Deltares

Enabling Delta Life

DAM Engine

Technical Design

Deltares

DAM Engine

Technical Design

1210702-000

Deltares

Deltares

Title
DAM Engine

Client	Project	Reference	Pages
Deltares - Geo engineering DKS	1210702-000	1210702-000-GEO-0004	23

Classification
-

Keywords
Dike, safety assessment, design, software, macro stability, piping

Summary
This document contains the technical design for DAM Engine, an application that computes the strength of a complete dikering with respect to several failure mechnanisms, such as macro stability and piping.

Samenvatting
Dit document bevat het technisch ontwerp voor DAM Engine, een User Interface applicatie die een gebruiker in staat stelt om voor een dijktraject berekeningen uit te voeren voor verschillende faalmechanismen, waaronder macrostabiliteit en piping.

References
Refer to chapter 8.

Version	Date	Author	Initials	Review	Initials	Approval	Initials
0.2	Mar 2017	Tom The		John Bokma		Maya Sule	

Status
draft
This is a draft report, intended for discussion purposes only. No part of this report may be relied upon by either principals or third parties.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Purpose and scope of this document

This document contains the technical design for the DAM Engine, a computational engine for the automated calculation of the strength of dikes. DAM was developed by Deltares with and for STOWA for all water authorities. This document describes the full intended architecture of the DAM Engine. What will actually will be implemented depends on the requirements of the clients using this DAM Engine. If some functionality is not (yet) needed, then that part does not need to be implemented.

1.2 Other system documents

The full documentation on the program comprises the following documents.

Title	Content
DAM Engine- Architecture Overall (The, 2017a)	Description of overall architecture of the DAM Engine and its components.
DAM Engine- Functional Design (Zwan, 2017)	Description of the requirements and functional design.
DAM Engine- Technical Design (this document) (The, 2017b)	Description of the implementation of the technical design of DAM Engine.
DAM Engine- Technical documentation (Doxygen, 2017)	Description of the arguments and usage of different software components, generated from in-line comment with Doxygen.
DAM Engine- Test Plan (Trompille, 2017a)	Description of the different regression and acceptation tests, including target values.
DAM Engine- Test Report (Trompille, 2017b)	Description of the test results (benchmarks and test scripts).

Table 1.1: *DAM Engine system documents.*

1.3 Document revisions

1.3.1 Revision 0.1

First concept of the document.

1.3.2 Revision 0.2

Adapted based on reviews of this document by Jan Noort and André Grijze.

2 System Architecture

2.1 DAM components

DAM Engine is part of the whole DAM system that contains several components. Please see Figure 2.1 for an overview of the components of DAM. In (The, 2017a) a description of the overall architecture of the DAM system can be found.

