Deltares

Enabling Delta Life

DAM

Architecture Overall

Deltares

DAM

Architecture Overall

1210702-000

Deltares

Deltares

Title
DAM

Client	Project	Reference	Pages
Deltares - Geo engineering DKS	1210702-000	1210702-000-GEO-0005	15

Classification
-

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

Summary
This document contains a description of the overall architecture for DAM, an application that computes the strength of a complete dikering with respect to several failure mechanisms, such as macro stability and piping.

Samenvatting
Dit document bevat een beschrijving van de totale architectuur van DAM, 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 6.

Version	Date	Author	Initials	Review	Initials	Approval	Initials
0.1	Jan 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

1 Introduction **1**
- 1.1 Purpose and scope of this document . . . 1
- 1.2 Other system documents . . . 1

2 Technical requirements **3**
- 2.1 Platform . . . 3
- 2.2 System requirements . . . 3
- 2.3 Regional settings . . . 3
- 2.4 Additional requirements . . . 3

3 DAM and its components **5**
- 3.1 DAM clients . . . 5
- 3.2 DAM Engine . . . 7
- 3.3 Failure mechanisms . . . 7

4 Architectural Choices **9**
- 4.1 Module dependencies . . . 9
- 4.2 External libraries and components . . . 9
- 4.3 DAM UI (DSL) . . . 9
- 4.4 DAM Live . . . 9
- 4.5 DAM Live Showcase . . . 10
- 4.6 DAM Kernel Comparison Runner . . . 10
- 4.7 DAM Engine . . . 10
- 4.8 Failure mechanisms . . . 10

5 Version control **11**
- 5.1 DAM repository main layout . . . 11
- 5.2 DAM repository clients layout . . . 12
- 5.3 DAM repository clients library layout . . . 12
- 5.4 DAM repository DAM Engine . . . 13
- 5.5 DAM repository full layout . . . 13

6 Literature **15**

List of Figures

3.1 DAM and its components. 5
3.2 DAM User Interface. 6
3.3 DAM Live Showcase User Interface. 6
3.4 DAM Live sensor. 7

5.1 DAM SVN main layout. 11
5.2 DAM SVN clients layout. 12
5.3 DAM SVN clients library layout. 12
5.4 DAM Engine layout. 13
5.5 DAM SVN full layout. 14

List of Tables

1.1 DAM system documents. 1

1 Introduction

1.1 Purpose and scope of this document

This document contains the overall architecture of DAM, a software package for the automated calculation of the strength of dikes, and all of its components. DAM was developed by Deltares with and for STOWA for all water authorities.

1.2 Other system documents

The full documentation on the program comprises the following documents.

Title	Content
DAM- Architecture Overall (The, 2017a)	Description of overall architecture of DAM and its components.
DAM Engine - Technical Design (The, 2017b)	Description of the implementation of the architecture and technical design of the DAM Engine.
DAM Engine - Technical documentation (Doxygen, 2017a)	Description of the arguments and usage of different software components of the DAM Engine, generated from in-line comment with Doxygen.
DAM Engine - Test Plan (Trompille, 2017a)	Description of the different regression and acceptation tests of the DAM Engine, including target values.
DAM Engine - Test Report (Trompille, 2017b)	Description of the test results (benchmarks and test scripts) of the DAM Engine.
DAM UI - Functional Design (Zwan, 2017)	Description of the requirements and functional design of the DAM User Interface.
DAM UI - Technical Design (The, 2017c)	Description of the implementation of the architecture and technical design of the DAM User Interface.
DAM UI - Technical documentation (Doxygen, 2017b)	Description of the arguments and usage of different software components of the DAM User Interface, generated from in-line comment with Doxygen.
DAM UI - Test Plan (Trompille, 2017c)	Description of the different regression and acceptation tests, including target values for the DAM User Interface.
DAM UI - Test Report (Trompille, 2017d)	Description of the test results (benchmarks and test scripts) of the DAM User Interface.
DAM UI - User Manual (Erik Vastenburg, 2013)	Description of the different functionalites available in the *User Interface* and background information.

Table 1.1: *DAM system documents.*

2 Technical requirements

2.1 Platform

Operating system: Windows 7- 32 bits or Windows 7- 64 bits, Dutch or UK version.
Read/write permission on user selectable folder for saving and calculating projects.
Administrator rights are required for installation of the application.
Required: Microsoft DotNet framework, version 4.5.

2.2 System requirements

Processor: Intel Core i5 or better.
Clockspeed processor: 2.4 GHz or better.
Memory (RAM): 4 GB or more.
Free harddisk space: 20 GB or more.
Monitor: 22 inch monitor, resolution 1920x1080.

2.3 Regional settings

The user can select between 2 languages in the application:

- Dutch (NL)
- English (UK)

The application should behave correctly with regional settings set to:

- UK
- NL

2.4 Additional requirements

- DAM must support the use of multiple processor cores, when they are available, to speed up the calculation.
- DAM is a standalone application, but must support the use of network storage.

