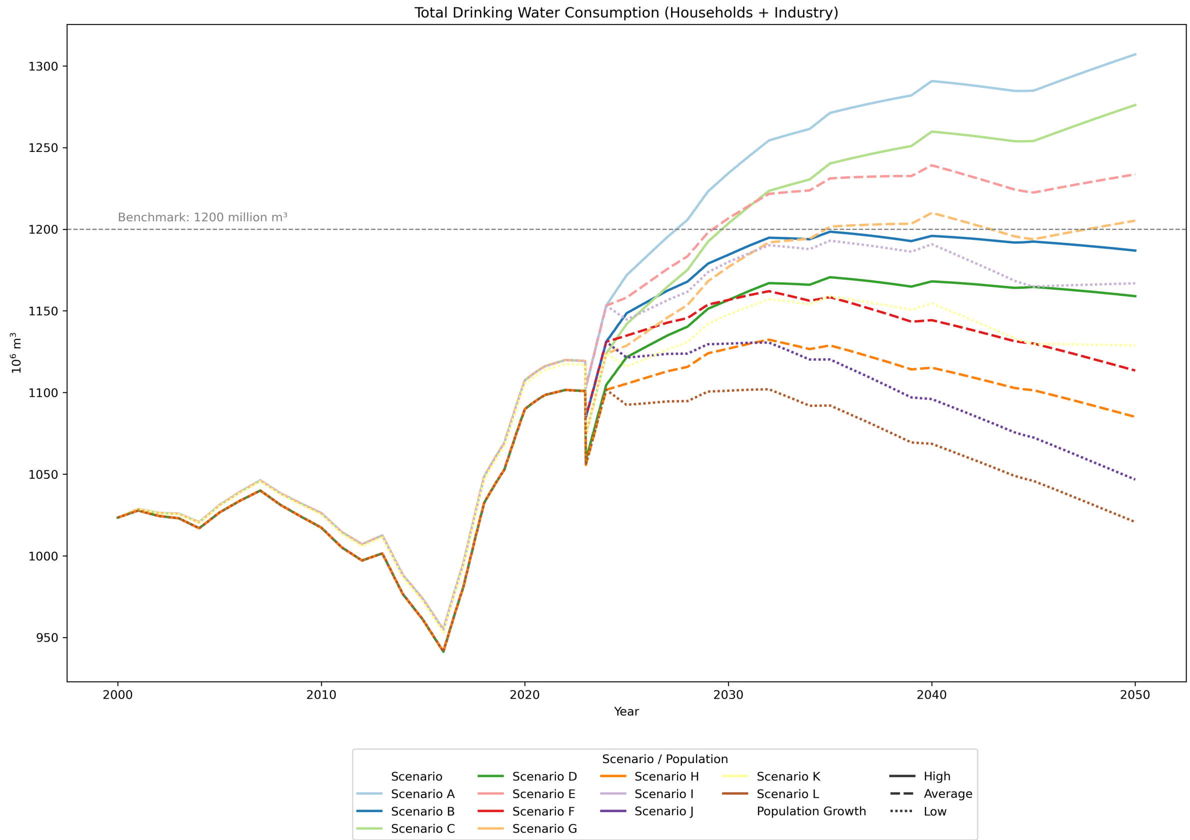


Figure 3.13: The total consumption by households and industry on a national level, grouped by scenario.

- There are four scenarios that remain above the benchmark, all of which have an increased economic growth factor. This shows that, in order to create a more robust DWDS, demand management policies are required to limit industrial demand.
- Meanwhile, if population growth is paired with economic contraction, then the drop in industrial demand seems to be able to compensate for the increased household demand, as shown in scenarios B and D.
- In times of limited population growth, the policy package simulated in this thesis appears to be effective in each of the four scenarios.

However, on average, water companies are unlikely to experience shortages in the coming decades if all the plans outlined for additional extraction sources come to fruition (Vewin, 2025), as can be seen in Figure 3.14. However, strategic reserves are still more likely to be challenged by increased population and economic growth. It is unlikely that each region of the Netherlands will experience these dynamics at the same time. In all likelihood, if these dynamics differ from area to area, companies could support each other in times of shortages.

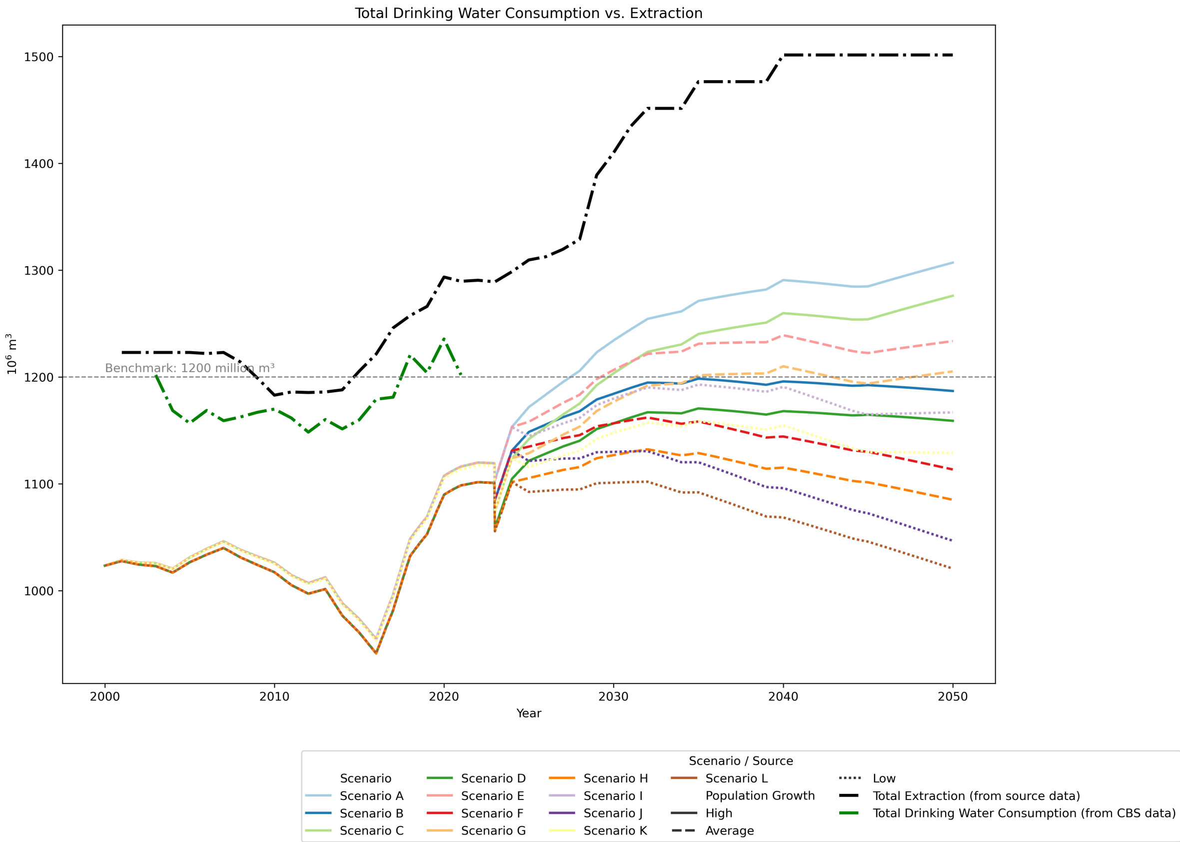


Figure 3.14: The total consumption by households and industry on a national level, grouped by scenario. Compared to total extraction, in black, and the historical consumption data, in green.

An important final observation to make is that the first half of the simulation does not match with the historic data for national drinking water consumption (Centraal Bureau voor de Statistiek, 2023). In Vewin (2017b) the authors already identified the challenge of bias, as it was becoming increasingly clear that residential consumers were consistently under-reporting their use of drinking water.

4 Discussion

This chapter provides context for the results presented in the previous chapter by examining their implications in relation to the broader research objective.

It addresses the 5th sub-question: *In what scenarios will the Ministry of Infrastructure meet its objectives and what challenges might arise?*

Furthermore, the chapter reflects on the limitations of this study and proposes recommendations for future researchers and policy makers.

4.1. Simulation results in a broader context

4.1.1. Plan of Action development

Considering that the National Plan of Action for Drinking Water Conservation (2024) has only been published recently, there is still a fair amount of uncertainty about its implementation. Quite a few of the proposed measures still require further investigation and have not yet been specified. The discrepancy between these conceptual measures and the quantitative needs of the model is likely to have skewed the results, as they were excluded from the simulation or misinterpreted when they were included. Some of the excluded measures, such as pricing mechanisms, crisis-time restrictions, or subsidies for water-saving technologies, could have had a significant impact on household demand if implemented.

Moreover, with this lack of specificity and operational clarity at the time of research, it does give an accurate representation of the current development of drinking water demand management in the Netherlands at this point in time. The absence of industrial demand management and several household measures in the model should not be interpreted as ineffectiveness, but rather as a reflection of insufficient data to simulate their impacts credibly. In addition, it remains unclear what the current political support is for the proposed measures, since it was published a week before a transition from one coalition to the next. As a result, some of these measures might not be implemented after all. Next to political support, this thesis does not explicitly evaluate the social feasibility of the proposed measures. Some of the more impactful levers, such as mandatory SWM replacement or household water reuse systems are more likely to meet resistance. Future iterations of this research could incorporate more levers as pilots finish and more policy evaluations become available.

As of now, the Netherlands is still in the middle of the European pack with regards to average household use, as can be seen in Figure 4.1. This matches with the historic image of a well managed DWDS designed around accessibility and affordability (Van den Berg et al., 2022). Other Western European countries such as Belgium and Denmark have started to manage their demand. Whether the Netherlands will join them in the nearby future, will depend on the willingness of Dutch policy makers to take decisive action.

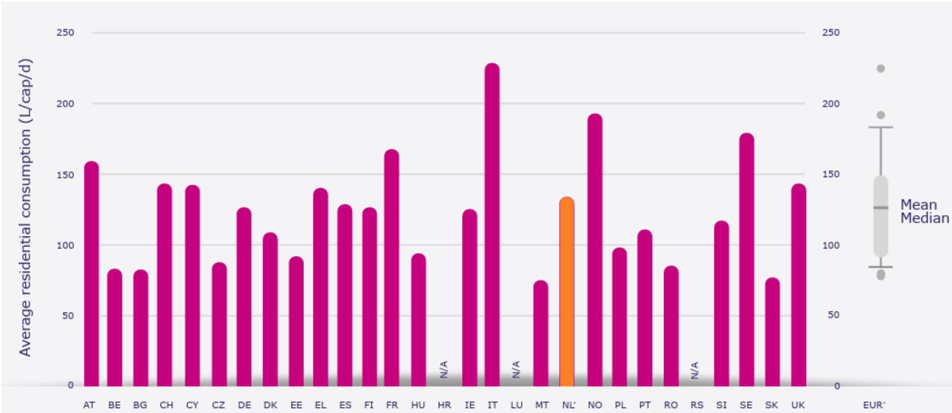


Figure 4.1: Average drinking water use in the European Union, as adopted from EurEau (2021). With the Netherlands in orange.

4.1.2. Meeting objectives and facing challenges

With the results of the simulations and the current development of the Plan of Action in mind, it is possible to provide an answer to the 5th sub-question.

While this thesis focused on households, the National Plan of Action for Drinking Water Conservation (2024) still holds a lot of potential for managing industrial demand. Through ensuring that the right type of water is employed for the right use, a lot can still be achieved in the industrial sector. Provided that the intended fieldwork through quick scans offers sufficient possibility for demand management. If the pressure of industrial demand would be decreased along with the household demand, a lot of pressure would be relieved from the Dutch DWDS.

This research showed the potential of moving quickly on implementation. In 10 years time a decrease of 10 lpppd could be achieved across the board. If more policies were to be implemented this decrease could even be more steep. Policy makers would be advised to act decisively, as each year of waiting will bring the policy target further out of reach.

Aside from time, the thesis also provided insights into two further barriers: population and economic growth. The DWDSs of rapidly urbanising regions are challenged by socio-demographic developments, which they are only able to meet if all actions lined out in this paper are taken.

After all, it was assumed that all the plans formulated in the Vewin (2025) were to be successfully implemented. However, the old approach of pushing the supply side of the Dutch DWDS still seems to be viable for now. That is, if all the proposed plans for new freshwater sources do not experience extensive delays or costs. Additionally, the limits of existing sources seem to be of little concern to the drinking water sector, as they remain focused on expansion to tackle this problem. To be truly successful in meeting the objectives, extensive demand management is needed.

4.2. Limitations

4.2.1. Method limitations

A key limitation lies in the model's assumption of the fixed reduction effect of the implemented policies. In reality, behavioural adaptation is likely to exhibit non-linear and time-varying characteristics.

In addition, the exogenous drivers and policy levers implemented in the model build on earlier research and rely on the modeller's choices. By choosing what to add and what to exclude the results are impacted directly. By designing the model structure around certain earlier findings other elements might be overlooked and thereby current insights could be overstated.

4.2.2. Data limitations

The data limitations for the supply components of the SD model are primarily based on the fragmented nature of the data and the way it is presented. Aside from the population data, nearly all other model inputs had to be drawn from tables in reports. As a result, this introduces large rounding errors, since a decimal can be quite impactful on the scale of the data used. Similarly, from year to year these reports employ shifting definitions for their contents, as the needs of the water companies shift over time. Furthermore, there is very little quantitative information available on both the impact of seasonality and extreme weather events on existing freshwater sources. For now, drinking water companies seem to be mainly focused on expanding their extraction permits, instead of investigating the limits of the ecosystems in which those sites reside. Therefore, for the future it remains unclear whether there will be enough days in the year to extract water from fresh water sources, as its quality might be below legal standards at that point in time due to high temperatures or salinity levels.

Data limitations for the demand components of the research are mostly on the end-use level. Especially since the surveys used have a complete reliance on self-reported information, which might introduce bias, as consumers might over- or underestimate their drinking water use. The surveyors already concluded this in the survey of 2016, when a simulation check provided vastly different results (Vewin, 2017b). Future research can hopefully build on the implementation of SMWs, as it is already proving to be quite promising (Cominola et al., 2015). Furthermore, these surveys also lack regional diversity and sample sizes. Currently, the model relies on a group of several thousand respondents to model the end-use of drinking water. This makes it very difficult to distinguish properly between the different distribution areas. Consequently, the same data set was used for each of the ten drinking water companies. This makes it challenging to determine whether certain policy directions are more worthwhile for different areas.

4.3. Research recommendations

Building on the findings and limitations discussed, several avenues for further investigation could be of interest to increase understanding of and support policymaking with regards to drinking water conservation.

4.3.1. Implementation of additional policies

Future studies could incorporate additional demand management levers, such as targeted subsidies for water-saving appliances, tax incentives, or tiered pricing structures. For now, these measures are still too unspecified to be included or lacking political support in the case of financial mechanisms.

4.3.2. Modelling the limits of existing sources

The current model follows the assumption made by the drinking water sector that the current availability of freshwater sources will not change. Future research should investigate the gradual decline or degradation of the quality of water sources under the conditions of increased periods of drought, salinisation, or over-extraction by different sectors. Incorporating this vulnerability would better reflect the strain on the system and ecological thresholds.

4.3.3. Trade-off's between behavioural change and hydraulic paradigm

It would be valuable to explore the trade-off's that could be made between encouraging behavioural change and the current path of expanding the extraction capacity of the water companies. Future research could assess the long-term sustainability and public acceptance of each strategy under different future scenarios.

There is a need to link water conservation policies with broader ecosystem constraints. Over time, population and infrastructure expansion may exceed local ecological carrying capacities. Integrated models that incorporate indicators for biodiversity and ecosystem health would help balance human and environmental water needs.

5 Conclusion

This thesis set out to investigate to what extent, and under what conditions, the Netherlands is likely to meet the drinking water conservation targets outlined in the National Plan of Action for Drinking Water Conservation (2024). In light of anticipated water shortages and thus increasing pressure on the Dutch drinking water distribution system (DWDS), the research aimed to assess the effectiveness of proposed household-level demand conservation measures.

The main research question guiding this thesis was: *To what extent and under what conditions are the objectives of the Ministry of Infrastructure and Water Management for drinking water conservation more likely to be met?*

A key contribution of this work is the creation of an integrated simulation model that combines a bottom-up household demand component with a system-level supply component. The model is designed using Exploratory System Dynamics Modelling and Analysis (ESDMA), which when combined provide an instrument to study policy effectiveness under conditions of deep uncertainty. It was applied across twelve scenarios for each of the ten Dutch drinking water companies, allowing for regional variation and future uncertainties related to population growth, drought circumstances and economic development.

The results show that the current policy measures are insufficient to reach the 2035 target of 100 litres of drinking water used per person per day. Instead, average demand remains around 120 litres by 2035 and declines only up to 110 litres by 2050. While the policies, such as framing campaigns, comparative invoices and smart meter deployment, do reduce demand, their combined impact under all tested conditions falls short of the national target.

The model further shows that total household and industrial demand is likely to increase up to 2030, particularly in scenarios with high population and economic growth. While national supply limits may not be exceeded, regional variation indicates that some companies may encounter local shortages. This highlights the importance of taking a regionalised approach to water management and policy implementation.