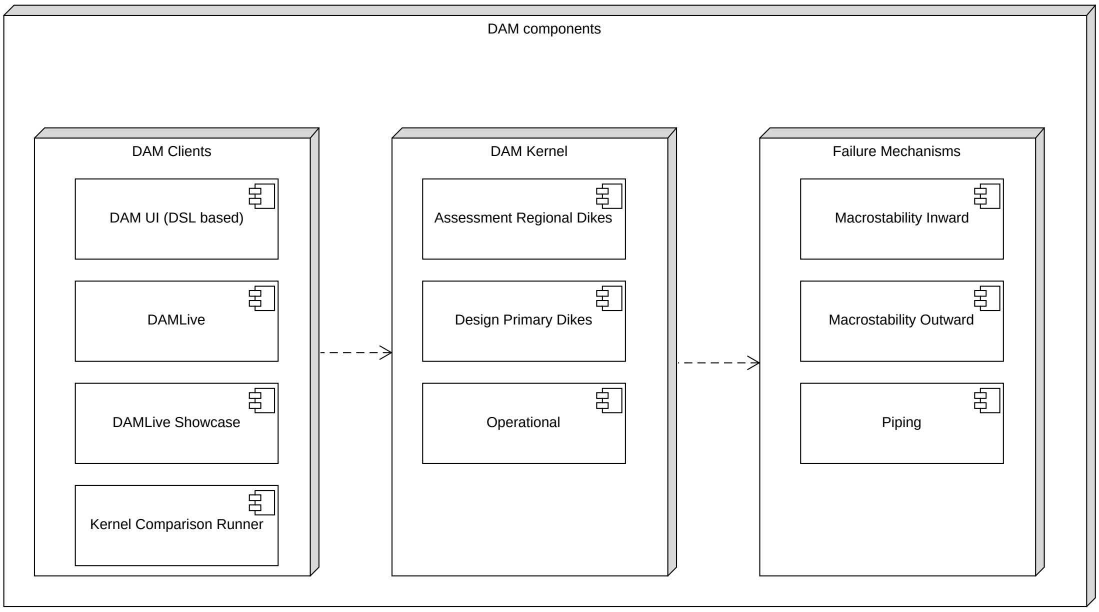

Figure 2.1: *DAM Engine and its components.*

The arrows illustrate the dependencies of the components.

2.2 DAM Engine components

The DAM Engine itself also consists of several modules. These can be seen in see Figure 2.2

All of the submodules inside the Main Modules are completely independent. All of the submodules inside the Supporting Modules are also independent. But all these submodules can be used by each of the main modules. The arrows show the allowed dependencies.

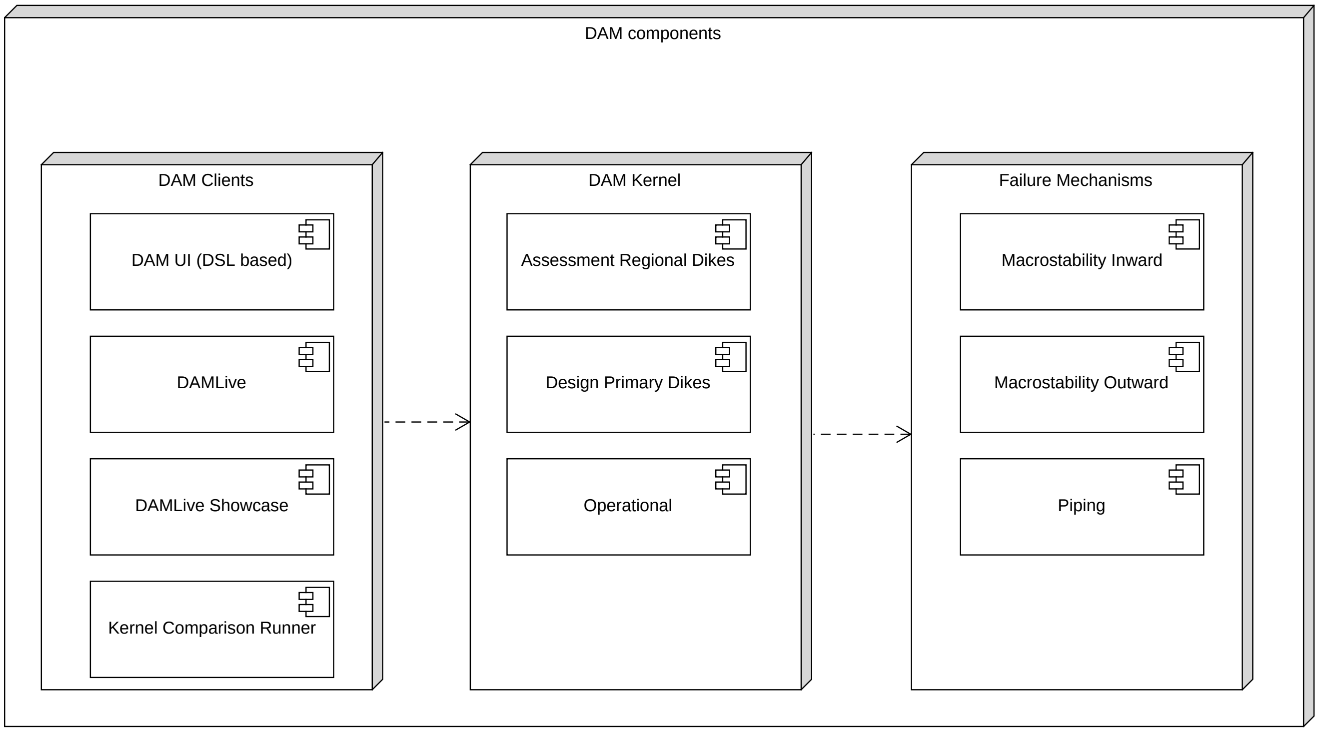

Figure 2.2: *DAM Engine and its components.*

2.3 DAM Engine sequence diagrams

In this section the sequence diagrams, showing the use of the submodules are shown.