3 DAM and its components

DAM contains several components. Please see Figure 3.1 for an overview of the components.

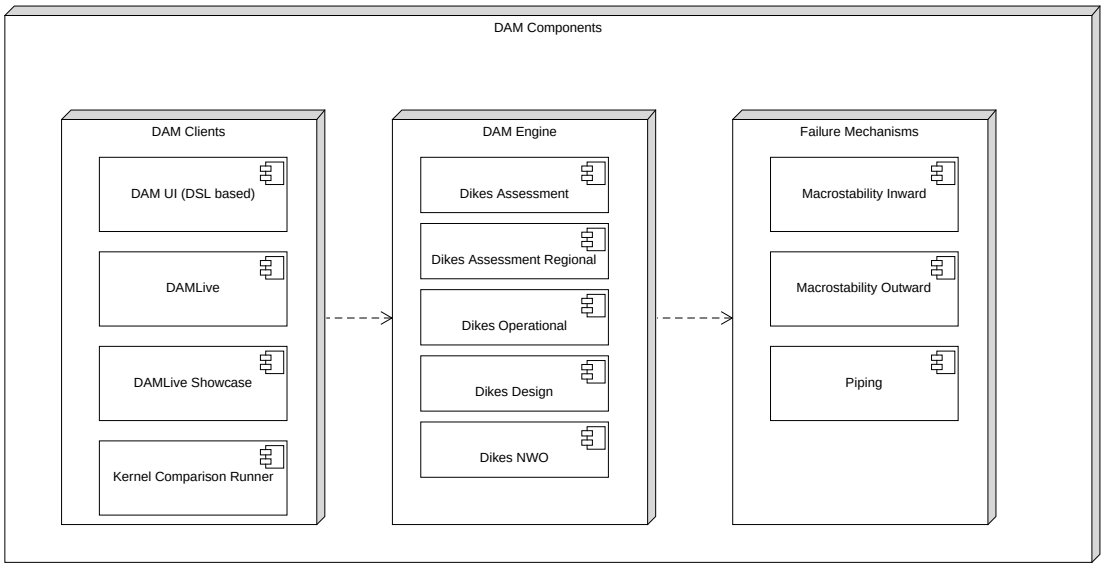

Figure 3.1: *DAM and its components.*

The arrows illustrate the dependencies of the components. A dependency also implies communication and interaction between the respective components. The exact definition of how to communicate and interact between these components is described in the technical design of each component (e.g for the DAM Engine (The, 2017b)).
In the following sections the components are described.

3.1 DAM clients

DAM clients are the modules that mostly interact with the user or sometimes with another system. These can be full graphical user interfaces (like DAM UI as shown in Figure 3.2 and DamLive Showcase as shown in Figure 3.3 and Figure 3.4), commandline parameter tools (like KernelComparisonRunner.exe) or a module that can be used by other systems (like DAMLive, that is to be used as a module in a FEWS system).