In this regard, the model provides a valuable tool for water companies and policymakers alike. Its structure allows for reuse and adaptation in future research or strategic planning efforts. In evaluating the broader impact of the proposed conservation strategy, the results show that demand management can be of importance in strengthening the robustness of the Dutch DWDS. However, to meet the targets and mitigate regional supply risks, additional measures are necessary.

Lastly, the model's exploratory nature emphasises the value of scenario-based approaches in public infrastructure planning. As future developments in climate, population, and behaviour are inherently uncertain, it is essential to utilise modelling tools that are adaptive and flexible. Future research could expand on this work by including industrial demand management policies and additional dynamics for fresh water sources.

This thesis demonstrates how combining a highly detailed, behavioural perspective with a systems approach can enhance the understanding of demand management policies. It provides a replicable method for testing the robustness of conservation strategies under uncertainty, providing insights that are of practical relevance to both policy design and implementation.

References

Arnold, R. D., & Wade, J. P. (2015). A definition of systems thinking: A systems approach. *Procedia Computer Science*, *44*, 669–678. https://doi.org/10.1016/j.procs.2015.03.050

Auping, W. L. (2018). *Modelling uncertainty: Developing and using simulation models for exploring the consequences of deep uncertainty in complex problems* [Doctoral dissertation, Delft University of Technology]. https://doi.org/10.4233/uuid:0e0da51a-e2c9-4aa0-80cc-d930b685fc53

Baggelaar, P., Kuin, P., & Geudens, P. (2022). *Prognoses drinkwatergebruik in nederland t/m 2040*. Vewin.

Bakker, J., van der Mooren, F., & Boonstra, H. J. (2022). Watergebruik thuis 2021. https://www.vewin.nl/wp-content/uploads/2024/06/vewin-watergebruik-thuis-2021.pdf

Baltus, R., Sanders, R., Schreuders, R., Dröge, J., & Posma, J. (2022). Bewust en zuinig drinkwatergebruik: Verkenning effectief instrumentarium. https://www.rijksoverheid.nl/documenten/rapporten/2023/02/09/bijlage-1-verkenning-bewust-en-zuinig-drinkwatergebruik

Bankes, S. (2010). Robustness, adaptivity, and resiliency analysis [AAAI Fall Symposium Technical Report]. *Proceedings of the AAAI Fall Symposium Series*.

Beleidsnota Drinkwater 2021-2026, Ministry of Infrastructure & Water Management (2021). https://open.overheid.nl/documenten/ronl-ae2317be-417a-409f-bd2d-59ec08a55620/pdf

Beleidstafel Droogte. (2019, December). *Nederland beter weerbaar tegen droogte: Eindrapportage beleidstafel droogte*. Ministerie van Infrastructuur en Waterstaat. https://open.overheid.nl/documenten/ronl-749b44a3-8e4b-427b-ad23-9be64203a619/pdf

Bernedo, M., Ferraro, P. J., & Price, M. (2014). The persistent impacts of norm-based messaging and their implications for water conservation. *Journal of Consumer Policy*, *37*(3), 437–452. https://doi.org/10.1007/s10603-014-9266-0

Blankesteijn, M. L., & Pot, W. D. (2024). Water governance in the netherlands. In *Oxford research encyclopedias*. Oxford University Press. https://doi.org/10.1093/acrefore/9780199389414.013.863

Brabant Water & Province of Noord-Brabant. (2024, October). *Regionaal actieplan beschikbaarheid drinkwaterbronnen tot 2030 - brabant*. Brabant Water and Province of Noord-Brabant. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-20-regionaal-actieplan-beschikbaarheid-drinkwaterbronnen-tot-2030-noord-brabant-brabant-water

Centraal Bureau voor de Statistiek. (2019). *Bevolking, huishoudens en bevolkingsontwikkeling; 1899-2019* [Accessed: April 6, 2025]. https://opendata.cbs.nl/#/CBS/nl/dataset/37556/table?dl=34F0C

Centraal Bureau voor de Statistiek. (2023). Watergebruik binnen de nederlandse economie; milieurekeningen [Dataset retrieved May 16, 2025]. https://opendata.cbs.nl/statline/#/CBS/nl/dataset/82883NED/table?dl=C0C1B

Centraal Bureau voor de Statistiek. (2024). *Bevolking; geslacht, leeftijd en burgerlijke staat, 1 januari — gemeenten* [Municipality-level population by gender, age, and marital status on January 1st]. https://opendata.cbs.nl/#/CBS/nl/dataset/03759ned/table

Centraal Bureau voor de Statistiek. (2025). *Voorraad woningen en niet-woningen; mutaties, gebruiksfunctie, regio* [Accessed: April 6, 2025]. https://opendata.cbs.nl/#/CBS/nl/dataset/81955NED/table

Centraal Bureau voor de Statistiek (CBS). (2023). Bevolking; kerncijfers wijken en buurten, 2023 [Accessed 7 June 2024]. https://opendata.cbs.nl/#/CBS/nl/dataset/86030NED/table

Coalitieakkoord 2021 – 2025: VVD, D66, CDA, ChristenUnie (2021, December 15). https://open.overheid.nl/documenten/ronl-f3cb0d9c-878b-4608-9f6a-8a2f6e24a410/pdf

Cominola, A., Guliani, M., Piga, D., Castelletii, A., & Rizzoli, A. (2015). Benefits and challenges of using smart meters for advancing residential water demand modeling and management: A review. *Environmental Modelling Software*, *72*, 198–214. https://doi.org/10.1016/j.envsoft.2015.07.012

CPB & PBL. (2015). Toekomstverkenning welvaart en leefomgeving: Scenario's voor 2016-2045 [Accessed 2024-06-07]. https://www.wlo2015.nl/

Drinkwaterbesluit (2011, July 1). https://wetten.overheid.nl/BWBR0030111/2023-07-01

Drinkwaterwet (2015, July 1). https://wetten.overheid.nl/BWBR0026338/2015-07-01

Dumaij, A., & Heezik, A. (2012). Productiviteitstrends in de drinkwatersector. een empirisch onderzoek naar het effect van regulering op de productiviteitsontwikkeling tussen 1985 en 2010.

EurEau. (2021). Europe's water in figures: An overview of the european drinking water and waste water sectors [Accessed: 2025-05-12]. https://www.eureau.org/resources/publications/eureau-publications/5824-europe-s-water-in-figures-2021/file

Directive (EU) 2020/2184 of the European Parliament and of the Council of 16 December 2020 on the quality of water intended for human consumption (2020, December 16). https://eur-lex.europa.eu/legal-content/EN/TXT/?uri=CELEX:32020L2184

Fuenfschilling, L., & Truffer, B. (2016). The interplay of institutions, actors and technologies in socio-technical systems – an analysis of transformations in the australian urban water sector. *Technological Forecasting and Social Change*, *103*, 298–312. https://doi.org/10.1016/j.techfore.2015.11.023

Hermans, P., Phernambucq, I., Klein, J., & Remijn, E. (2023, June). *Hemelwater- en grijswatergebruik in het gebouw. mogelijke verplichting in het bbl*. Ministerie van Binnenlandse Zaken en Koninkrijksrelaties. https://open.overheid.nl/documenten/fdd63490-0688-4caf-b9e1-d9c83d29c68a/file

Kanne, P. (2005, January). *Watergebruik thuis 2004* (Uitgevoerd in opdracht van de Vereniging van Waterbedrijven in Nederland (Vewin)). TNS NIPO. https://edepot.wur.nl/106399

Katz, D., Kronrod, A., Grinstein, A., & Nisan, U. (2018). Still waters run deep: Comparing assertive and suggestive language in water conservation campaigns. *Water*, *10*(3), 275. https://doi.org/10.3390/w10030275

Koop, S. H. A., Van Dorssen, A. J., & Brouwer, S. (2019). Enhancing domestic water conservation behaviour: A review of empirical studies on influencing tactics. *Journal of Environmental Management*, *247*, 867–876. https://doi.org/10.1016/j.jenvman.2019.06.126

Koop, S., Brouwer, S., & Zeidan, M. (2023). *100 liter per persoon per dag. welke waterbesparingsmaatregelen zijn nodig?* KWR Water Research Institute. https://library.kwrwater.nl/publication/70932833/100-liter-per-persoon-per-dag-welke-waterbesparingsmaatregelen-zijn-nodig/

Kwakkel, J. H., Auping, W. L., & Pruyt, E. (2013). Dynamic scenario discovery under deep uncertainty: The future of copper. *Technological Forecasting and Social Change*, *80*(3), 789–800. https://doi.org/10.1016/j.techfore.2012.10.005

Kwakkel, J. H., & Pruyt, E. (2015). Using System Dynamics for Grand Challenges: The ESDMA Approach. *Systems Research and Behavioral Science*, *32*, 358–375. https://doi.org/10.1002/sres.225

Landon, A. C., Kyle, G. T., & Kaiser, R. A. (2018). Evaluating the efficacy of an information-based residential outdoor water conservation program. *Journal of Cleaner Production*, *195*, 1248–1257. https://doi.org/10.1016/j.jclepro.2018.05.196

Lempert, R. J., Popper, S. W., & Bankes, S. C. (2003). *Shaping the next one hundred years: New methods for quantitative, long-term policy analysis*. RAND Corporation. https://www.rand.org/pubs/monograph_reports/MR1626.html

Mauro, A. D., Cominola, A., Castelletti, A., & Nardo, A. D. (2021). Urban water consumption at multiple spatial and temporal scales. a review of existing datasets. *Water*, *13*. https://doi.org/10.3390/w13010036

Mazzoni, F., Alvisi, S., Blokker, M., Buchberger, S. G., Castelletti, A., Cominola, A., Gross, M.-P., Jacobs, H. E., Mayer, P., Steffelbauer, D. B., Stewart, R. A., Stillwell, A. S., Tzatchkov, V., Yamanaka, V.-H. A., & Franchini, M. (2023). Investigating the characteristics of residential end uses of water: A worldwide review. *Water Research*, *230*. https://doi.org/10.1016/j.watres.2022.119500

National Plan of Action for Drinking Water Conservation, Ministry of Infrastructure & Water Management (2024). https://open.overheid.nl/documenten/dpc-0654c0aee2fc2ada28a98f250b992ea7060863f8/pdf

Nota Water en Bodem Sturend, Ministry of Infrastructure & Water Management (2023). https://open.overheid.nl/documenten/ronl-c35e65eba0903d738ae26dab222462337b0d8de7/pdf

Oasen & Provinces of Zuid-Holland and Utrecht. (2024). *Regionaal actieplan beschikbaarheid drinkwaterbronnen - oasen*. Oasen, Provinces of Zuid-Holland, and Utrecht. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-9-regionaal-actieplan-utrecht-oasen

PBL & CBS. (2022). *Regionale bevolkings- en huishoudensprognose 2022 — gemeenten* [Population and household forecasts at the municipal level in the Netherlands]. https://www.pbl.nl/publicaties/regionale-bevolkings-en-huishoudensprognose-2022

Phan, T. D., Bertone, E., & Stewart, R. A. (2021). Critical review of system dynamics modelling applications for water resources planning and management. *Cleaner Environmental Systems*, *2*, 100031. https://doi.org/10.1016/j.cesys.2021.100031

Province of Drenthe. (2023). *Regionaal uitvoeringsprogramma drinkwaterbronnen 2023-2030*. Province of Drenthe. https://www.provincie.drenthe.nl/publish/pages/143405/w2502_020-dr-regionaal_uitvoeringsprogramma_drinkwaterbronnen_2023_2030.pdf

Province of Limburg. (2023). *Limburgs actieprogramma beschikbaarheid drinkwaterbronnen 2023-2030*. Province of Limburg. https://www.limburg.nl/publish/pages/9096/wml_limburgs_actieprogramma_beschikbaarheid_drinkwaterbronnen_2023-2030_def_1_.pdf

Province of Zuid-Holland. (2024). *Actieplan beschikbaarheid drinkwaterbronnen 2030 - dunea*. Province of Zuid-Holland. https://www.zuid-holland.nl/publish/besluitenattachments/actieplannen-beschikbaarheid-drinkwaterbronnen-2030-zuid-holland/actieplan-beschikbaarheid-drinkwaterbronnen-2030-dunea.pdf

PWN Waterleidingbedrijf Noord-Holland N.V. (2024). *Regionaal actieplan beschikbaarheid drinkwaterbronnen 2023 tot 2030 - pwn*. PWN Waterleidingbedrijf Noord-Holland N.V. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-18-regionaal-actieplan-beschikbaarheid-drinkwaterbronnen-2023-2030-noord-holland-pwn

Royal Netherlands Meteorological Institute. (2024). Knmi climate scenarios data. https://klimaatscenarios-data.knmi.nl/

Sterman, J. D. (2000). *Business dynamics: Systems thinking and modeling for a complex world*. Irwin/McGraw-Hill.

Tiefenbeck, V., Goette, L., Degen, K., Tasic, V., Fleisch, E., Lalive, R., & Staake, T. (2018). Overcoming salience bias: How real-time feedback fosters resource conservation. *Management Science*, *64*(3), 1458–1476. https://doi.org/10.1287/mnsc.2016.2646

TNS NIPO. (2013). *Synopsis: Watergebruik thuis 2013* (Onderzoek in opdracht van Vewin). TNS NIPO. https://sciencetalk.nl/forum/download/file.php?id=16886

Tom, G., Tauchus, G., Williams, J., & Tong, S. (2011). The role of communicative feedback in successful water conservation programs. *Applied Environmental Education & Communication*, *10*(2), 80–90. https://doi.org/10.1080/1533015X.2011.575632

van Leerdam, R., Rook, J., Riemer, L., & van der Aa, N. (2023). *Water availability for drinking water production up to 2030: Difficulties and solutions* (Briefrapport No. 2023-0005). Rijksinstituut voor Volksgezondheid en Milieu. Bilthoven, The Netherlands. https://doi.org/10.21945/RIVM-2023-0005

Van Alphen, H., De Boer, J., Van der Velde, R., & Van der Wal, B. (2022, July). *Denkkaders voor de drinkwaterketen* (Published in H2O-Online). H2O Waternetwerk. https://www.h2owaternetwerk.nl/images/2022/Juli/H2O-Online_220725_Denkkaders_drinkwaterketen.pdf

Van den Berg, R., Van der Roest, E., & Van Aalderen, N. (2022, July). Denkkaders in de drinkwaterketen: Terugblik op 200 jaar systeemontwikkeling [Accessed: May 17, 2025]. https://www.h2owaternetwerk.nl/images/2022/Juli/H2O-Online_220725_Denkkaders_drinkwaterketen.pdf

Van der Linden, E. (2020). *Exploration of the cobalt system: Scenarios for a critical material for the energy system* [Master's thesis]. TU Delft. https://repository.tudelft.nl/record/uuid:e51dbb87-09f7-4c33-a956-226874a1e7b7

Ventana Systems, Inc. (2025). *Vensim pro academic* (Version 10.3.2) [Licensed academic version]. https://vensim.com/vensim-pro/

Vewin. (2005, January). *Water in zicht 2003* (Bijlage No. vrom050135) (Vergaderjaar 2004–2005, officiële publicatie). Tweede Kamer der Staten-Generaal.