2.3.1 Assessment

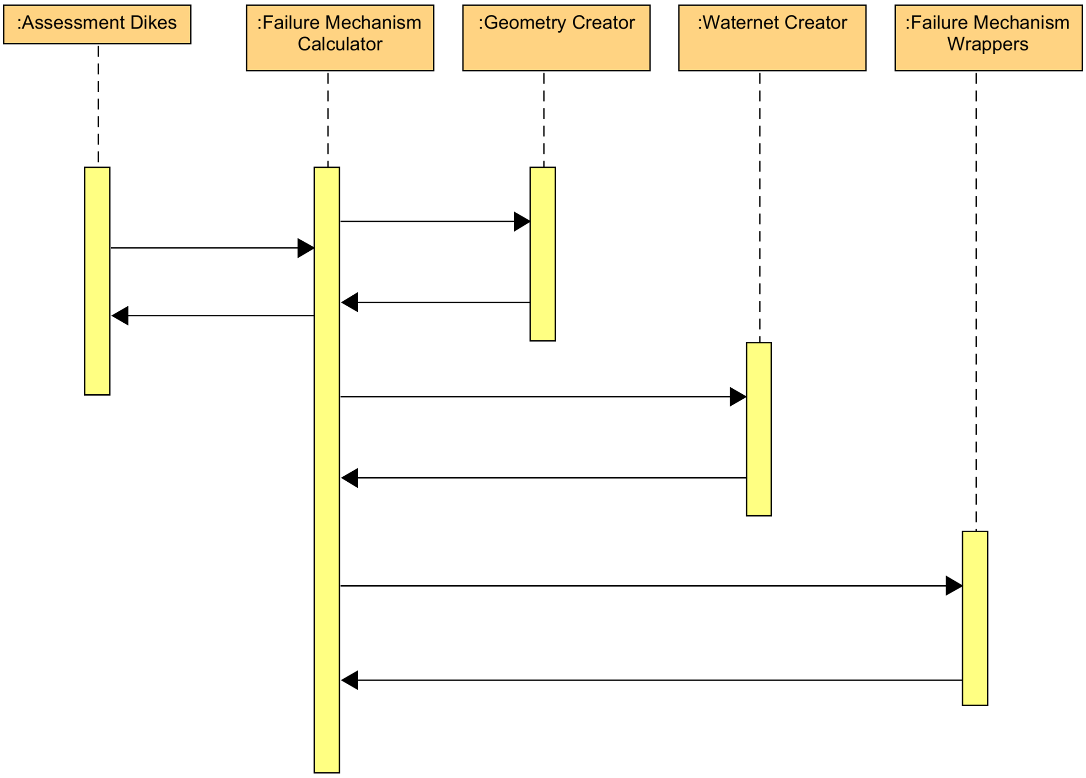

Figure 2.3: *DAM Engine Assessment sequence diagram.*

2.3.2 Assessment Regional

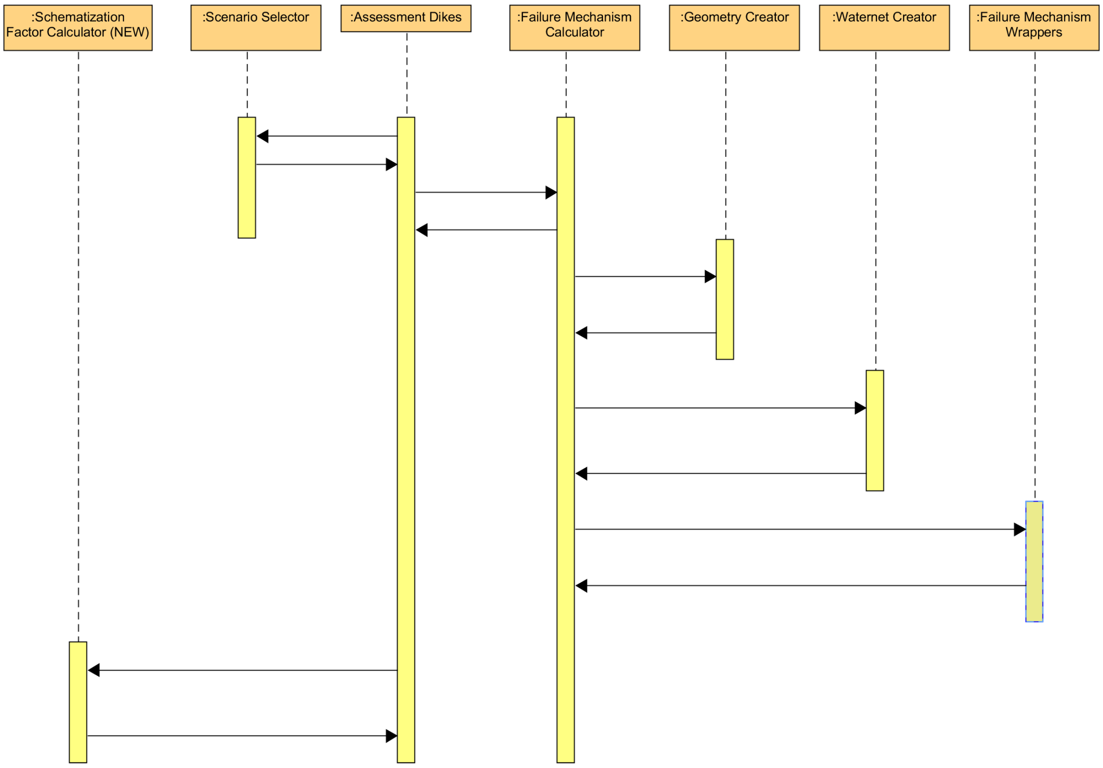

Figure 2.4: *DAM Engine Regional assessment sequence diagram.*

2.3.3 Design

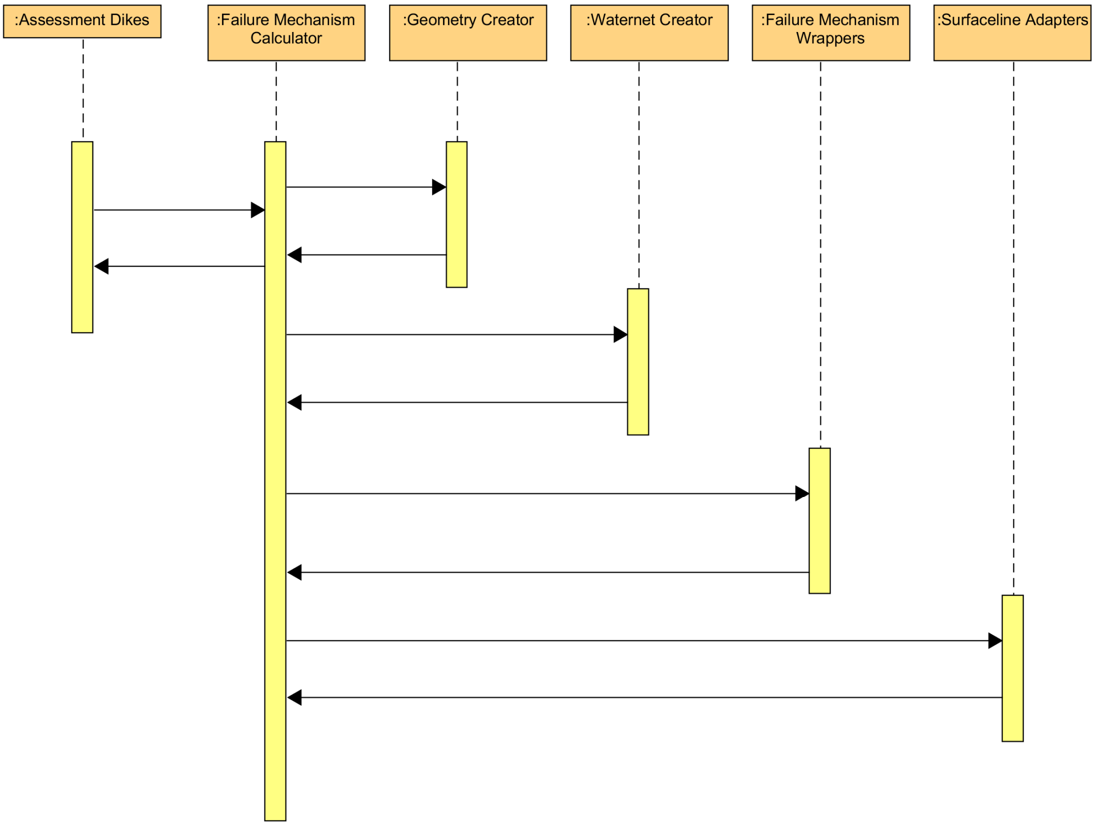

Figure 2.5: *DAM Engine Design sequence diagram.*

3 Architectural Choices

3.1 Design principles

- No circular references between objects. When it is really unavoidable, then do it through a generic interface (e.g. IParentObject)
- The calculation will support parallellization. So do not use global variables and avoid using statics.
- Failure mechanisms will be connected through wrapper classes, which will share a common IFailureMechanism interface
- Surfaceline adapter classes will share a common ISurfacelineAdapter interface
- The DAM Engine must provide progress information of the calculation, so clients of the DAM Engine can show a progressbar
- The DAM Engine must provide the possiblity to abort a calculation within a reasonable timespan.
- There should be no User Interface elements shown anytime during the calculation.

3.2 Programming environment

The DAM Engine will be developed in C# with the .NET 4.5 framework. The development environment will be Visual Studio 2015.

3.3 Error handling

Errors are handled through the standard exception handling of the .NET framework. Error messages must contain as much information as possible, so a user can trace back the error to the input data.

3.4 External libraries and components

DAM Engine uses third-party libraries and components. Only open sources and free components, that are free to redistribute are allowed to be used.

Furthermore DAM Engine uses the Delta Shell Light (DSL) library, that is developed by Deltares.

3.5 DSL

The DAM Engine only uses part (the non-UI modules) of the DSL library

- DSL-Core
- DSL-Probabilistic
- DSL-Geo

3.6 Other libraries

Other libraries that are used bij the DAM Engine are:

- Math.Net: mathematical library

4 Data Model

4.1 Main Data Model

The main data model can be seen in see Figure 4.1

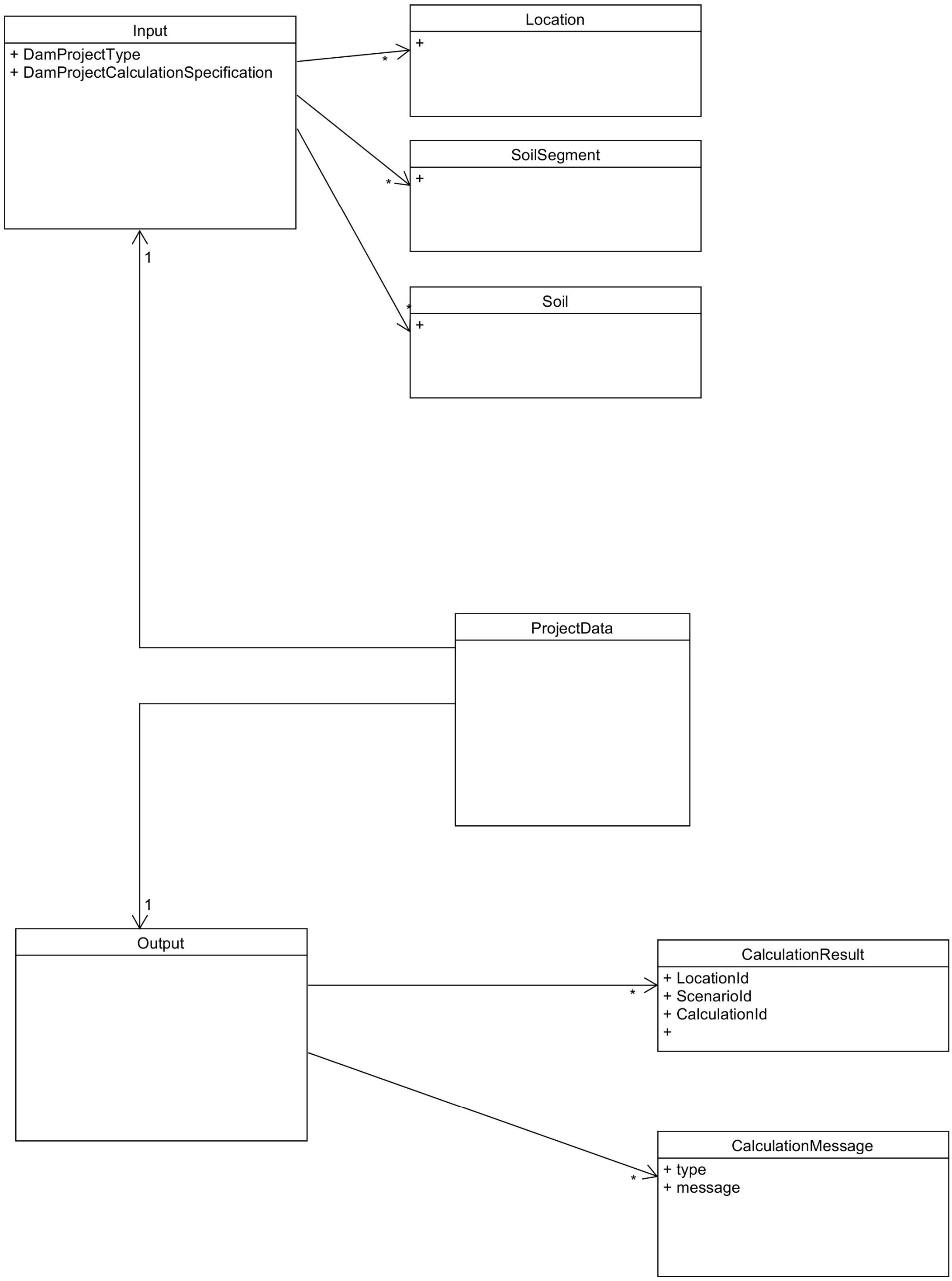

Figure 4.1: *DAM Engine main data model.*

4.2 Location

The data model of the Location class can be seen in see Figure 4.2

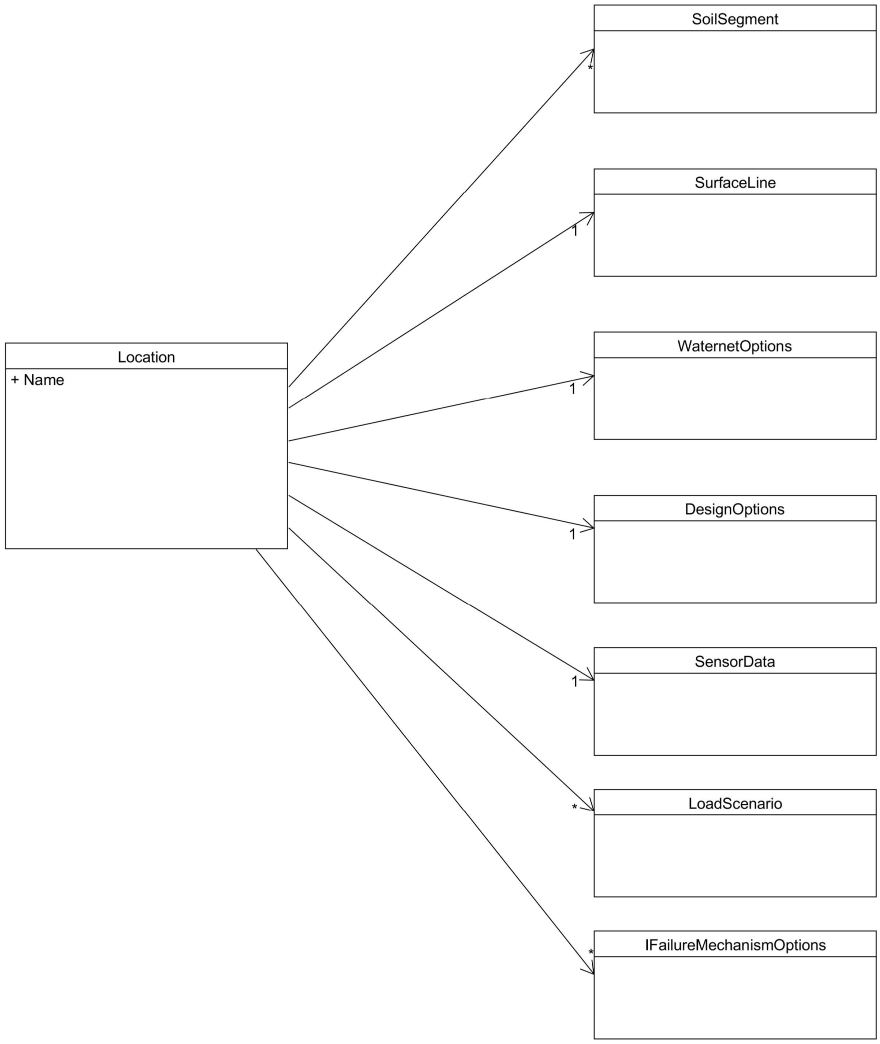

Figure 4.2: *DAM Engine Location object.*

5 Data Description

5.1 Type enumerations

5.1.1 MainMechanismType

- Stability
- Piping

5.2 Main Data Model

5.2.1 Input

DamProjectType

- Assessment
- Design
- Operational

DamProjectCalculationSpecification

This class specifies which failuremechanism is to be calculated and it also contains the specific options for the selelected mechanism (e.g. which calculation model)

Locations

This is a collection of locations, with each location containing the location specific data.

Soil Segments

This is a collection of soil segments, with each segment containing the subsoil data for a specific failure mechanism.

Soils

This is a collection of soils, with each soil containing the soil parameters needed for the calculation of all failure mechanisms.

5.2.2 Output

CalculationResults

A calculation result holds the result for a specific location, a specific failure mechanism, and a specific subsoil scenario of a specific segment defined in the location data.

CalculationMessages

These are all the message that are generated by the calculation. A message must containt as much information as possible to trace back the information tho the input data (e.g. a specific location, a specific failure mechanism, and a specific subsoil scenario of a specific segment defined in the location data).

5.3 Location

SoilSegment

A soil segment contains the subsoil data for a specific failure mechanism

SurfaceLine

A surfaceline is describes the dike profile in a specific location. In the Design calculation it can also be the new dike profile, which can meet design creteria in a specific load scenario.

WaternetOptions