Vewin. (2007). *Waterleidingstatistiek 2007*. https://adoc.pub/waterleidingstatistiek.html

Vewin. (2008). *Drinkwaterstatistieken 2008*. Vewin. https://web.archive.org/web/20221005214613/http://www.vewin.nl/SiteCollectionDocuments/Publicaties/Vewin_Drinkwaterstatistieken2008_lr.pdf

Vewin. (2009). *Kerngegevens drinkwater 2010*. Vewin. https://web.archive.org/web/20240712070221/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Kerngegevens%20drinkwater%202009.pdf

Vewin. (2011, January). *Watergebruik thuis 2010*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/08/Vewin_Watergebruik_thuis_2010.pdf

Vewin. (2012a). *Drinking water fact sheet 2012*. Vewin. https://web.archive.org/web/20220401033104/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/722.162%20Drinking%20water%20fact%20sheet%202012%20(1).pdf

Vewin. (2012b). *Drinkwaterstatistieken 2012*. Vewin. https://www.yumpu.com/nl/document/read/20029010/drinkwaterstatistieken-2012-vewin

Vewin. (2013). *Kerngegevens drinkwater 2013*. Vewin. https://web.archive.org/web/20240626181513/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Kerngegevens-2013-Vewin.pdf

Vewin. (2014). *Kerngegevens drinkwater 2014*. Vewin. https://web.archive.org/web/20221005215957/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Kerngegevens%20Drinkwater_2014.pdf

Vewin. (2015). *Drinkwaterstatistieken 2015*. Vewin. https://web.archive.org/web/20220901053013/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Drinkwaterstatistieken_Vewin_2015.pdf

Vewin. (2016). *Kerngegevens drinkwater 2016*. Vewin. https://web.archive.org/web/20220615204430/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Cijfers/Vewin_Kerngegevens_2016-NL.pdf

Vewin. (2017a). *Drinkwaterstatistieken 2017*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/08/Drinkwaterstatistieken-2017-NL.pdf

Vewin. (2017b, July). *Watergebruik thuis 2016*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/06/watergebruik-thuis-2016.pdf

Vewin. (2018). *Kerngegevens drinkwater 2018*. Vewin. https://web.archive.org/web/20220325101633/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Cijfers/Kerngegevens2018-NL-web.pdf

Vewin. (2019). *Kerngegevens drinkwater 2019*. Vewin. https://web.archive.org/web/20221121090521/https://www.vewin.nl/SiteCollectionDocuments/Publicaties/Kerngegevens%20drinkwater%202019.pdf

Vewin. (2020). *Kerngegevens drinkwater 2020*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/06/vewin-kerngegevens-drinkwater-2020.pdf

Vewin. (2022). *Drinkwaterstatistieken 2022*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/06/vewin-drinkwaterstatistieken-2022-nl-web.pdf

Vewin. (2024a, June). *Kerngegevens 2023*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/06/5283-vewin-kerngegevens-2023-a6-staand_nl-web.pdf

Vewin. (2024b, June). *Kerngegevens drinkwater 2022*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/06/vewin-kerngegevens-drinkwater-2022.pdf

Vewin. (2024c, September). *Kerngegevens drinkwater 2024*. Vewin. https://www.vewin.nl/wp-content/uploads/2024/09/Vewin-Kerngegevens-Drinkwater-2024.pdf

Vewin. (2025, January). *Actieprogramma beschikbaarheid drinkwaterbronnen*. Vewin. https://www.vewin.nl/wp-content/uploads/2025/01/Actieprogramma-Beschikbaarheid-Drinkwaterbronnen_digitaal_def.pdf

Vewin. (n.d.). Vewin overzicht [Accessed: 2025-05-11]. https://www.vewin.nl/_layouts/15/vewin/webapp/

Vitens & Province of Gelderland. (2024). *Regionaal actieplan beschikbaarheid drinkwaterbronnen - vitens gelderland*. Vitens and Province of Gelderland. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-3-actieplan-drinkwaterbronnen-2023-2030-gelderland-vitens

Vitens & Province of Utrecht. (2024). *Regionaal actieplan beschikbaarheid drinkwaterbronnen - vitens utrecht*. Vitens and Province of Utrecht. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-10-regionaal-actieplan-utrecht-vitens

Waterbedrijf Groningen & Province of Groningen. (2023). *Regionaal actieplan beschikbaarheid drinkwaterbronnen - groningen*. Waterbedrijf Groningen and Province of Groningen. https://provinciegroningen.bestuurlijkeinformatie.nl/Document/View/f20f1a51-ef83-4cdc-83ec-9e25599eabd0

Waternet & Provinces of Noord-Holland and Utrecht. (2024). *Regionaal actieplan beschikbaarheid drinkwaterbronnen - waternet*. Waternet, Provinces of Noord-Holland, and Utrecht. https://www.rijksoverheid.nl/documenten/rapporten/2025/01/13/bijlage-11-regionaal-actieplan-utrecht-waternet

A

Selected Data

Table A.1: Overview of the average household size data selected for the model

Year	Modelling choice made	Source
2000-2019	The average size for a regular residential household is used to assess the amount of households connected to the DWDS.	(Centraal Bureau voor de Statistiek, 2019)
2020-2024	No data is available on the current average size for a regular residential household. To assess the amount of households connected to the DWDS linear behaviour between the two datasets is assumed.	-
2025-2050	The prognosis for the average size for a regular residential household is used to assess the amount of households connected to the DWDS.	(Centraal Bureau voor de Statistiek (CBS), 2023)

Table A.2: Overview of the average new construction rate data selected for the model

Year	Modelling choice made	Source
2000-2050	The prognosis for the average rate of construction is based on housing data from the period 2012-2024. For each province the ratio between newly built houses and the total housing supply is calculated on a yearly basis. Then for each drinking water company the average is calculated, according to the provinces they operate in.	(Centraal Bureau voor de Statistiek, 2025)

Table A.3: Overview of the extraction data selected for the model

Year	Modelling choice made	Source
2000-2006 2009 2011-2013 2015 2017-2019 2021-2022	The years 2000-2006 are assumed to have the same extraction data as in 2007. The other years without reported data are assumed to follow a linear pattern with the surrounding years. (Eg. for 2009 the data will be right in between the data of 2008 & 2010)	-
2007 2008 2010 2014 2016 2020	Data on extraction of fresh water by each company is taken from periodic reports. Water transport society Rijn-Kenemmerland as a company only extracts water. It is assumed that 50% is delivered to PWN, 25% to Waternet and 25% directly to industrial clients, which is based on historic distribution patterns.	Vewin (2007) Vewin (2008) Vewin (2012b) Vewin (2015) Vewin (2017a) Vewin (2022)
2023-2050	Data taken for 2023 from RIVM analysis on freshwater sources and this is set as a baseline. Afterwards the individual future plans of each company were used to increase the extraction capacity in line with their own insights.	van Leerdam et al. (2023), Vewin (2025), Brabant Water and Province of Noord-Brabant (2024), Province of Zuid-Holland (2024), Oasen and Provinces of Zuid-Holland and Utrecht (2024), PWN Waterleidingbedrijf Noord-Holland N.V. (2024), Vitens and Province of Gelderland (2024), Vitens and Province of Utrecht (2024), Waterbedrijf Groningen and Province of Groningen (2023), Province of Limburg (2023), Province of Drenthe (2023), Waternet and Provinces of Noord-Holland and Utrecht (2024)

Table A.4: Overview of the production data selected for the model

Year	Modelling choice made	Source
2000 2004-2006	The years 2000-2002 are assumed to have the same production data as in 2003. The other years without reported data are assumed to follow a linear pattern with the surrounding years. So for 2004-2006 the data increases linearly between the data of 2003 & 2007.	-
2008 2010 2014 2016 2020	Data on production of drinking water by each company is taken from periodic reports. These reports provide information on the quantities produced for industry and households.	Vewin (2008) Vewin (2012b) Vewin (2015) Vewin (2017a) Vewin (2022)
2003 2007 2009 2011-2013 2015 2017-2019 2020-2023	Data on production of drinking water by each company is taken from periodic reports. These reports do not provide information on the quantities produced for industry and households, only the total amount. Therefore the split between the two consumer groups is based on historic patterns.	Vewin (2005), Vewin (2007), Vewin (2009), Vewin (2012a), Vewin (2013), Vewin (2014), Vewin (2016), Vewin (2018), Vewin (2019), Vewin (2020), Vewin (2024b), Vewin (2024a), Vewin (2024c)
2024-2050	Data for 2024 onwards is extrapolated from the average division between the two consumer groups in the previous decades and combined with the average rate of production over extraction.	-

Table A.5: Overview of the end use data selected for the model

Year	Modelling choice made	Source
2000 2022-2050	The years 2000, and 2022-2050 are assumed to have the same end use data as the nearest year. In this manner, it is ensured that the end use data from 2022 onwards is only impacted by the external factors and policy levers of the model.	-
2004 2007 2010 2013 2016 2020	Data on each of the end use factors is taken from the period reports on the end use surveys. For showers and toilets there were no clear frequency splits in the earlier surveys, so the ratio of 2021 was applied to the frequency of the specific year.	Kanne (2005) Vewin (2011) TNS NIPO (2013) Vewin (2017b) Bakker et al. (2022)
2005-2006 2008-2009 2011-2012 2014-2015 2017-2019 2017-2019	For the years between the end use surveys it is assumed that the change between the data points is linear.	-
2024-2050	Data for 2024 onwards is extrapolated from the average division between the two consumer groups in the previous decades and combined with the average rate of production over extraction.	-

B

Python Notebooks

Municipalities 2000-2024

May 23, 2025

0.1 Data cleaning cencus data CBS - Municipalities (2000-2024)

[1]:
```
#Import of all relevant libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
#Retrieve population data
Pop_current = pd.read_csv("./Municipal Population 2000-2024.csv", sep=';')

#Rename columns
Pop_currentEN = Pop_current.rename(columns={"Perioden": "Year", "Regio's":␣
↪"Area", "Bevolking op 1 januari (aantal)" : "Population"})

#Create subsets
PC_NL = Pop_currentEN.loc[Pop_currentEN.Area == 'Nederland']
PC_Pro = Pop_currentEN.iloc[25:324]
PC_Pro.reset_index(inplace=True, drop=True)
PC_Mun = Pop_currentEN.iloc[325:]
PC_Mun.reset_index(inplace=True, drop=True)

#Fill NaN values for municipalities
PC_Mun = PC_Mun.fillna(0)

PC_Mun.head()
```

[2]:

	Year	Area	Population
0	2000	Aa en Hunze	24824.0
1	2001	Aa en Hunze	25208.0
2	2002	Aa en Hunze	25552.0
3	2003	Aa en Hunze	25305.0
4	2004	Aa en Hunze	25218.0

[3]:
```
#Identify municipalities that did not yet exist in 2000
PC_MunF = PC_Mun.loc[PC_Mun.Population == 0]

FusedM = PC_MunF.loc[(PC_MunF.Year == 2000)]
```

```
FusedM['Area'].unique()
```

```
[3]: array(['Altena', 'Beekdaelen', 'Berg en Dal', 'Berkelland',
            'Bodegraven-Reeuwijk', 'Bronckhorst', 'Dantumadiel',
            'Dijk en Waard', 'Dinkelland', 'Echt-Susteren', 'Eemsdelta',
            'Eijsden-Margraten', 'De Friese Meren', 'De Fryske Marren',
            'Geldrop-Mierlo', 'Goeree-Overflakkee', 'Gooise Meren',
            'Hoeksche Waard', 'Hof van Twente', 'Het Hogeland',
            'Hollands Kroon', 'Horst aan de Maas', 'Kaag en Braassem',
            'Koggenland', 'Krimpenerwaard', 'Land van Cuijk', 'Lansingerland',
            'Leidschendam-Voorburg', 'Leudal', 'Lingewaard', 'Maasgouw',
            'Maashorst', 'Meierijstad', 'Menameradiel', 'Midden-Delfland',
            'Midden-Groningen', 'Molenlanden', 'Molenwaard', 'Montferland',
            'Neder-Betuwe', 'Nissewaard', 'Noardeast-Fryslân', 'Oldambt',
            'Olst-Wijhe', 'Oost Gelre', 'Oude IJsselstreek', 'Overbetuwe',
            'Peel en Maas', 'Pijnacker-Nootdorp', 'Rijssen-Holten',
            'Sittard-Geleen', 'Sluis', 'Steenwijkerland', 'Stichtse Vecht',
            'Súdwest-Fryslân', 'Teylingen', 'Twenterand',
            'Utrechtse Heuvelrug', 'Vijfheerenlanden', 'Voorne aan Zee',
            'Waadhoeke', 'West Betuwe', 'Westerkwartier', 'Westerwolde',
            'Westland', 'Wijdemeren', 'Zuidplas', 'Zwartewaterland'],
           dtype=object)
```

0.1.1 Fuse together municipalities in line with their current status

```
[4]: # Define fusion mapping for fused municipalities with a new name
fusion_map = {
    'Altena': ['Aalburg', 'Werkendam', 'Woudrichem'],
    'Beekdaelen': ['Onderbanken', 'Nuth', 'Schinnen'],
    'Berg en Dal' : ['Groesbeek', 'Millingen aan de Rijn', 'Ubbergen'],
    'Berkelland' : ['Borculo', 'Eibergen', 'Neede', 'Ruurlo'],
    'Bodegraven-Reeuwijk': ['Bodegraven', 'Reeuwijk'],
    'Bronckhorst': ['Hengelo (Gld.)', 'Hummelo en Keppel', 'Steenderen',␣
↪'Vorden', 'Zelhem'],
    'Dantumadiel': ['Dantumadeel'],
    'Dijk en Waard': ['Heerhugowaard', 'Langedijk'],
    'Dinkelland': ['Ootmarsum', 'Denekamp', 'Weerselo'],
    'Echt-Susteren': ['Echt', 'Susteren'],
    'Eemsdelta': ['Appingedam', 'Delfzijl', 'Loppersum'],
    'Eijsden-Margraten': ['Eijsden', 'Margraten'],
    'De Friese Meren': ['Gaasterlân-Sleat', 'Lemsterland', 'Skarsterlân'],
    'De Fryske Marren': ['De Friese Meren'],
    'Geldrop-Mierlo': ['Geldrop', 'Mierlo'],
    'Goeree-Overflakkee': ['Goedereede', 'Dirksland', 'Middelharnis',␣
↪'Oostflakkee'],
    'Gooise Meren': ['Bussum', 'Naarden', 'Muiden'],
```

```
    'Hoeksche Waard': ['Oud-Beijerland', 'Korendijk', 'Strijen', 'Cromstrijen',␣
↪'Binnenmaas'],
    'Hof van Twente': ['Diepenheim', 'Goor', 'Markelo', 'Ambt Delden', 'Stad␣
↪Delden'],
    'Het Hogeland': ['Bedum', 'Eemsmond', 'De Marne', 'Winsum'],
    'Hollands Kroon': ['Wieringen', 'Wieringermeer', 'Anna Paulowna',␣
↪'Niedorp'],
    'Horst aan de Maas': ['Broekhuizen', 'Horst', 'Grubbenvorst', 'Sevenum',␣
↪'Meerlo-Wanssum'],
    'Kaag en Braassem': ['Alkemade', 'Jacobswoude'],
    'Koggenland': ['Obdam', 'Wester-Koggenland'],
    'Krimpenerwaard': ['Nederlek', 'Ouderkerk','Vlist','Bergambacht',␣
↪'Schoonhoven'],
    'Land van Cuijk': ['Boxmeer', 'Cuijk', 'Sint Anthonis', 'Mill en Sint␣
↪Hubert', 'Grave'],
    'Lansingerland': ['Berkel en Rodenrijs', 'Bleiswijk', 'Bergschenhoek'],
    'Leidschendam-Voorburg': ['Leidschendam', 'Voorburg'],
    'Leudal': ['Haelen', 'Heythuysen', 'Hunsel', 'Roggel en Neer'],
    'Lingewaard': ['Bemmel', 'Gendt', 'Huissen'],
    'Maasgouw': ['Heel', 'Maasbracht', 'Thorn'],
    'Maashorst': ['Landerd', 'Uden'],
    'Meierijstad': ['Schijndel', 'Sint-Oedenrode', 'Veghel'],
    'Menameradiel': ['Menaldumadeel'],
    'Midden-Delfland': ['Schipluiden', 'Maasland'],
    'Midden-Groningen': ['Hoogezand-Sappemeer', 'Slochteren', 'Menterwolde'],
    'Molenwaard' : ['Graafstroom', 'Liesveld','Nieuw-Lekkerland'],
    'Molenlanden': ['Molenwaard', 'Giessenlanden'],
    'Montferland': ['Bergh', 'Didam'],
    'Neder-Betuwe': ['Dodewaard', 'Echteld','Kesteren'],
    'Nissewaard': ['Spijkenisse', 'Bernisse'],
    'Noardeast-Fryslân' : ['Dongeradeel', 'Ferwerderadiel', 'Kollumerland en␣
↪Nieuwkruisland'],
    'Oldambt': ['Scheemda', 'Winschoten', 'Reiderland'],
    'Olst-Wijhe': ['Olst', 'Wijhe'],
    'Oost Gelre': ['Groenlo', 'Lichtenvoorde'],
    'Oude IJsselstreek': ['Wisch', 'Gendringen'],
    'Overbetuwe': ['Elst', 'Heteren', 'Valburg'],
    'Peel en Maas': ['Helden', 'Kessel', 'Meijel','Maasbree'],
    'Pijnacker-Nootdorp': ['Pijnacker', 'Nootdorp'],
    'Rijssen-Holten' : ['Rijssen', 'Holten'],
    'Sittard-Geleen': ['Sittard', 'Geleen', 'Born'],
    'Sluis': ['Oostburg', 'Sluis-Aardenburg'],
    'Steenwijkerland': ['Brederwiede', 'Steenwijk','IJsselham'],
    'Stichtse Vecht': ['Breukelen', 'Loenen', 'Maarssen'],
    'Súdwest-Fryslân': ['Bolsward', 'Nijefurd', 'Sneek', 'Wûnseradiel',␣
↪'Wymbritseradiel'],
```

```python
    'Teylingen': ['Sassenheim', 'Voorhout','Warmond'],
    'Twenterand': ['Den Ham', 'Vriezenveen'],
    'Utrechtse Heuvelrug': ['Amerongen', 'Doorn','Driebergen-Rijsenburg',␣
 ↪'Leersum', 'Maarn'],
    'Vijfheerenlanden': ['Leerdam', 'Vianen', 'Zederik'],
    'Voorne aan Zee': ['Brielle', 'Hellevoetsluis', 'Westvoorne'],
    'Waadhoeke': ['Franekeradeel', 'het Bildt','Menameradiel',␣
 ↪'Littenseradeel'],
    'West Betuwe': ['Geldermalsen', 'Neerijnen', 'Lingewaal'],
    'Westerkwartier': ['Grootegast', 'Leek', 'Marum', 'Zuidhorn'],
    'Westerwolde': ['Bellingwedde', 'Vlagtwedde'],
    'Westland': ['De Lier', "'s-Gravenzande", 'Monster', 'Naaldwijk',␣
 ↪'Wateringen'],
    'Wijdemeren': ['Loosdrecht', 'Nederhorst den Berg', "'s-Graveland"],
    'Zuidplas': ['Moordrecht', 'Nieuwerkerk aan den IJssel',␣
 ↪'Zevenhuizen-Moerkapelle'],
    'Zwartewaterland': ['Genemuiden', 'Zwartsluis', 'Hasselt'],
}

# Define absorption mapping for fused municipalities with an existing name
absorption_map = {
    'Aalten': ['Dinxperlo'],
    'Alkmaar': ['Graft-De Rijp', 'Schermer'],
    'Alphen aan den Rijn': ['Boskoop', 'Rijnwoude'],
    'Amsterdam': ['Weesp'],
    'Bergen (NH.)': ['Egmond', 'Schoorl'],
    'Binnenmaas': ["'s-Gravendeel"],
    'Bloemendaal': ['Bennebroek'],
    'Castricum': ['Akersloot', 'Limmen'],
    'Dalfsen': ['Nieuwleusen'],
    'De Ronde Venen': ['Abcoude'],
    'De Bilt': ['Maartensdijk'],
    'Deventer': ['Bathmen'],
    'Doetinchem': ['Wehl'],
    'Drechterland': ['Venhuizen'],
    'Edam-Volendam': ['Zeevang'],
    'Groningen (gemeente)': ['Haren', 'Ten Boer'],
    'Haarlemmermeer': ['Haarlemmerliede en Spaarnwoude'],
    'Hardenberg': ['Avereest', 'Gramsbergen'],
    'Heerenveen': ['Boarnsterhim'],
    'Hulst': ['Hontenisse'],
    'Kampen': ['IJsselmuiden'],
    'Katwijk': ['Rijnsburg', 'Valkenburg (ZH.)'],
    'Leeuwarden': ['Leeuwarderadeel', 'Littenseradiel'],
    'Lochem': ['Gorssel'],
    'Medemblik': ['Noorder-Koggenland', 'Wognum', 'Wervershoof','Andijk'],
    'Nieuwkoop': ['Ter Aar', 'Liemeer'],
```

```
    'Noordwijk': ['Noordwijkerhout'],
    'Oisterwijk': ['Haaren'],
    'Oss': ['Ravenstein', 'Lith', 'Maasdonk'],
    'Purmerend': ['Beemster'],
    'Raalte': ['Heino'],
    'Roerdalen': ['Ambt Montfort'],
    'Roermond': ['Swalmen'],
    'Rotterdam': ['Rozenburg'],
    'Schagen': ['Harenkarspel', 'Zijpe'],
    'Terneuzen': ['Axel', 'Sas van Gent'],
    'Utrecht (gemeente)': ['Vleuten-De Meern'],
    'Venlo': ['Tegelen', 'Belfeld','Arcen en Velden'],
    'Woerden': ['Harmelen'],
    'Zevenaar': ['Angerlo', 'Rijnwaarden'],
    'Zutphen': ['Warnsveld'],
    'Zwijndrecht': ['Heerjansdam']
}


def fill_fused_population(PC_Mun, fusion_map):
    """Fill in population values for fused municipalities without overwriting existing values, and remove their predecessors."""
    for fused_name, old_names in fusion_map.items():
        for year in PC_Mun['Year'].unique():
            # Sum the population of all predecessor municipalities
            total_pop = PC_Mun.loc[(PC_Mun['Year'] == year) & (PC_Mun['Area'].isin(old_names)), 'Population'].sum()

            # Only update the fused municipality if its population is 0
            mask = (PC_Mun['Year'] == year) & (PC_Mun['Area'] == fused_name) & (PC_Mun['Population'] == 0)
            PC_Mun.loc[mask, 'Population'] = total_pop

    # Remove all predecessor municipalities from the dataset
    fused_predecessors = [mun for mun_list in fusion_map.values() for mun in mun_list]
    PC_Mun = PC_Mun[~PC_Mun['Area'].isin(fused_predecessors)]

    return PC_Mun  # Return the modified DataFrame

def update_absorbed_population(PC_Mun, absorption_map):
    """Immediately merge absorbed municipalities into their new municipality from 2000 onwards."""
    for new_name, old_names in absorption_map.items():
        for year in PC_Mun['Year'].unique():
            # Sum the population of all absorbed municipalities
```

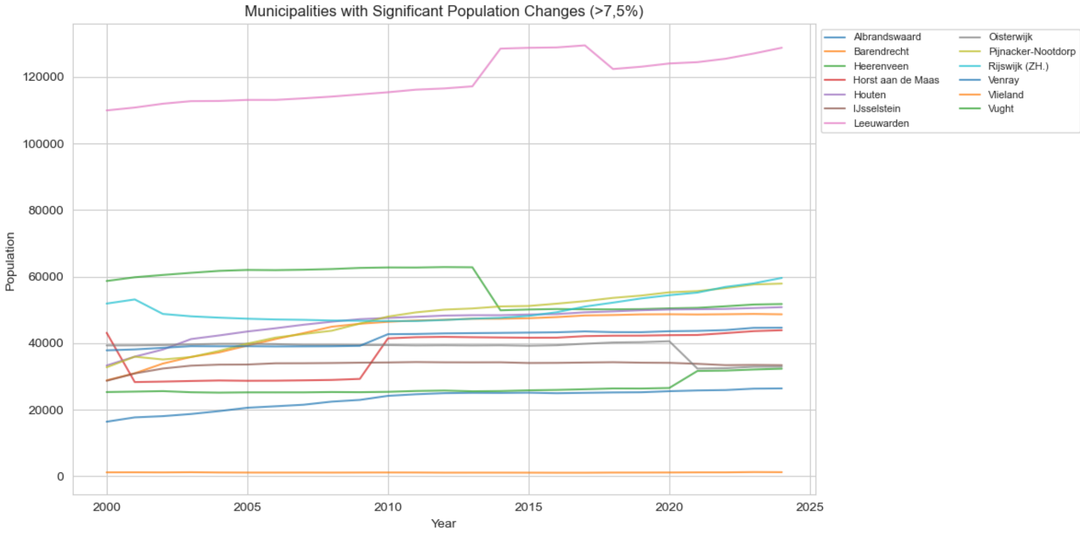
```
            total_pop = PC_Mun.loc[(PC_Mun['Year'] == year) & (PC_Mun['Area'].
↪isin(old_names)), 'Population'].sum()
            # Add absorbed municipalities' population to the absorbing␣
↪municipality
            PC_Mun.loc[(PC_Mun['Year'] == year) & (PC_Mun['Area'] == new_name),␣
↪'Population'] += total_pop

    # Remove all absorbed municipalities from the dataset entirely
    absorbed_municipalities = [mun for mun_list in absorption_map.values() for␣
↪mun in mun_list]
    PC_Mun = PC_Mun[~PC_Mun['Area'].isin(absorbed_municipalities)]

    return PC_Mun  # Return the modified DataFrame

# Apply transformations
PC_Mun = update_absorbed_population(PC_Mun, absorption_map)
PC_Mun = fill_fused_population(PC_Mun, fusion_map)
PC_Mun.to_excel("Municipal Population Fused 2000-2024.xlsx", index=False)

# Debug print before saving
print(PC_Mun[PC_Mun['Area'].isin(["Horst aan de Maas"])])
```

```
      Year               Area  Population
6200  2000  Horst aan de Maas     43156.0
6201  2001  Horst aan de Maas     28334.0
6202  2002  Horst aan de Maas     28476.0
6203  2003  Horst aan de Maas     28655.0
6204  2004  Horst aan de Maas     28813.0
6205  2005  Horst aan de Maas     28722.0
6206  2006  Horst aan de Maas     28750.0
6207  2007  Horst aan de Maas     28852.0
6208  2008  Horst aan de Maas     28975.0
6209  2009  Horst aan de Maas     29286.0
6210  2010  Horst aan de Maas     41465.0
6211  2011  Horst aan de Maas     41814.0
6212  2012  Horst aan de Maas     41917.0
6213  2013  Horst aan de Maas     41810.0
6214  2014  Horst aan de Maas     41727.0
6215  2015  Horst aan de Maas     41661.0
6216  2016  Horst aan de Maas     41675.0
6217  2017  Horst aan de Maas     42139.0
6218  2018  Horst aan de Maas     42271.0
6219  2019  Horst aan de Maas     42291.0
6220  2020  Horst aan de Maas     42429.0
6221  2021  Horst aan de Maas     42487.0
6222  2022  Horst aan de Maas     43039.0
6223  2023  Horst aan de Maas     43641.0
```

```
6224  2024  Horst aan de Maas     43917.0
```

0.1.2 Final check to identify strange mergers

[5]:
```python
# Compute the year-over-year percentage change
PC_Mun["Pop_Change"] = PC_Mun.groupby("Area")["Population"].pct_change()

# Define a threshold for "significant change" (e.g., more than 5% increase or␣
↪decrease)
threshold = 0.075

# Filter only municipalities where at least one year saw a significant change
changing_municipalities = PC_Mun.loc[
    (PC_Mun["Pop_Change"].abs() > threshold), "Area"
].unique()

# Filter the dataset to include only those municipalities
filtered_data = PC_Mun[PC_Mun["Area"].isin(changing_municipalities)]

# Set seaborn style
sns.set_style("whitegrid")

# Create the figure
plt.figure(figsize=(12, 6))

# Plot each selected municipality
for area in filtered_data["Area"].unique():
    subset = filtered_data[filtered_data["Area"] == area]
    plt.plot(subset["Year"], subset["Population"], label=area, alpha=0.7)

# Labels and title
plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Municipalities with Significant Population Changes (>7,5%)")

# Add a legend outside for readability
plt.legend(loc="upper left", bbox_to_anchor=(1, 1), fontsize=8, ncol=2)

# Show the plot
plt.tight_layout()
plt.show()
```



Municipalities with Significant Population Changes (>7,5%)
Population
Year
Albrandswaard
Barendrecht
Heerenveen
Horst aan de Maas
Houten
IJsselstein
Leeuwarden
Oisterwijk
Pijnacker-Nootdorp
Rijswijk (ZH.)
Venray
Vlieland
Vught

Municipalities 2000-2050

May 23, 2025

0.1 Data cleaning prognosis data (PBL) & fusing it with cleaned cencus data (CBS) - Municipalities (2000-2050)

[1]:
```
#Import of all relevant libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
from fuzzywuzzy import process
```

[2]:
```
#Load Fused CBS dataset
Pop_current = pd.read_excel("./Municipal Population Fused 2000-2024.xlsx")

#Load Prognosis PBL dataset - Expected population growth
Pop_future_short = pd.read_excel("./Municipal Population 2023-2050.xlsx",␣
↪sheet_name = "Bevolking", header = 59, usecols = "B, E:AD")

Pop_futureEN = Pop_future_short.rename(columns={"Naam": "Area"})

Pop_futureEN.head()
```

[2]:
```
         Area   2025   2026   2027   2028   2029   2030   2031   2032   2033  \
0   Groningen  246.1  250.0  253.6  256.7  259.3  261.1  262.4  263.3  264.2
1      Almere  227.4  230.2  233.3  236.5  239.7  243.1  246.3  249.6  253.1
2 Stadskanaal   31.9   31.8   31.7   31.5   31.2   30.9   30.6   30.4   30.1
3     Veendam   27.3   27.2   27.0   26.7   26.5   26.3   26.1   26.0   25.8
4    Zeewolde   24.1   24.3   24.5   24.6   24.7   24.9   25.0   25.1   25.2

   …   2041   2042   2043   2044   2045   2046   2047   2048   2049   2050
0  …  268.3  268.6  268.8  269.0  269.1  269.2  269.2  269.2  269.4  269.4
1  …  277.0  279.6  282.3  285.0  287.8  290.8  294.0  297.2  300.6  304.1
2  …   28.2   28.0   27.8   27.5   27.3   27.1   26.9   26.7   26.5   26.3
3  …   24.7   24.5   24.4   24.3   24.2   24.1   24.0   23.9   23.8   23.6
4  …   25.6   25.6   25.6   25.6   25.6   25.6   25.6   25.5   25.5   25.5

[5 rows x 27 columns]
```

[3]:
```
# Melt the DataFrame to long format
Pop_future = Pop_futureEN.melt(id_vars=["Area"], var_name="Year",␣
↪value_name="Population")

# Convert Year column to integer
Pop_future["Year"] = Pop_future["Year"].astype(int)

# Sort values for better readability
Pop_future = Pop_future.sort_values(by=["Area", "Year"]).reset_index(drop=True)

# Set population column to thousands
Pop_future["Population"] = 1000 * Pop_future["Population"]

# Display transformed DataFrame
Pop_future.head()
```

[3]:

	Area	Year	Population
0	Aa en Hunze	2025	24600.0
1	Aa en Hunze	2026	24300.0
2	Aa en Hunze	2027	24100.0
3	Aa en Hunze	2028	23900.0
4	Aa en Hunze	2029	23700.0

0.1.1 Combine the two datasets

[4]:
```
# Convert to sets for comparison
set1 = set(Pop_current["Area"])  # First dataset
set2 = set(Pop_future["Area"])  # Second dataset

# Find municipalities in df1 but not in df2
missing_in_future = set1 - set2
print("Municipalities in 2000-2024 but missing after 2024:", missing_in_future)

# Find municipalities in df2 but not in df1
missing_in_past = set2 - set1
print("Municipalities in 2025-2050 but missing before 2025:", missing_in_past)
```

```
Municipalities in 2000-2024 but missing after 2024: {'Dijk en Waard',
"'s-Gravenhage (gemeente)", 'Land van Cuijk', 'Utrecht (gemeente)', 'Middelburg
(Z.)', 'Laren (NH.)', 'Rijswijk (ZH.)', 'Groningen (gemeente)', 'Voorne aan
Zee', 'Hengelo (O.)', 'Stein (L.)', 'Beek (L.)', 'Maashorst',
"'s-Hertogenbosch"}
Municipalities in 2025-2050 but missing before 2025: {'Westvoorne', 'Boxmeer',
'Beek', 'Cuijk', 'Sint Anthonis', 'Brielle', 'Beemster', 'Stein', 'Hengelo',
'Rijswijk', 'Middelburg', 'Grave', 'Heerhugowaard', 'Mill en Sint Hubert',
'Landerd', 'Laren', 'Hellevoetsluis', 'Uden', 'Utrecht', 's-Gravenhage',
'Langedijk', 'Weesp', 's-Hertogenbosch', 'Groningen'}
```

[5]:
```python
# Function to find best matches
def find_similar_names(missing_names, reference_set, threshold=80):
    similar_names = {}
    for name in missing_names:
        match, score = process.extractOne(name, reference_set)
        if score >= threshold:
            similar_names[name] = match
    return similar_names

# Check for close matches
similar_in_past = find_similar_names(missing_in_future, set2)
similar_in_future = find_similar_names(missing_in_past, set1)

print("Possible similar names for 2000-2024 missing after 2024:", similar_in_past)
print("Possible similar names for 2025-2050 missing before 2025:", similar_in_future)
```

```
Possible similar names for 2000-2024 missing after 2024: {'Dijk en Waard':
'Nuenen Gerwen en Nederwetten', "'s-Gravenhage (gemeente)": 'Grave', 'Land van
Cuijk': 'Cuijk', 'Utrecht (gemeente)': 'Utrecht', 'Middelburg (Z.)':
'Middelburg', 'Laren (NH.)': 'Laren', 'Rijswijk (ZH.)': 'Rijswijk', 'Groningen
(gemeente)': 'Groningen', 'Voorne aan Zee': 'Capelle aan den IJssel', 'Hengelo
(O.)': 'Hengelo', 'Stein (L.)': 'Stein', 'Beek (L.)': 'Beek',
"'s-Hertogenbosch": 's-Hertogenbosch'}
Possible similar names for 2025-2050 missing before 2025: {'Beek': 'Beekdaelen',
'Cuijk': 'Land van Cuijk', 'Stein': 'Stein (L.)', 'Hengelo': 'Hengelo (O.)',
'Rijswijk': 'Rijswijk (ZH.)', 'Middelburg': 'Middelburg (Z.)', 'Grave':
"'s-Gravenhage (gemeente)", 'Mill en Sint Hubert': 'Aa en Hunze', 'Laren':
'Laren (NH.)', 'Hellevoetsluis': 'Sluis', 'Uden': 'Woudenberg', 'Utrecht':
'Utrechtse Heuvelrug', 's-Gravenhage': "'s-Gravenhage (gemeente)",
's-Hertogenbosch': "'s-Hertogenbosch", 'Groningen': 'Midden-Groningen'}
```

[6]:
```python
#Change names in current data set
name_corrections_current = {
    "Beek (L.)": "Beek",
    "'s-Gravenhage (gemeente)": "'s-Gravenhage",
    "Groningen (gemeente)": "Groningen",
    "Hengelo (O.)": "Hengelo",
    "Laren (NH.)": "Laren",
    "Middelburg (Z.)": "Middelburg",
    "Rijswijk (ZH.)": "Rijswijk",
    "Stein (L.)": "Stein",
    "Utrecht (gemeente)": "Utrecht",
    # Add more corrections here
}
```

```python
Pop_current["Area"] = Pop_current["Area"].replace(name_corrections_current)

#Change names in future data set
name_corrections_future = {
    "s-Gravenhage": "'s-Gravenhage",
    "s-Hertogenbosch": "'s-Hertogenbosch"
    # Add more corrections here
}

Pop_future["Area"] = Pop_future["Area"].replace(name_corrections_future)

set1 = set(Pop_current["Area"])
set2 = set(Pop_future["Area"])

print("Remaining unmatched in current:", set1 - set2)
print("Remaining unmatched in future:", set2 - set1)
```

```
Remaining unmatched in current: {'Dijk en Waard', 'Land van Cuijk', 'Voorne aan
Zee', 'Maashorst'}
Remaining unmatched in future: {'Grave', 'Westvoorne', 'Heerhugowaard',
'Boxmeer', 'Mill en Sint Hubert', 'Landerd', 'Cuijk', 'Sint Anthonis',
'Brielle', 'Hellevoetsluis', 'Weesp', 'Beemster', 'Langedijk', 'Uden'}
```