The options that support the creation of a waternet in a specific location.

DesignOptions

The options that will be used in the Design calculation (e.g. how to design a shoulder when needed).

SensorData

The sensor data can be used to define a waternet based on live sensor data. This sensor data holds information about ID and location of the sensor. The actaul sensor readings are defined as timeseries readings for each sensor in each location.

LoadScenario

Used for Design calculation. A load scenario contains the following items:

- Riverlevel low
- Riverlevel high
- Dike table height
- Required safety factor for each specified failure mechanism
- Uplift criterium for each specified failure mechanism
- Waternet options for each specified failure mechanism

IFailureMechanismOptions

Specific options for each location for each failure mechanism.

6 Module Description

6.1 DAM Engine main modules

6.1.1 Assessment Regional Dikes

This module performs an assessment for regional dikes.

Regional assessment calculation

This is the main submodule of the regional assessement. This submodule contains the main loop of the calculation.

Regional scenario selector

This submodule generates all the scenarios that have to be evaluated for a specific location. The scenarios are selected based on the local conditions.

Regional schematization factor calculator

This submodule calculates the schematization factor in a location based on all results of all scenarios in a location.

6.1.2 Assessment Primary Dikes

This module performs an assessment for primary dikes.

Primary assessment calculation

This is the main submodule of the primary assessement. This submodule contains the main loop of the calculation.

6.1.3 Design Primary Dikes

This module performs an design calculation for primary dikes.

Primary design calculation

This is the main submodule of the primary design calculation. This submodule contains the main loop of the calculation.

6.1.4 NWO Calculation

This module performs an NWO (Niet Waterkerende Objecten) calculation for primary dikes.