[7]:
```python
# Define fusion mapping
fusion_map = {
    'Dijk en Waard': ['Heerhugowaard', 'Langedijk'],
    'Land van Cuijk': ['Boxmeer', 'Cuijk', 'Sint Anthonis', 'Mill en Sint␣
↪Hubert', 'Grave'],
    'Maashorst': ['Landerd', 'Uden'],
    'Voorne aan Zee': ['Brielle', 'Hellevoetsluis', 'Westvoorne']
}

absorption_map = {
    'Amsterdam': ['Weesp'],
    'Purmerend': ['Beemster']
}

def fill_fused_population(Pop_future, fusion_map):
    """Add fused municipalities, sum their predecessors' populations, and␣
↪remove the predecessors."""

    # Ensure all fused municipalities exist in the dataset
    all_years = Pop_future['Year'].unique()
    missing_rows = []

    for fused_name, old_names in fusion_map.items():
        for year in all_years:
```

```python
            if not ((Pop_future['Year'] == year) & (Pop_future['Area'] == fused_name)).any():
                missing_rows.append({'Year': year, 'Area': fused_name, 'Population': 0})  # Placeholder row

    # Add missing municipalities to the dataset
    Pop_future = pd.concat([Pop_future, pd.DataFrame(missing_rows)], ignore_index=True)

    # Now sum populations and assign them to the fused municipalities
    for fused_name, old_names in fusion_map.items():
        for year in all_years:
            # Calculate total population of predecessor municipalities
            total_pop = Pop_future.loc[(Pop_future['Year'] == year) & (Pop_future['Area'].isin(old_names)), 'Population'].sum()

            # Assign total population to the fused municipality
            Pop_future.loc[(Pop_future['Year'] == year) & (Pop_future['Area'] == fused_name), 'Population'] = total_pop

    # Remove all predecessor municipalities
    fused_predecessors = [mun for mun_list in fusion_map.values() for mun in mun_list]
    Pop_future = Pop_future[~Pop_future['Area'].isin(fused_predecessors)]

    return Pop_future  # Return the modified DataFrame

def absorb_municipalities(Pop_future, absorption_map):
    """Absorb populations of smaller municipalities into larger ones and remove the absorbed ones."""

    for absorbing_mun, absorbed_muns in absorption_map.items():
        for year in Pop_future['Year'].unique():
            # Sum the population of absorbed municipalities
            total_pop = Pop_future.loc[(Pop_future['Year'] == year) & (Pop_future['Area'].isin(absorbed_muns)), 'Population'].sum()

            # Add the absorbed population to the absorbing municipality
            Pop_future.loc[(Pop_future['Year'] == year) & (Pop_future['Area'] == absorbing_mun), 'Population'] += total_pop

    # Remove all absorbed municipalities from the dataset
    absorbed_municipalities = [mun for mun_list in absorption_map.values() for mun in mun_list]
    Pop_future = Pop_future[~Pop_future['Area'].isin(absorbed_municipalities)]
```

```
    return Pop_future  # Return the modified DataFrame

# Apply the functions
Pop_future = fill_fused_population(Pop_future, fusion_map)
Pop_future = absorb_municipalities(Pop_future, absorption_map)

set1 = set(Pop_current["Area"])
set2 = set(Pop_future["Area"])

print("Remaining unmatched in current:", set1 - set2)
print("Remaining unmatched in future:", set2 - set1)
```

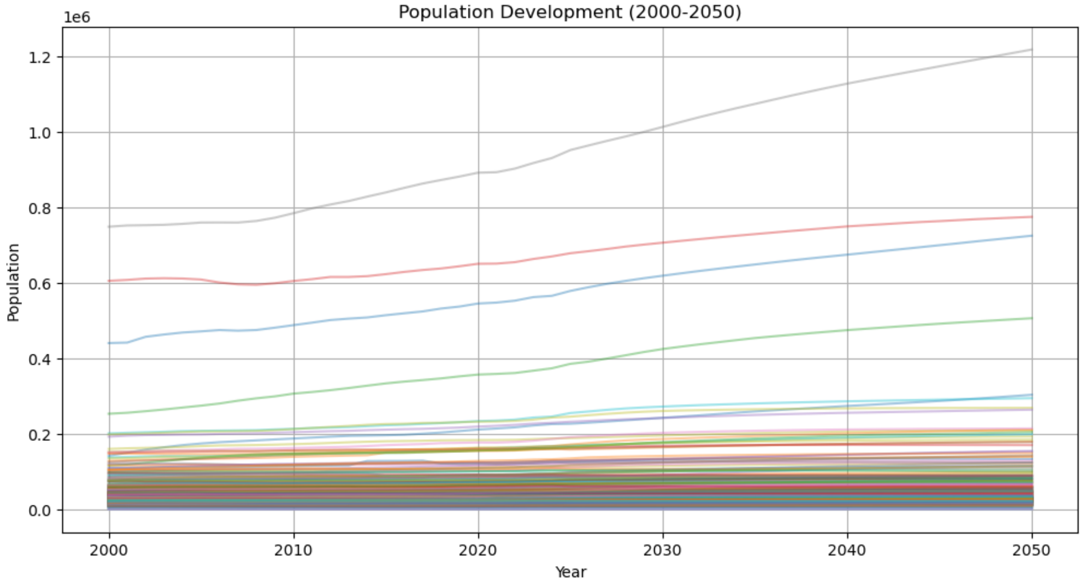
```
Remaining unmatched in current: set()
Remaining unmatched in future: set()
```

[8]:
```
#Combine the two datasets
Pop_future = Pop_future.sort_values(by=["Area", "Year"]).reset_index(drop=True)
Pop_current = Pop_current.sort_values(by=["Area", "Year"]).
 ↪reset_index(drop=True)

Pop_CandF = pd.concat([Pop_future, Pop_current], ignore_index=True)

Pop_CandF = Pop_CandF.sort_values(by=["Area", "Year"]).reset_index(drop=True)

Pop_CandF.head()
```

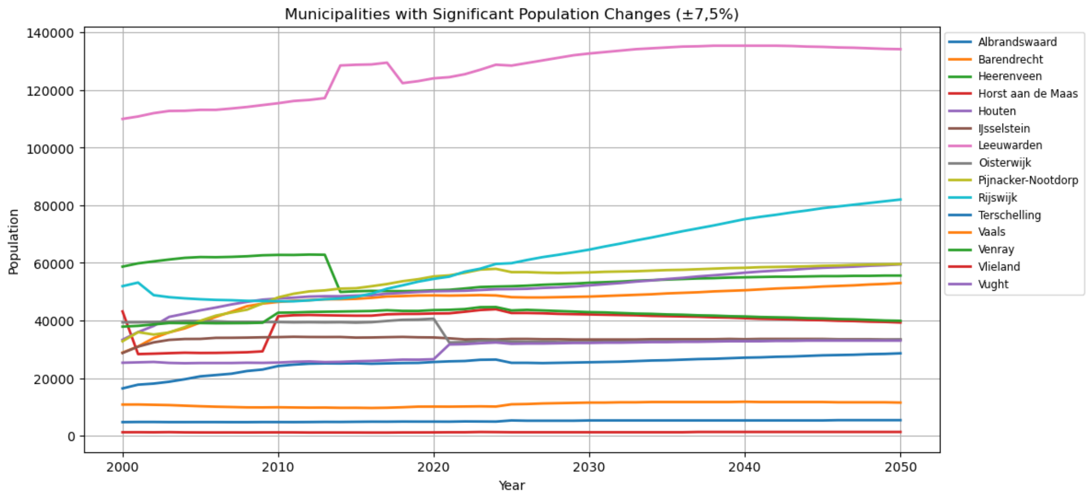
[8]:

	Area	Year	Population
0	's-Gravenhage	2000	441094.0
1	's-Gravenhage	2001	442356.0
2	's-Gravenhage	2002	457726.0
3	's-Gravenhage	2003	463826.0
4	's-Gravenhage	2004	469059.0

[9]:
```
plt.figure(figsize=(12, 6))

# Plot each municipality separately to look for anomalies
for area in Pop_CandF["Area"].unique():
    subset = Pop_CandF[Pop_CandF["Area"] == area]
    plt.plot(subset["Year"], subset["Population"], label=area, alpha=0.4)

plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Population Development (2000-2050)")
plt.grid(True)
plt.show()
```



[10]:
```
# Compute percentage change for each municipality
Pop_CandF["Pop_Change"] = Pop_CandF.groupby("Area")["Population"].pct_change()

# Define a threshold for significant change (e.g., ±5% change)
threshold = 0.075

# Find municipalities with large population jumps/drops
highlight_municipalities = Pop_CandF[Pop_CandF["Pop_Change"].abs() >␣
↪threshold]["Area"].unique()

plt.figure(figsize=(12, 6))

# Plot only municipalities that had significant changes
for area in highlight_municipalities:
    subset = Pop_CandF[Pop_CandF["Area"] == area]
    plt.plot(subset["Year"], subset["Population"], label=area, linewidth=2)

plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Municipalities with Significant Population Changes (±7,5%)")
plt.legend(loc="upper left", fontsize="small", bbox_to_anchor=(1, 1))
plt.grid(True)
plt.show()
```



[11]:
```
Pop_CandF = Pop_CandF.drop(columns=["Pop_Change"])

Pop_CandF.to_excel("Municipal Population Fused 2000-2050.xlsx", index=False)
```

0.1.2 Building the small Pop Growth Dataset

[12]:
```
#Load Prognosis PBL dataset - Low population growth
Pop_future_low_short = pd.read_excel("./Municipal Population 2023-2050.xlsx",␣
↪sheet_name = "Bevolking_Ondergrens", header = 59, usecols = "B, E:AD")

Pop_future_lowEN = Pop_future_low_short.rename(columns={"Naam": "Area"})

# Melt the DataFrame to long format
Pop_future_low = Pop_future_lowEN.melt(id_vars=["Area"], var_name="Year",␣
↪value_name="Population")

# Convert Year column to integer
Pop_future_low["Year"] = Pop_future_low["Year"].astype(int)

# Sort values for better readability
Pop_future_low = Pop_future_low.sort_values(by=["Area", "Year"]).
↪reset_index(drop=True)

# Set population column to thousands
Pop_future_low["Population"] = 1000 * Pop_future_low["Population"]

# Display transformed DataFrame
Pop_future_low.head()
```

[12]:

```
           Area  Year  Population
0  Aa en Hunze  2025     24100.0
1  Aa en Hunze  2026     23700.0
2  Aa en Hunze  2027     23400.0
3  Aa en Hunze  2028     23100.0
4  Aa en Hunze  2029     22900.0
```

[13]:

```
#Change names in future data set
name_corrections_future_low = {
    "s-Gravenhage": "'s-Gravenhage",
    "s-Hertogenbosch": "'s-Hertogenbosch"
    # Add more corrections here
}

Pop_future_low["Area"] = Pop_future_low["Area"].
 ↪replace(name_corrections_future_low)

# Define fusion mapping
fusion_map = {
    'Dijk en Waard': ['Heerhugowaard', 'Langedijk'],
    'Land van Cuijk': ['Boxmeer', 'Cuijk', 'Sint Anthonis', 'Mill en Sint␣
 ↪Hubert', 'Grave'],
    'Maashorst': ['Landerd', 'Uden'],
    'Voorne aan Zee': ['Brielle', 'Hellevoetsluis', 'Westvoorne']
}

absorption_map = {
    'Amsterdam': ['Weesp'],
    'Purmerend': ['Beemster']
}

def fill_fused_population(Pop_future_low, fusion_map):
    """Add fused municipalities, sum their predecessors' populations, and␣
 ↪remove the predecessors."""

    # Ensure all fused municipalities exist in the dataset
    all_years = Pop_future_low['Year'].unique()
    missing_rows = []

    for fused_name, old_names in fusion_map.items():
        for year in all_years:
            if not ((Pop_future_low['Year'] == year) & (Pop_future_low['Area']␣
 ↪== fused_name)).any():
                missing_rows.append({'Year': year, 'Area': fused_name,␣
 ↪'Population': 0})  # Placeholder row

    # Add missing municipalities to the dataset
```

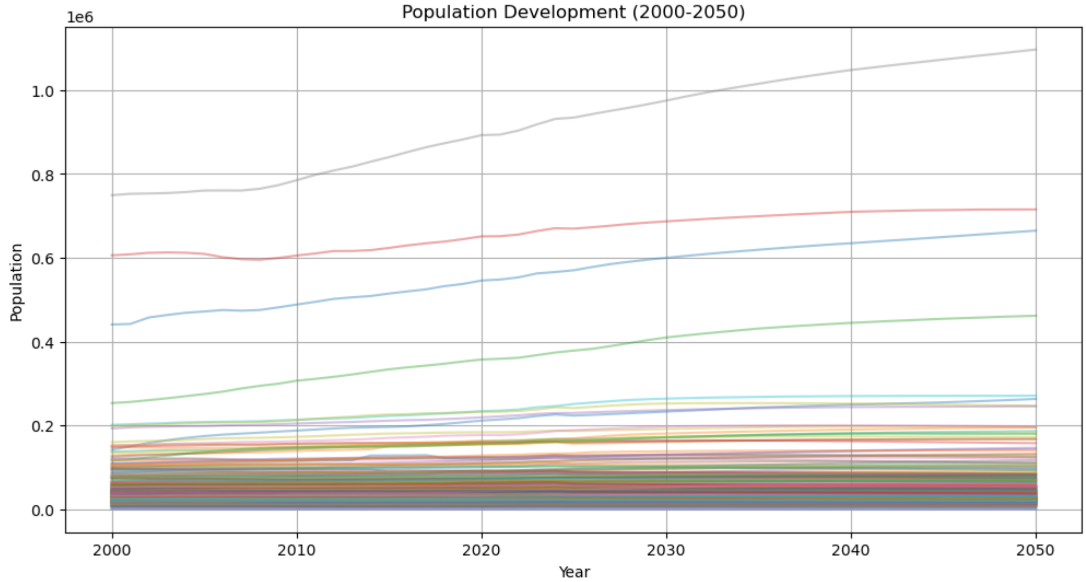
```python
    Pop_future_low = pd.concat([Pop_future_low, pd.DataFrame(missing_rows)], ignore_index=True)

    # Now sum populations and assign them to the fused municipalities
    for fused_name, old_names in fusion_map.items():
        for year in all_years:
            # Calculate total population of predecessor municipalities
            total_pop = Pop_future_low.loc[(Pop_future_low['Year'] == year) & (Pop_future_low['Area'].isin(old_names)), 'Population'].sum()

            # Assign total population to the fused municipality
            Pop_future_low.loc[(Pop_future_low['Year'] == year) & (Pop_future_low['Area'] == fused_name), 'Population'] = total_pop

    # Remove all predecessor municipalities
    fused_predecessors = [mun for mun_list in fusion_map.values() for mun in mun_list]
    Pop_future_low = Pop_future_low[~Pop_future_low['Area'].isin(fused_predecessors)]

    return Pop_future_low  # Return the modified DataFrame

def absorb_municipalities(Pop_future_low, absorption_map):
    """Absorb populations of smaller municipalities into larger ones and remove the absorbed ones."""

    for absorbing_mun, absorbed_muns in absorption_map.items():
        for year in Pop_future_low['Year'].unique():
            # Sum the population of absorbed municipalities
            total_pop = Pop_future_low.loc[(Pop_future_low['Year'] == year) & (Pop_future_low['Area'].isin(absorbed_muns)), 'Population'].sum()

            # Add the absorbed population to the absorbing municipality
            Pop_future_low.loc[(Pop_future_low['Year'] == year) & (Pop_future_low['Area'] == absorbing_mun), 'Population'] += total_pop

    # Remove all absorbed municipalities from the dataset
    absorbed_municipalities = [mun for mun_list in absorption_map.values() for mun in mun_list]
    Pop_future_low = Pop_future_low[~Pop_future_low['Area'].isin(absorbed_municipalities)]

    return Pop_future_low  # Return the modified DataFrame

# Apply the functions
Pop_future_low = fill_fused_population(Pop_future_low, fusion_map)
```

```
Pop_future_low = absorb_municipalities(Pop_future_low, absorption_map)

set1 = set(Pop_current["Area"])
set2 = set(Pop_future_low["Area"])

print("Remaining unmatched in current:", set1 - set2)
print("Remaining unmatched in future:", set2 - set1)
```

```
Remaining unmatched in current: set()
Remaining unmatched in future: set()
```

[14]:

```
#Combine the two datasets
Pop_CandF_low = pd.concat([Pop_future_low, Pop_current], ignore_index=True)

Pop_CandF_low = Pop_CandF_low.sort_values(by=["Area", "Year"]).
 ↪reset_index(drop=True)

#Save to excel
Pop_CandF_low.to_excel("Municipal Population Fused 2000-2050 (Low).xlsx",␣
 ↪index=False)

plt.figure(figsize=(12, 6))

# Plot each municipality separately to look for anomalies
for area in Pop_CandF_low["Area"].unique():
    subset = Pop_CandF_low[Pop_CandF_low["Area"] == area]
    plt.plot(subset["Year"], subset["Population"], label=area, alpha=0.4)

plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Population Development (2000-2050)")
plt.grid(True)
plt.show()
```