Primary NWO calculation

This is the main submodule of the NWO calculation. This submodule contains the main loop of the calculation.

6.2 DAM Engine supporting modules

6.2.1 Failure mechanism wrappers

These are calculation wrappers for each kernel to be implemented for each failure mechanism that is supported by the DAM Engine. Each wrapper should implement the IFailureMechanism interface.
Example: Lets say that for the failure mechanism piping we have 3 kernels: Bligh, Sellmeijer and VNK. Then for each of these kernels a calculation wrapper has to be written.
Another example: D-Geo Stability kernel has the ability to calculate the failure mechanism macrostability inwards en the failure mechanism macrostability outwards. In this case 2 wrappers (one for each failure mechanism) are needed for this single kernel.

Macrostability inwards

Calculation wrapper for Macrostability inward. Note that (as already mentioned above) for each specific kernel implementation for a failure mechanism, a separate wrapper has to be build (e.g. Slope/W and D-Geo Stability)

Macrostability outwards

Calculation wrapper for Macrostability outward.

Piping

Calculation wrapper for Piping.

6.2.2 Surfaceline adapters

A collection of surfaceline adapters to support the design calculation. Each adapter should adhere to the ISurfaceLineAdapter interface.

Surfaceline Adapter Height

Adapts the surfaceline by adding extra height to the dike crest.

Surfaceline Adapter Slope

Adapts the surfaceline by changing the slope of the dike on the inside.

Surfaceline Adapter Shoulder

Adapts the surfaceline by adding a shoulder or enlarging the shouldeer on the inside of the dike.

Surfaceline Adapter NWO

Adapts the surfaceline by adding a NWO on a specifi place in the surfaceline.

6.2.3 Calculation Runner

Failure mechanism Calculation Runner

This submodule calculates a specific failure mechanism by calling the IFailureMechanism interface of the wrapper implementation.

Design Calculation Runner

This submodule performs a design calculation for a specific failure mechanism by calling the IFailureMechanism interface and several surfaceline adapters through their ISurfacelineAdaptert interface.

Operational Calculation Runner

This submodule can perform a calculation based on sensor readings (as time-series). The load on the systems (the waternet) will be based on those sensor readings. This can be used in operational systems like DamLive.

Probabilistic Calculation Runner

This submodule performs a probabilistic calculation for a specific location and failure mechanism. The outcome is a failure probability for that location and failure mechanism.

6.2.4 General submodules

Geometry creator

This submodule combines a surfaceline with a subsoil scenario. The output is a gemetry that can be used by the failure mechanisms to perform a calculation.

Waternet creator

This submodule determines the waternet that will be used by the failure mechanisms. At first only the DAM implementation will be used as a waternet creator. Later on new implementations can be made. E.g. specific for each failure mechanism, or an implementation based on a numerical model like DgFlow.

6.2.5 Scripting engine

To enable advanced users to experiment with how the DAM Engine works a Python scripting engine is implemented as a submodule. The scripting engine has acces to the data model and the submodules through well defined interfaces.

7 Programing Interface

This is the definition of the programming interface. The only way to communicate to the DAM Engine is through this interface.

TO DO: Add interface description

8 Literature

Doxygen, 2017. *DAM Engine - Technical documentation, Generated by Doxygen 1.8.10*. Tech. rep., Deltares.

The, T., 2017a. *DAM Architecture Overall*. Tech. Rep. 1210702-000-GEO-0005, version 0.1, jan. 2017, concept, Deltares.

The, T., 2017b. *DAM Engine - Technical Design*. Tech. Rep. 1210702-000-GEO-0004, version 0.2, mar. 2017, concept, Deltares.

Trompille, V., 2017a. *DAM Engine - Test Plan*. Tech. Rep. 1210702-000-GEO-0006, version 0.1, jan. 2017, concept, Deltares.

Trompille, V., 2017b. *DAM Engine - Test Report*. Tech. Rep. 1210702-000-GEO-0007, version 0.1, jan. 2017, concept, Deltares.

Zwan, I. v., 2017. *DAM Engine - Functional Design*. Tech. Rep. 1210702-000-GEO-0003, version 0.1, jan. 2017, concept, Deltares.