0.2 Building the large Pop Growth Dataset

[15]:
```
#Load Prognosis PBL dataset - High population growth
Pop_future_high_short = pd.read_excel("./Municipal Population 2023-2050.xlsx",␣
↪sheet_name = "Bevolking_Bovengrens", header = 59, usecols = "B, E:AD")

Pop_future_highEN = Pop_future_high_short.rename(columns={"Naam": "Area"})

Pop_future_highEN

# Melt the DataFrame to long format
Pop_future_high = Pop_future_highEN.melt(id_vars=["Area"], var_name="Year",␣
↪value_name="Population")

# Convert Year column to integer
Pop_future_high["Year"] = Pop_future_high["Year"].astype(int)

# Sort values for better readability
Pop_future_high = Pop_future_high.sort_values(by=["Area", "Year"]).
↪reset_index(drop=True)

# Set population column to thousands
Pop_future_high["Population"] = 1000 * Pop_future_high["Population"]

# Display transformed DataFrame
Pop_future_high.head()
```

[15]:
```
           Area  Year  Population
0  Aa en Hunze  2025     25100.0
1  Aa en Hunze  2026     24900.0
2  Aa en Hunze  2027     24800.0
3  Aa en Hunze  2028     24700.0
4  Aa en Hunze  2029     24600.0
```

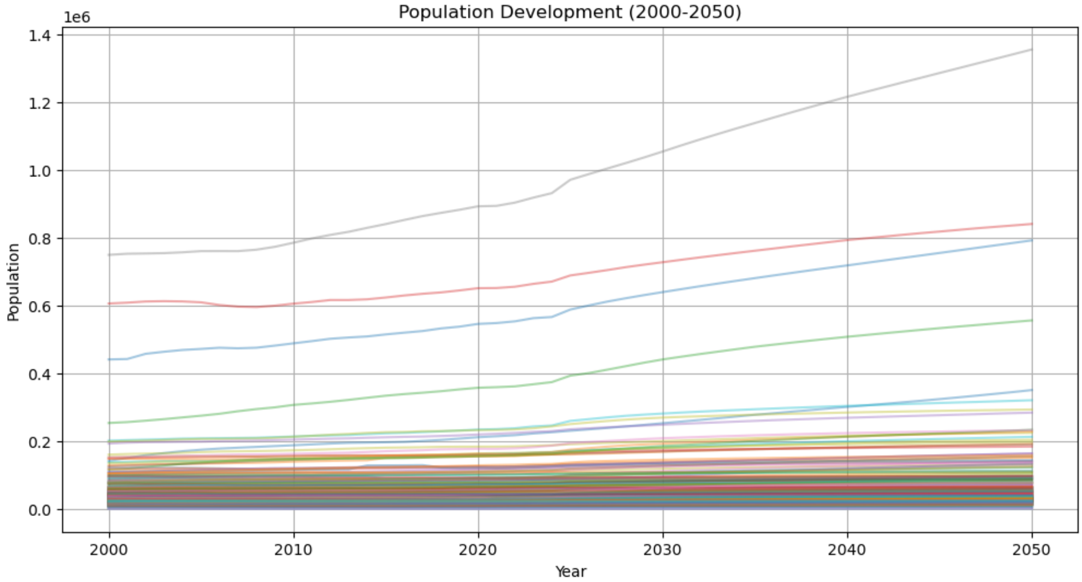
[16]:
```
#Change names in future data set
name_corrections_future_high = {
    "s-Gravenhage": "'s-Gravenhage",
    "s-Hertogenbosch": "'s-Hertogenbosch"
    # Add more corrections here
}

Pop_future_high["Area"] = Pop_future_high["Area"].
 ↪replace(name_corrections_future_high)

# Define fusion mapping
fusion_map = {
    'Dijk en Waard': ['Heerhugowaard', 'Langedijk'],
    'Land van Cuijk': ['Boxmeer', 'Cuijk', 'Sint Anthonis', 'Mill en Sint␣
 ↪Hubert', 'Grave'],
    'Maashorst': ['Landerd', 'Uden'],
    'Voorne aan Zee': ['Brielle', 'Hellevoetsluis', 'Westvoorne']
}

absorption_map = {
    'Amsterdam': ['Weesp'],
    'Purmerend': ['Beemster']
}

def fill_fused_population(Pop_future_high, fusion_map):
    """Add fused municipalities, sum their predecessors' populations, and␣
 ↪remove the predecessors."""

    # Ensure all fused municipalities exist in the dataset
    all_years = Pop_future_high['Year'].unique()
    missing_rows = []

    for fused_name, old_names in fusion_map.items():
        for year in all_years:
            if not ((Pop_future_high['Year'] == year) &␣
 ↪(Pop_future_high['Area'] == fused_name)).any():
                missing_rows.append({'Year': year, 'Area': fused_name,␣
 ↪'Population': 0})  # Placeholder row

    # Add missing municipalities to the dataset
```

```
    Pop_future_high = pd.concat([Pop_future_high, pd.DataFrame(missing_rows)],␣
↪ignore_index=True)

    # Now sum populations and assign them to the fused municipalities
    for fused_name, old_names in fusion_map.items():
        for year in all_years:
            # Calculate total population of predecessor municipalities
            total_pop = Pop_future_high.loc[(Pop_future_high['Year'] == year) &␣
↪(Pop_future_high['Area'].isin(old_names)), 'Population'].sum()

            # Assign total population to the fused municipality
            Pop_future_high.loc[(Pop_future_high['Year'] == year) &␣
↪(Pop_future_high['Area'] == fused_name), 'Population'] = total_pop

    # Remove all predecessor municipalities
    fused_predecessors = [mun for mun_list in fusion_map.values() for mun in␣
↪mun_list]
    Pop_future_high = Pop_future_high[~Pop_future_high['Area'].
↪isin(fused_predecessors)]

    return Pop_future_high  # Return the modified DataFrame

def absorb_municipalities(Pop_future_high, absorption_map):
    """Absorb populations of smaller municipalities into larger ones and remove␣
↪the absorbed ones."""

    for absorbing_mun, absorbed_muns in absorption_map.items():
        for year in Pop_future_low['Year'].unique():
            # Sum the population of absorbed municipalities
            total_pop = Pop_future_high.loc[(Pop_future_high['Year'] == year) &␣
↪(Pop_future_high['Area'].isin(absorbed_muns)), 'Population'].sum()

            # Add the absorbed population to the absorbing municipality
            Pop_future_high.loc[(Pop_future_high['Year'] == year) &␣
↪(Pop_future_high['Area'] == absorbing_mun), 'Population'] += total_pop

    # Remove all absorbed municipalities from the dataset
    absorbed_municipalities = [mun for mun_list in absorption_map.values() for␣
↪mun in mun_list]
    Pop_future_high = Pop_future_high[~Pop_future_high['Area'].
↪isin(absorbed_municipalities)]

    return Pop_future_high  # Return the modified DataFrame

# Apply the functions
Pop_future_high = fill_fused_population(Pop_future_high, fusion_map)
```

```
Pop_future_high = absorb_municipalities(Pop_future_high, absorption_map)

set1 = set(Pop_current["Area"])
set2 = set(Pop_future_high["Area"])

print("Remaining unmatched in current:", set1 - set2)
print("Remaining unmatched in future:", set2 - set1)
```

```
Remaining unmatched in current: set()
Remaining unmatched in future: set()
```

[17]:
```
Pop_CandF_high = pd.concat([Pop_future_high, Pop_current], ignore_index=True)

Pop_CandF_high = Pop_CandF_high.sort_values(by=["Area", "Year"]).
  ↪reset_index(drop=True)

Pop_CandF_high.to_excel("Municipal Population Fused 2000-2050 (High).xlsx",␣
  ↪index=False)

plt.figure(figsize=(12, 6))

# Plot each municipality separately to look for anomalies
for area in Pop_CandF_high["Area"].unique():
    subset = Pop_CandF_high[Pop_CandF_high["Area"] == area]
    plt.plot(subset["Year"], subset["Population"], label=area, alpha=0.4)

plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Population Development (2000-2050)")
plt.grid(True)
plt.show()
```


Population Development (2000-2050)
1e6
1.4
1.2
1.0
0.8
0.6
0.4
0.2
0.0
Population
2000
2010
2020
2030
2040
2050
Year

Water Companies 2000-2050

May 23, 2025

0.1 Transforming the municipality population data (CBS & PBL) to drinking water companies

[1]:
```
#Import of all relevant libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
#Load fused municipal population data
Pop_fused = pd.read_excel("./Municipal Population Fused 2000-2050.xlsx")

Pop_fused.head()
```

[2]:
```
             Area  Year  Population
0  's-Gravenhage  2000    441094.0
1  's-Gravenhage  2001    442356.0
2  's-Gravenhage  2002    457726.0
3  's-Gravenhage  2003    463826.0
4  's-Gravenhage  2004    469059.0
```

[3]:
```
#Create fusion mapping for drinking water companies
fusion_map = {
    'Puur Water & Natuur (PWN)': ['Aalsmeer', 'Alkmaar', 'Beemster', 'Bergen␣
 ↪(NH.)', 'Beverwijk', 'Blaricum', 'Bloemendaal', 'Bussum',
                                  'Castricum', 'Dijk en Waard', 'Drechterland',␣
 ↪'Edam-Volendam', 'Enkhuizen', 'Gooise Meren', 'Haarlem',
                                  'Haarlemmerliede & Spaarnwoude',␣
 ↪'Haarlemmermeer', 'Heemskerk', 'Heerhugowaard', 'Heiloo', 'Den Helder',
                                  'Hollands Kroon', 'Hoorn', 'Huizen',␣
 ↪'Koggenland', 'Landsmeer', 'Langedijk', 'Laren', 'Medemblik', 'Naarden',
                                  'Oostzaan', 'Opmeer', 'Purmerend', 'Schagen',␣
 ↪'Stede Broec', 'Texel', 'Uitgeest', 'Uithoorn', 'Velsen', 'Waterland',
                                  'Weesp', 'Wormerland', 'Wijdemeren',␣
 ↪'Zaanstad', 'Zandvoort'],
    'Waterbedrijf Groningen': ['Eemsdelta', 'Groningen', 'Het Hogeland',␣
 ↪'Midden-Groningen', 'Oldambt', 'Pekela', 'Stadskanaal',
```

```
                                 'Veendam', 'Westerkwartier', 'Westerwolde'],
   'Waterleidingmaatschappij Drenthe (WMD)': ['Aa en Hunze','Assen',␣
↪'Borger-Odoorn', 'Coevorden', 'Emmen', 'Hoogeveen',
                                                   'Midden-Drenthe', 'Noordenveld',␣
↪'Tynaarlo', 'De Wolden'],
   'Vitens': [
       # Friesland
       'Achtkarspelen', 'Ameland', 'Dantumadiel', 'De Fryske Marren',␣
↪'Harlingen', 'Heerenveen', 'Leeuwarden', 'Noardeast-Fryslân',
       'Ooststellingwerf', 'Opsterland', 'Schiermonnikoog', 'Smallingerland',␣
↪'Súdwest-Fryslân', 'Terschelling', 'Tytsjerksteradiel',
       'Vlieland', 'Waadhoeke', 'Weststellingwerf',

       # Drenthe
       'Meppel', 'Westerveld',

       # Overijssel
       'Almelo', 'Borne', 'Dalfsen', 'Deventer', 'Dinkelland', 'Enschede',␣
↪'Haaksbergen', 'Hardenberg',
       'Hellendoorn', 'Hengelo', 'Hof van Twente', 'Kampen', 'Losser',␣
↪'Oldenzaal', 'Olst-Wijhe', 'Ommen',
       'Raalte', 'Rijssen-Holten', 'Staphorst', 'Steenwijkerland',␣
↪'Tubbergen', 'Twenterand', 'Wierden', 'Zwartewaterland', 'Zwolle',

       # Gelderland
       'Aalten', 'Apeldoorn', 'Arnhem', 'Barneveld', 'Berg en Dal',␣
↪'Berkelland', 'Beuningen', 'Bronckhorst',
       'Brummen', 'Buren', 'Culemborg', 'Doesburg', 'Doetinchem', 'Druten',␣
↪'Duiven', 'Ede', 'Elburg', 'Epe',
       'Ermelo', 'Harderwijk', 'Hattem', 'Heerde', 'Heumen', 'Lingewaard',␣
↪'Lochem', 'Maasdriel', 'Montferland',
       'Neder-Betuwe', 'Nijkerk', 'Nijmegen', 'Nunspeet', 'Oldebroek', 'Oost␣
↪Gelre', 'Oude IJsselstreek',
       'Overbetuwe', 'Putten', 'Renkum', 'Rheden', 'Rozendaal',␣
↪'Scherpenzeel', 'Tiel', 'Voorst', 'Wageningen',
       'West Betuwe', 'West Maas en Waal', 'Westervoort', 'Wijchen',␣
↪'Winterswijk', 'Zaltbommel', 'Zevenaar', 'Zutphen',

       # Flevoland
       'Almere', 'Dronten', 'Lelystad', 'Noordoostpolder', 'Urk', 'Zeewolde',

       # Utrecht
       'Amersfoort', 'Baarn', 'Bunnik', 'Bunschoten', 'De Bilt', 'De Ronde␣
↪Venen', 'Eemnes', 'Houten',
       'IJsselstein', 'Leusden', 'Lopik', 'Montfoort', 'Nieuwegein',␣
↪'Oudewater', 'Renswoude', 'Rhenen',
```

```
        'Soest', 'Stichtse Vecht', 'Utrecht', 'Utrechtse Heuvelrug',␣
↪'Veenendaal', 'Wijk bij Duurstede',
        'Woerden', 'Woudenberg', 'Zeist',

        # Noord-Holland
        'Hilversum',
    ],
    'Waternet': ['Amsterdam','Amstelveen', 'Diemen', 'Heemstede',␣
↪'Ouder-Amstel'],
    'Oasen': ['Alblasserdam', 'Alphen aan den Rijn','Bodegraven-Reeuwijk',␣
↪'Gorinchem', 'Gouda', 'Hardinxveld-Giessendam',
             'Hendrik-Ido-Ambacht', 'Kaag en Braassem', 'Krimpen aan den␣
↪IJssel', 'Krimpenerwaard', 'Leiderdorp', 'Molenlanden',
             'Nieuwkoop', 'Papendrecht', 'Ridderkerk', 'Sliedrecht',␣
↪'Vijfheerenlanden', 'Waddinxveen', 'Zoeterwoude', 'Zwijndrecht'],
    'Dunea': ["'s-Gravenhage",'Hillegom', 'Katwijk', 'Lansingerland', 'Leiden',␣
↪'Leidschendam-Voorburg', 'Lisse', 'Noordwijk',
             'Oegstgeest',  'Pijnacker-Nootdorp', 'Rijswijk', 'Teylingen',␣
↪'Voorschoten', 'Wassenaar', 'Zoetermeer', 'Zuidplas'],
    'Brabant Water': ['Alphen-Chaam', 'Altena', 'Asten', 'Baarle-Nassau',␣
↪'Bergeijk', 'Bergen op Zoom', 'Bernheze', 'Best', 'Bladel',
                      'Boekel', 'Boxtel', 'Breda', 'Cranendonck', 'Deurne',␣
↪'Dongen', 'Drimmelen', 'Eersel', 'Eindhoven', 'Etten-Leur',
                      'Geertruidenberg', 'Geldrop-Mierlo', 'Gemert-Bakel',␣
↪'Gilze en Rijen', 'Goirle', 'Haaren', 'Halderberge',
                      'Heeze-Leende', 'Helmond', "'s-Hertogenbosch", 'Heusden',␣
↪'Hilvarenbeek', 'Laarbeek', 'Land van Cuijk',
                      'Loon op Zand', 'Maashorst', 'Meierijstad',␣
↪'Sint-Michielsgestel', 'Moerdijk', 'Nuenen Gerwen en Nederwetten',
                      'Oirschot', 'Oisterwijk', 'Oosterhout', 'Oss', 'Reusel-De␣
↪Mierden', 'Roosendaal', 'Rucphen', 'Someren',
                      'Son en Breugel', 'Steenbergen', 'Tilburg',␣
↪'Valkenswaard', 'Veldhoven', 'Vught', 'Waalre', 'Waalwijk', 'Zundert'],
    'Waterleiding Maatschappij Limburg (WML)': ['Beek', 'Beekdaelen', 'Beesel',␣
↪'Bergen (L.)', 'Brunssum', 'Echt-Susteren', 'Eijsden-Margraten', 'Gennep',␣
↪'Gulpen-Wittem',
                                                 'Heerlen', 'Horst aan de Maas',␣
↪'Kerkrade', 'Landgraaf', 'Leudal', 'Maasgouw', 'Maastricht', 'Meerssen',␣
↪'Mook en Middelaar',
                                                 'Nederweert', 'Peel en Maas',␣
↪'Roerdalen', 'Roermond', 'Simpelveld', 'Sittard-Geleen', 'Stein', 'Vaals',␣
↪'Valkenburg aan de Geul',
                                                 'Venlo', 'Venray',␣
↪'Voerendaal', 'Weert'],
```

```python
    'Evides': ['Albrandswaard', 'Barendrecht', 'Borsele', 'Capelle aan den IJssel', 'Delft', 'Dordrecht', 'Goeree-Overflakkee', 'Goes', 'Hoeksche Waard', 'Hulst',
               'Kapelle', 'Maassluis', 'Middelburg', 'Midden-Delfland', 'Nissewaard', 'Noord-Beveland', 'Reimerswaal', 'Rotterdam', 'Schiedam',
               'Schouwen-Duiveland', 'Sluis', 'Terneuzen', 'Tholen', 'Veere', 'Vlaardingen', 'Vlissingen', 'Voorne aan Zee', 'Westland', 'Woensdrecht']
}

def fill_fused_population(Pop_fused, fusion_map):
    """Aggregate municipalities into their respective water companies by summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused['Year'] == year) & (Pop_fused['Area'] == company)).any():
                missing_rows.append({'Year': year, 'Area': company, 'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused = pd.concat([Pop_fused, pd.DataFrame(missing_rows)], ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this water company
            total_pop = Pop_fused.loc[(Pop_fused['Year'] == year) & (Pop_fused['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused.loc[(Pop_fused['Year'] == year) & (Pop_fused['Area'] == company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun in mun_list]
    Pop_fused = Pop_fused[~Pop_fused['Area'].isin(absorbed_municipalities)]

    return Pop_fused  # Return the modified Data
```

```
#Run the function
Pop_fused = fill_fused_population(Pop_fused, fusion_map)

Pop_fused['Area'].unique()
```

[3]:
```
array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
       'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
       'Oasen', 'Dunea', 'Brabant Water',
       'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

[4]:
```
def check_duplicate_municipalities(fusion_map):
    """Check if any municipality has been assigned to multiple water companies.
 ↪"""

    # Dictionary to track which companies each municipality belongs to
    municipality_to_companies = {}

    for company, municipalities in fusion_map.items():
        for municipality in municipalities:
            if municipality not in municipality_to_companies:
                municipality_to_companies[municipality] = []
            municipality_to_companies[municipality].append(company)

    # Find municipalities assigned to multiple water companies
    duplicates = {mun: comps for mun, comps in municipality_to_companies.
 ↪items() if len(comps) > 1}

    if duplicates:
        print("  The following municipalities are assigned to multiple water␣
 ↪companies:")
        for mun, comps in duplicates.items():
            print(f"  - {mun}: {', '.join(comps)}")
    else:
        print("  No duplicate assignments found!")

    return duplicates  # Return dictionary for further inspection if needed

# Run the function on your fusion_map
duplicates = check_duplicate_municipalities(fusion_map)
```

```
No duplicate assignments found!
```

0.1.1 Repeat for low population growth

[5]:
```python
Pop_fused_low = pd.read_excel("./Municipal Population Fused 2000-2050 (Low).
↪xlsx")

def fill_fused_population(Pop_fused_low, fusion_map):
    """Aggregate municipalities into their respective water companies by␣
↪summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused_low['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused_low['Year'] == year) & (Pop_fused_low['Area'] ==␣
↪company)).any():
                missing_rows.append({'Year': year, 'Area': company,␣
↪'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused_low = pd.concat([Pop_fused_low, pd.DataFrame(missing_rows)],␣
↪ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this␣
↪water company
            total_pop = Pop_fused_low.loc[(Pop_fused_low['Year'] == year) &␣
↪(Pop_fused_low['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused_low.loc[(Pop_fused_low['Year'] == year) &␣
↪(Pop_fused_low['Area'] == company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun␣
↪in mun_list]
    Pop_fused_low = Pop_fused_low[~Pop_fused_low['Area'].
↪isin(absorbed_municipalities)]

    return Pop_fused_low  # Return the modified Data

#Run the function
Pop_fused_low = fill_fused_population(Pop_fused_low, fusion_map)
```

```
Pop_fused_low['Area'].unique()
```

```
[5]: array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
            'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
            'Oasen', 'Dunea', 'Brabant Water',
            'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

0.1.2 Repeat for high population growth

```
[6]: Pop_fused_high = pd.read_excel("./Municipal Population Fused 2000-2050 (High).
     ↪xlsx")

def fill_fused_population(Pop_fused_high, fusion_map):
    """Aggregate municipalities into their respective water companies by␣
 ↪summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused_high['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused_high['Year'] == year) & (Pop_fused_high['Area']␣
 ↪== company)).any():
                missing_rows.append({'Year': year, 'Area': company,␣
 ↪'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused_high = pd.concat([Pop_fused_high, pd.DataFrame(missing_rows)],␣
 ↪ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this␣
 ↪water company
            total_pop = Pop_fused_high.loc[(Pop_fused_high['Year'] == year) &␣
 ↪(Pop_fused_high['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused_high.loc[(Pop_fused_high['Year'] == year) &␣
 ↪(Pop_fused_high['Area'] == company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun␣
 ↪in mun_list]
```

```
    Pop_fused_high = Pop_fused_high[~Pop_fused_high['Area'].
 ↪isin(absorbed_municipalities)]

    return Pop_fused_high  # Return the modified Data

#Run the function
Pop_fused_high = fill_fused_population(Pop_fused_high, fusion_map)

Pop_fused_high['Area'].unique()
```

[6]:
```
array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
       'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
       'Oasen', 'Dunea', 'Brabant Water',
       'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

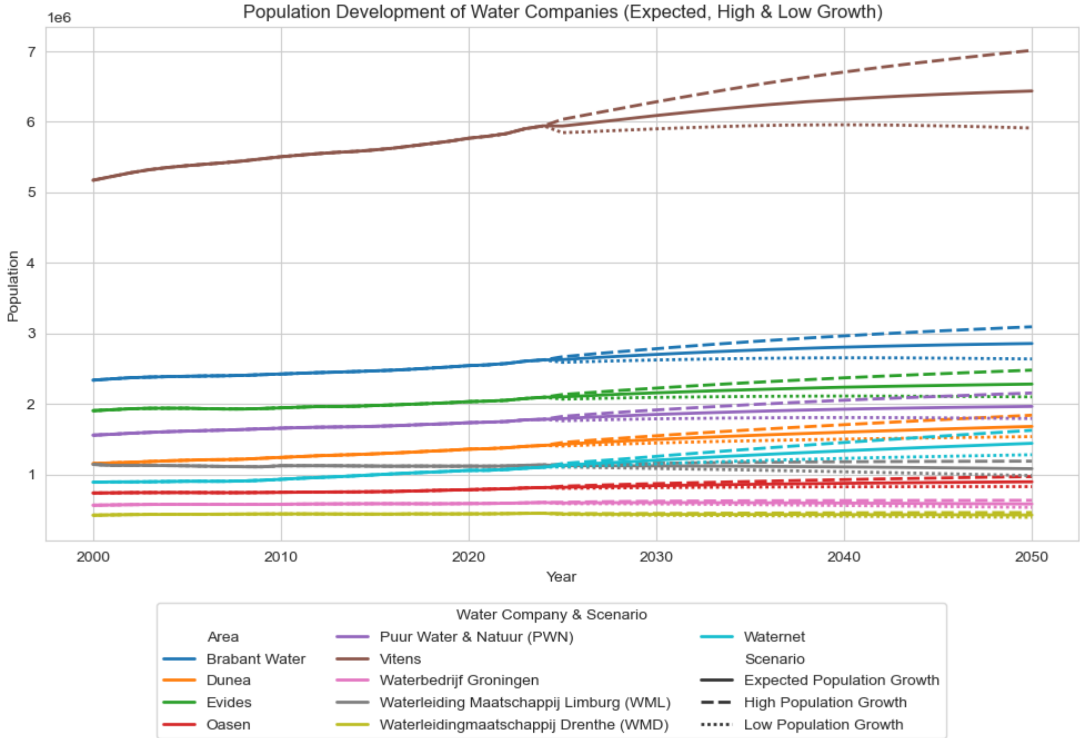
[7]:
```
# Fuse together the three datasets
Pop_fused['High Population Growth'] = Pop_fused_high['Population']

Pop_fused['Low Population Growth'] = Pop_fused_low['Population']

Pop_fused = Pop_fused.rename(columns={"Population": "Expected Population␣
 ↪Growth"})

Pop_fused = Pop_fused.sort_values(by=["Area", "Year"]).reset_index(drop=True)

Pop_fused.head()
```

[7]:
```
            Area  Year  Expected Population Growth  High Population Growth  \
0  Brabant Water  2000                   2335009.0               2335009.0
1  Brabant Water  2001                   2353660.0               2353660.0
2  Brabant Water  2002                   2369575.0               2369575.0
3  Brabant Water  2003                   2378641.0               2378641.0
4  Brabant Water  2004                   2385579.0               2385579.0

   Low Population Growth
0              2335009.0
1              2353660.0
2              2369575.0
3              2378641.0
4              2385579.0
```

0.1.3 Create a figure for population growth & save to Excel

[8]:
```
# Set Seaborn style
sns.set_style("whitegrid")

# Convert DataFrame from wide format to long format for easier plotting
```

```python
Pop_long = Pop_fused.melt(id_vars=["Year", "Area"],
                          value_vars=["Expected Population Growth", "High Population Growth", "Low Population Growth"],
                          var_name="Scenario",
                          value_name="No. of Citizens")

# Add this filter to focus on specific water companies
selected_companies = ['Puur Water & Natuur (PWN)',
                      'Waterbedrijf Groningen',
                      'Waterleidingmaatschappij Drenthe (WMD)',
                      'Vitens',
                      'Waternet',
                      'Oasen',
                      'Dunea',
                      'Brabant Water',
                      'Waterleiding Maatschappij Limburg (WML)',
                      'Evides'
                     ]  # Modify this list

Pop_filtered = Pop_long[Pop_long["Area"].isin(selected_companies)]

# Create the plot
plt.figure(figsize=(12, 6))
sns.lineplot(data=Pop_filtered, x="Year", y="No. of Citizens", hue="Area", style="Scenario", linewidth=2)

# Labels and title
plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Population Development of Water Companies (Expected, High & Low Growth)")

# Improve legend readability
plt.legend(title="Water Company & Scenario",  bbox_to_anchor=(0.1, -0.4), loc="lower left", ncols = 3)

#  Save the plot
plt.savefig("population_water_companies_lowlegend.png", dpi=300, bbox_inches="tight")

# Show the plot
plt.show()
```



[9]:
```
#Save the scenarios to Excel sheets
Pop_BW = Pop_fused.loc[Pop_fused['Area']== 'Brabant Water']
Pop_D = Pop_fused.loc[Pop_fused['Area']== 'Dunea']
Pop_E = Pop_fused.loc[Pop_fused['Area']== 'Evides']
Pop_O = Pop_fused.loc[Pop_fused['Area']== 'Oasen']
Pop_PWN = Pop_fused.loc[Pop_fused['Area']== 'Puur Water & Natuur (PWN)']
Pop_V = Pop_fused.loc[Pop_fused['Area']== 'Vitens']
Pop_WBGR = Pop_fused.loc[Pop_fused['Area']== 'Waterbedrijf Groningen']
Pop_WML = Pop_fused.loc[Pop_fused['Area']== 'Waterleiding Maatschappij Limburg␣
↪(WML)']
Pop_WMD = Pop_fused.loc[Pop_fused['Area']== 'Waterleidingmaatschappij Drenthe␣
↪(WMD)']
Pop_W = Pop_fused.loc[Pop_fused['Area']== 'Waternet']


with pd.ExcelWriter('Water Company Population 2000-2050 (Growth Scenarios).
↪xlsx') as writer:
    Pop_BW.to_excel(writer, sheet_name='Brabant Water')
    Pop_D.to_excel(writer, sheet_name='Dunea')
    Pop_E.to_excel(writer, sheet_name='Evides')
    Pop_O.to_excel(writer, sheet_name='Oasen')
    Pop_PWN.to_excel(writer, sheet_name='PWN')
```

```
    Pop_V.to_excel(writer, sheet_name='Vitens')
    Pop_WBGR.to_excel(writer, sheet_name='Wb Gr')
    Pop_WML.to_excel(writer, sheet_name='WML')
    Pop_WMD.to_excel(writer, sheet_name='WMD')
    Pop_W.to_excel(writer, sheet_name='Waternet')
```

Water Companies 2000-2050

May 23, 2025

0.1 Transforming the municipality population data (CBS & PBL) to drinking water companies

[1]:
```
#Import of all relevant libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

[2]:
```
#Load fused municipal population data
Pop_fused = pd.read_excel("./Municipal Population Fused 2000-2050.xlsx")

Pop_fused.head()
```

[2]:

	Area	Year	Population
0	's-Gravenhage	2000	441094.0
1	's-Gravenhage	2001	442356.0
2	's-Gravenhage	2002	457726.0
3	's-Gravenhage	2003	463826.0
4	's-Gravenhage	2004	469059.0

[3]:
```
#Create fusion mapping for drinking water companies
fusion_map = {
    'Puur Water & Natuur (PWN)': ['Aalsmeer', 'Alkmaar', 'Beemster', 'Bergen␣
↪(NH.)', 'Beverwijk', 'Blaricum', 'Bloemendaal', 'Bussum',
                              'Castricum', 'Dijk en Waard', 'Drechterland',␣
↪'Edam-Volendam', 'Enkhuizen', 'Gooise Meren', 'Haarlem',
                              'Haarlemmerliede & Spaarnwoude',␣
↪'Haarlemmermeer', 'Heemskerk', 'Heerhugowaard', 'Heiloo', 'Den Helder',
                              'Hollands Kroon', 'Hoorn', 'Huizen',␣
↪'Koggenland', 'Landsmeer', 'Langedijk', 'Laren', 'Medemblik', 'Naarden',
                              'Oostzaan', 'Opmeer', 'Purmerend', 'Schagen',␣
↪'Stede Broec', 'Texel', 'Uitgeest', 'Uithoorn', 'Velsen', 'Waterland',
                              'Weesp', 'Wormerland', 'Wijdemeren',␣
↪'Zaanstad', 'Zandvoort'],
    'Waterbedrijf Groningen': ['Eemsdelta', 'Groningen', 'Het Hogeland',␣
↪'Midden-Groningen', 'Oldambt', 'Pekela', 'Stadskanaal',
```

```
                              'Veendam', 'Westerkwartier', 'Westerwolde'],
   'Waterleidingmaatschappij Drenthe (WMD)': ['Aa en Hunze','Assen',␣
↪'Borger-Odoorn', 'Coevorden', 'Emmen', 'Hoogeveen',
                                              'Midden-Drenthe', 'Noordenveld',␣
↪'Tynaarlo', 'De Wolden'],
   'Vitens': [
       # Friesland
       'Achtkarspelen', 'Ameland', 'Dantumadiel', 'De Fryske Marren',␣
↪'Harlingen', 'Heerenveen', 'Leeuwarden', 'Noardeast-Fryslân',
       'Ooststellingwerf', 'Opsterland', 'Schiermonnikoog', 'Smallingerland',␣
↪'Súdwest-Fryslân', 'Terschelling', 'Tytsjerksteradiel',
       'Vlieland', 'Waadhoeke', 'Weststellingwerf',

       # Drenthe
       'Meppel', 'Westerveld',

       # Overijssel
       'Almelo', 'Borne', 'Dalfsen', 'Deventer', 'Dinkelland', 'Enschede',␣
↪'Haaksbergen', 'Hardenberg',
       'Hellendoorn', 'Hengelo', 'Hof van Twente', 'Kampen', 'Losser',␣
↪'Oldenzaal', 'Olst-Wijhe', 'Ommen',
       'Raalte', 'Rijssen-Holten', 'Staphorst', 'Steenwijkerland',␣
↪'Tubbergen', 'Twenterand', 'Wierden', 'Zwartewaterland', 'Zwolle',

       # Gelderland
       'Aalten', 'Apeldoorn', 'Arnhem', 'Barneveld', 'Berg en Dal',␣
↪'Berkelland', 'Beuningen', 'Bronckhorst',
       'Brummen', 'Buren', 'Culemborg', 'Doesburg', 'Doetinchem', 'Druten',␣
↪'Duiven', 'Ede', 'Elburg', 'Epe',
       'Ermelo', 'Harderwijk', 'Hattem', 'Heerde', 'Heumen', 'Lingewaard',␣
↪'Lochem', 'Maasdriel', 'Montferland',
       'Neder-Betuwe', 'Nijkerk', 'Nijmegen', 'Nunspeet', 'Oldebroek', 'Oost␣
↪Gelre', 'Oude IJsselstreek',
       'Overbetuwe', 'Putten', 'Renkum', 'Rheden', 'Rozendaal',␣
↪'Scherpenzeel', 'Tiel', 'Voorst', 'Wageningen',
       'West Betuwe', 'West Maas en Waal', 'Westervoort', 'Wijchen',␣
↪'Winterswijk', 'Zaltbommel', 'Zevenaar', 'Zutphen',

       # Flevoland
       'Almere', 'Dronten', 'Lelystad', 'Noordoostpolder', 'Urk', 'Zeewolde',

       # Utrecht
       'Amersfoort', 'Baarn', 'Bunnik', 'Bunschoten', 'De Bilt', 'De Ronde␣
↪Venen', 'Eemnes', 'Houten',
       'IJsselstein', 'Leusden', 'Lopik', 'Montfoort', 'Nieuwegein',␣
↪'Oudewater', 'Renswoude', 'Rhenen',
```

```
        'Soest', 'Stichtse Vecht', 'Utrecht', 'Utrechtse Heuvelrug',␣
↪'Veenendaal', 'Wijk bij Duurstede',
        'Woerden', 'Woudenberg', 'Zeist',

        # Noord-Holland
        'Hilversum',
    ],
    'Waternet': ['Amsterdam','Amstelveen', 'Diemen', 'Heemstede',␣
↪'Ouder-Amstel'],
    'Oasen': ['Alblasserdam', 'Alphen aan den Rijn','Bodegraven-Reeuwijk',␣
↪'Gorinchem', 'Gouda', 'Hardinxveld-Giessendam',
             'Hendrik-Ido-Ambacht', 'Kaag en Braassem', 'Krimpen aan den␣
↪IJssel', 'Krimpenerwaard', 'Leiderdorp', 'Molenlanden',
             'Nieuwkoop', 'Papendrecht', 'Ridderkerk', 'Sliedrecht',␣
↪'Vijfheerenlanden', 'Waddinxveen', 'Zoeterwoude', 'Zwijndrecht'],
    'Dunea': ["'s-Gravenhage",'Hillegom', 'Katwijk', 'Lansingerland', 'Leiden',␣
↪'Leidschendam-Voorburg', 'Lisse', 'Noordwijk',
             'Oegstgeest',  'Pijnacker-Nootdorp', 'Rijswijk', 'Teylingen',␣
↪'Voorschoten', 'Wassenaar', 'Zoetermeer', 'Zuidplas'],
    'Brabant Water': ['Alphen-Chaam', 'Altena', 'Asten', 'Baarle-Nassau',␣
↪'Bergeijk', 'Bergen op Zoom', 'Bernheze', 'Best', 'Bladel',
                      'Boekel', 'Boxtel', 'Breda', 'Cranendonck', 'Deurne',␣
↪'Dongen', 'Drimmelen', 'Eersel', 'Eindhoven', 'Etten-Leur',
                      'Geertruidenberg', 'Geldrop-Mierlo', 'Gemert-Bakel',␣
↪'Gilze en Rijen', 'Goirle', 'Haaren', 'Halderberge',
                      'Heeze-Leende', 'Helmond', "'s-Hertogenbosch", 'Heusden',␣
↪'Hilvarenbeek', 'Laarbeek', 'Land van Cuijk',
                      'Loon op Zand', 'Maashorst', 'Meierijstad',␣
↪'Sint-Michielsgestel', 'Moerdijk', 'Nuenen Gerwen en Nederwetten',
                      'Oirschot', 'Oisterwijk', 'Oosterhout', 'Oss', 'Reusel-De␣
↪Mierden', 'Roosendaal', 'Rucphen', 'Someren',
                      'Son en Breugel', 'Steenbergen', 'Tilburg',␣
↪'Valkenswaard', 'Veldhoven', 'Vught', 'Waalre', 'Waalwijk', 'Zundert'],
    'Waterleiding Maatschappij Limburg (WML)': ['Beek', 'Beekdaelen', 'Beesel',␣
↪'Bergen (L.)', 'Brunssum', 'Echt-Susteren', 'Eijsden-Margraten', 'Gennep',␣
↪'Gulpen-Wittem',
                                                  'Heerlen', 'Horst aan de Maas',␣
↪'Kerkrade', 'Landgraaf', 'Leudal', 'Maasgouw', 'Maastricht', 'Meerssen',␣
↪'Mook en Middelaar',
                                                  'Nederweert', 'Peel en Maas',␣
↪'Roerdalen', 'Roermond', 'Simpelveld', 'Sittard-Geleen', 'Stein', 'Vaals',␣
↪'Valkenburg aan de Geul',
                                                  'Venlo', 'Venray',␣
↪'Voerendaal', 'Weert'],
```

```
    'Evides': ['Albrandswaard', 'Barendrecht', 'Borsele', 'Capelle aan den␣
↪IJssel', 'Delft', 'Dordrecht', 'Goeree-Overflakkee', 'Goes', 'Hoeksche␣
↪Waard', 'Hulst',
                'Kapelle', 'Maassluis', 'Middelburg', 'Midden-Delfland',␣
↪'Nissewaard', 'Noord-Beveland', 'Reimerswaal', 'Rotterdam', 'Schiedam',
                'Schouwen-Duiveland', 'Sluis', 'Terneuzen', 'Tholen', 'Veere',␣
↪'Vlaardingen', 'Vlissingen', 'Voorne aan Zee', 'Westland', 'Woensdrecht']
}

def fill_fused_population(Pop_fused, fusion_map):
    """Aggregate municipalities into their respective water companies by␣
↪summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused['Year'] == year) & (Pop_fused['Area'] ==␣
↪company)).any():
                missing_rows.append({'Year': year, 'Area': company,␣
↪'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused = pd.concat([Pop_fused, pd.DataFrame(missing_rows)],␣
↪ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this␣
↪water company
            total_pop = Pop_fused.loc[(Pop_fused['Year'] == year) &␣
↪(Pop_fused['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused.loc[(Pop_fused['Year'] == year) & (Pop_fused['Area'] ==␣
↪company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun␣
↪in mun_list]
    Pop_fused = Pop_fused[~Pop_fused['Area'].isin(absorbed_municipalities)]

    return Pop_fused  # Return the modified Data
```

```
#Run the function
Pop_fused = fill_fused_population(Pop_fused, fusion_map)

Pop_fused['Area'].unique()
```

[3]:
```
array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
       'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
       'Oasen', 'Dunea', 'Brabant Water',
       'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

[4]:
```
def check_duplicate_municipalities(fusion_map):
    """Check if any municipality has been assigned to multiple water companies.
 ↪"""

    # Dictionary to track which companies each municipality belongs to
    municipality_to_companies = {}

    for company, municipalities in fusion_map.items():
        for municipality in municipalities:
            if municipality not in municipality_to_companies:
                municipality_to_companies[municipality] = []
            municipality_to_companies[municipality].append(company)

    # Find municipalities assigned to multiple water companies
    duplicates = {mun: comps for mun, comps in municipality_to_companies.
 ↪items() if len(comps) > 1}

    if duplicates:
        print("  The following municipalities are assigned to multiple water␣
 ↪companies:")
        for mun, comps in duplicates.items():
            print(f"  - {mun}: {', '.join(comps)}")
    else:
        print("  No duplicate assignments found!")

    return duplicates  # Return dictionary for further inspection if needed

# Run the function on your fusion_map
duplicates = check_duplicate_municipalities(fusion_map)
```

```
  No duplicate assignments found!
```

0.1.1 Repeat for low population growth

[5]:
```
Pop_fused_low = pd.read_excel("./Municipal Population Fused 2000-2050 (Low).
↪xlsx")

def fill_fused_population(Pop_fused_low, fusion_map):
    """Aggregate municipalities into their respective water companies by␣
↪summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused_low['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused_low['Year'] == year) & (Pop_fused_low['Area'] ==␣
↪company)).any():
                missing_rows.append({'Year': year, 'Area': company,␣
↪'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused_low = pd.concat([Pop_fused_low, pd.DataFrame(missing_rows)],␣
↪ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this␣
↪water company
            total_pop = Pop_fused_low.loc[(Pop_fused_low['Year'] == year) &␣
↪(Pop_fused_low['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused_low.loc[(Pop_fused_low['Year'] == year) &␣
↪(Pop_fused_low['Area'] == company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun␣
↪in mun_list]
    Pop_fused_low = Pop_fused_low[~Pop_fused_low['Area'].
↪isin(absorbed_municipalities)]

    return Pop_fused_low  # Return the modified Data

#Run the function
Pop_fused_low = fill_fused_population(Pop_fused_low, fusion_map)
```

```
Pop_fused_low['Area'].unique()
```

[5]:
```
array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
       'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
       'Oasen', 'Dunea', 'Brabant Water',
       'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

0.1.2 Repeat for high population growth

[6]:
```
Pop_fused_high = pd.read_excel("./Municipal Population Fused 2000-2050 (High).
↪xlsx")

def fill_fused_population(Pop_fused_high, fusion_map):
    """Aggregate municipalities into their respective water companies by␣
↪summing their populations."""

    # Ensure all water companies exist in the dataset
    all_years = Pop_fused_high['Year'].unique()
    missing_rows = []

    for company, municipalities in fusion_map.items():
        for year in all_years:
            if not ((Pop_fused_high['Year'] == year) & (Pop_fused_high['Area']␣
↪== company)).any():
                missing_rows.append({'Year': year, 'Area': company,␣
↪'Population': 0})  # Placeholder

    # Add missing water companies to the dataset
    Pop_fused_high = pd.concat([Pop_fused_high, pd.DataFrame(missing_rows)],␣
↪ignore_index=True)

    # Now sum populations and assign them to the water companies
    for company, municipalities in fusion_map.items():
        for year in all_years:
            # Calculate total population of the municipalities served by this␣
↪water company
            total_pop = Pop_fused_high.loc[(Pop_fused_high['Year'] == year) &␣
↪(Pop_fused_high['Area'].isin(municipalities)), 'Population'].sum()

            # Assign the total population to the water company
            Pop_fused_high.loc[(Pop_fused_high['Year'] == year) &␣
↪(Pop_fused_high['Area'] == company), 'Population'] = total_pop

    # Remove all individual municipalities that have been aggregated
    absorbed_municipalities = [mun for mun_list in fusion_map.values() for mun␣
↪in mun_list]
```

```
    Pop_fused_high = Pop_fused_high[~Pop_fused_high['Area'].
 ↪isin(absorbed_municipalities)]

    return Pop_fused_high  # Return the modified Data

#Run the function
Pop_fused_high = fill_fused_population(Pop_fused_high, fusion_map)

Pop_fused_high['Area'].unique()
```

[6]:
```
array(['Puur Water & Natuur (PWN)', 'Waterbedrijf Groningen',
       'Waterleidingmaatschappij Drenthe (WMD)', 'Vitens', 'Waternet',
       'Oasen', 'Dunea', 'Brabant Water',
       'Waterleiding Maatschappij Limburg (WML)', 'Evides'], dtype=object)
```

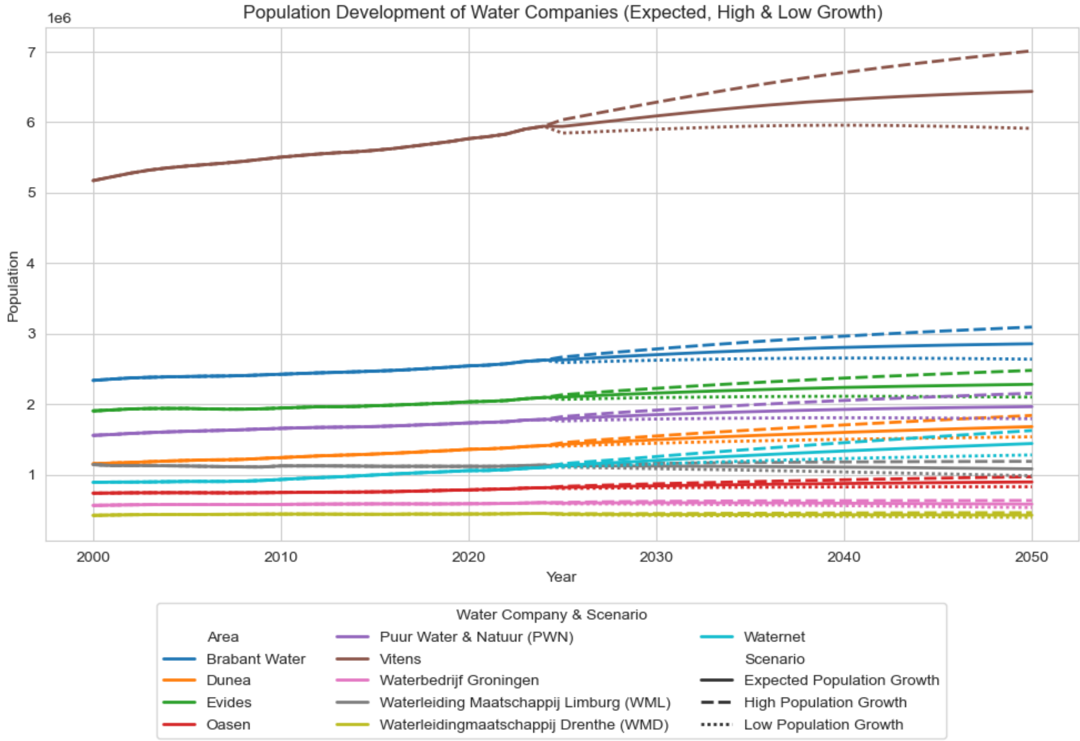
[7]:
```
# Fuse together the three datasets
Pop_fused['High Population Growth'] = Pop_fused_high['Population']

Pop_fused['Low Population Growth'] = Pop_fused_low['Population']

Pop_fused = Pop_fused.rename(columns={"Population": "Expected Population␣
 ↪Growth"})

Pop_fused = Pop_fused.sort_values(by=["Area", "Year"]).reset_index(drop=True)

Pop_fused.head()
```

[7]:
```
            Area  Year  Expected Population Growth  High Population Growth  \
0  Brabant Water  2000                   2335009.0               2335009.0
1  Brabant Water  2001                   2353660.0               2353660.0
2  Brabant Water  2002                   2369575.0               2369575.0
3  Brabant Water  2003                   2378641.0               2378641.0
4  Brabant Water  2004                   2385579.0               2385579.0

   Low Population Growth
0              2335009.0
1              2353660.0
2              2369575.0
3              2378641.0
4              2385579.0
```

0.1.3 Create a figure for population growth & save to Excel

[8]:
```
# Set Seaborn style
sns.set_style("whitegrid")

# Convert DataFrame from wide format to long format for easier plotting
```

```
Pop_long = Pop_fused.melt(id_vars=["Year", "Area"],
                          value_vars=["Expected Population Growth", "High␣
 ↪Population Growth", "Low Population Growth"],
                          var_name="Scenario",
                          value_name="No. of Citizens")

# Add this filter to focus on specific water companies
selected_companies = ['Puur Water & Natuur (PWN)',
                      'Waterbedrijf Groningen',
                      'Waterleidingmaatschappij Drenthe (WMD)',
                      'Vitens',
                      'Waternet',
                      'Oasen',
                      'Dunea',
                      'Brabant Water',
                      'Waterleiding Maatschappij Limburg (WML)',
                      'Evides'
                     ]  # Modify this list

Pop_filtered = Pop_long[Pop_long["Area"].isin(selected_companies)]

# Create the plot
plt.figure(figsize=(12, 6))
sns.lineplot(data=Pop_filtered, x="Year", y="No. of Citizens", hue="Area",␣
 ↪style="Scenario", linewidth=2)

# Labels and title
plt.xlabel("Year")
plt.ylabel("Population")
plt.title("Population Development of Water Companies (Expected, High & Low␣
 ↪Growth)")

# Improve legend readability
plt.legend(title="Water Company & Scenario",  bbox_to_anchor=(0.1, -0.4),␣
 ↪loc="lower left", ncols = 3)

#  Save the plot
plt.savefig("population_water_companies_lowlegend.png", dpi=300,␣
 ↪bbox_inches="tight")

# Show the plot
plt.show()
```



[9]:
```
#Save the scenarios to Excel sheets
Pop_BW = Pop_fused.loc[Pop_fused['Area']== 'Brabant Water']
Pop_D = Pop_fused.loc[Pop_fused['Area']== 'Dunea']
Pop_E = Pop_fused.loc[Pop_fused['Area']== 'Evides']
Pop_O = Pop_fused.loc[Pop_fused['Area']== 'Oasen']
Pop_PWN = Pop_fused.loc[Pop_fused['Area']== 'Puur Water & Natuur (PWN)']
Pop_V = Pop_fused.loc[Pop_fused['Area']== 'Vitens']
Pop_WBGR = Pop_fused.loc[Pop_fused['Area']== 'Waterbedrijf Groningen']
Pop_WML = Pop_fused.loc[Pop_fused['Area']== 'Waterleiding Maatschappij Limburg␣
↪(WML)']
Pop_WMD = Pop_fused.loc[Pop_fused['Area']== 'Waterleidingmaatschappij Drenthe␣
↪(WMD)']
Pop_W = Pop_fused.loc[Pop_fused['Area']== 'Waternet']


with pd.ExcelWriter('Water Company Population 2000-2050 (Growth Scenarios).
↪xlsx') as writer:
    Pop_BW.to_excel(writer, sheet_name='Brabant Water')
    Pop_D.to_excel(writer, sheet_name='Dunea')
    Pop_E.to_excel(writer, sheet_name='Evides')
    Pop_O.to_excel(writer, sheet_name='Oasen')
    Pop_PWN.to_excel(writer, sheet_name='PWN')
```

```
    Pop_V.to_excel(writer, sheet_name='Vitens')
    Pop_WBGR.to_excel(writer, sheet_name='Wb Gr')
    Pop_WML.to_excel(writer, sheet_name='WML')
    Pop_WMD.to_excel(writer, sheet_name='WMD')
    Pop_W.to_excel(writer, sheet_name='Waternet')
```