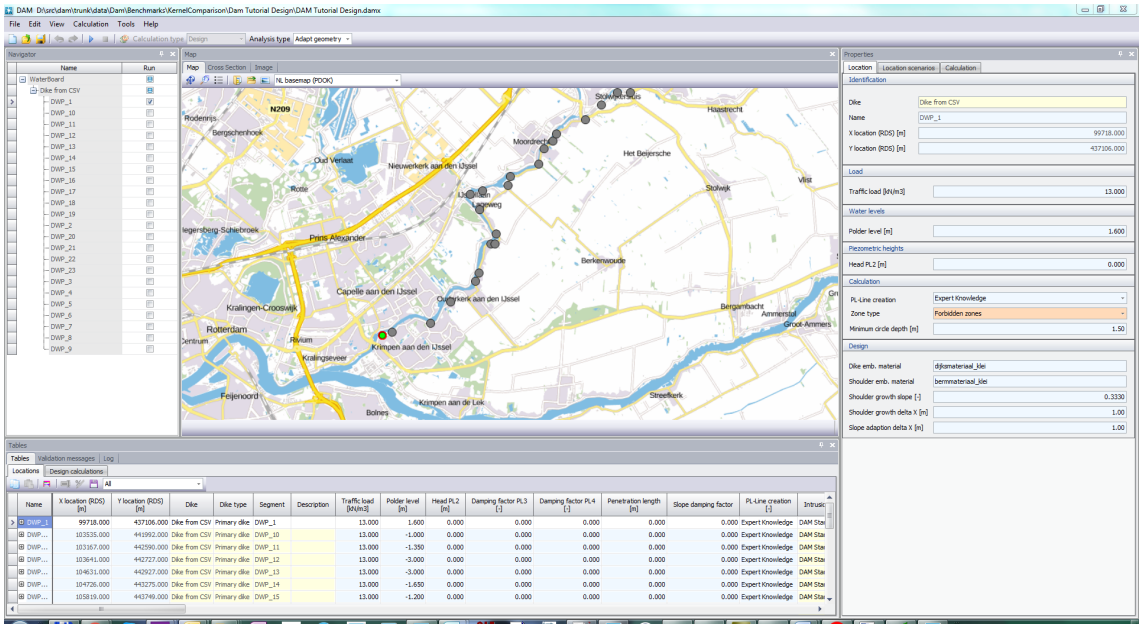

Figure 3.2: *DAM User Interface.*

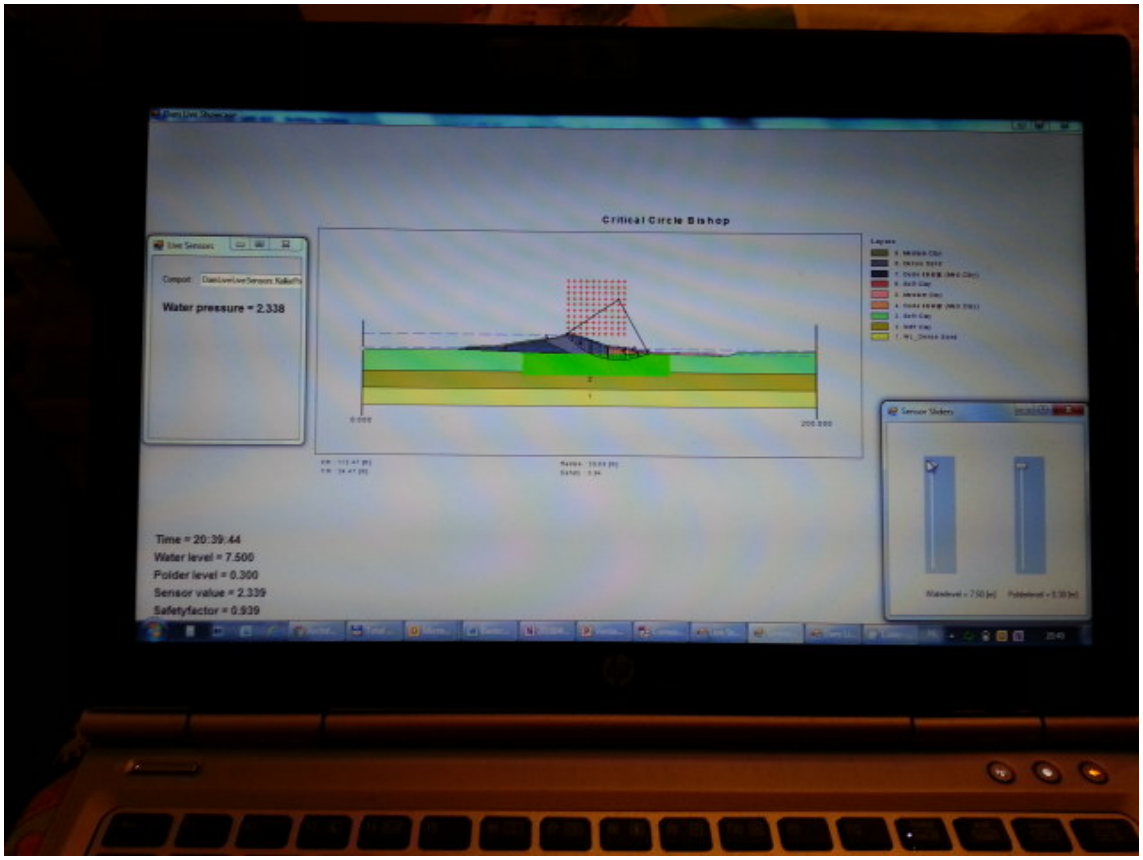

Figure 3.3: *DAM Live Showcase User Interface.*

Figure 3.4: *DAM Live sensor.*

These clients interact directly with the DAM Engine (see next section).

3.2 DAM Engine

The DAM Engine is the engine for the DAM computation. It contains several submodules, depending on which computation has to be made. As can be seen in Figure 3.1 the submodules are

- Assessment regional dikes
- Design primary dikes
- Operational module (for realtime calculations)

Depending on the client, 1 or more of these submodules can be addressed by the client. The DAM Engine has no knowledge of the clients that use the Engine and there will be no code dependencies from the Engine to the clients.

3.3 Failure mechanisms

The DAM Engine uses external failure mechanisms. These are completely independent, and have no knowledge of the DAM Engine. So there will be no code dependencies from the failure mechanisms to the DAM Engine. Furthermore development, maintenance and support of these failure mechanisms are not in the scope of the DAM system.

4 Architectural Choices

4.1 Module dependencies

As can be seen in Figure 3.1 the arrows pointing between the main parts of the system are only 1 way. This means that e.g. the DAM clients may address the DAM Engine, but the DAM Engine may not address the DAM clients.

4.2 External libraries and components

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

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

Due to choices that have been made in the past, the UI modules of DSL use DevExpress, which is a commercial library. Free redistribution of the DevExpress is allowed by the development license that is used by Deltares.

In the future the dependency on DevExpress should be removed, e.g. by using other, open source, UI libraries.

In the next sections the libraries that are used by the components are summarized.

4.3 DAM UI (DSL)

This client uses the full Delta Shell Light (DSL) library

- DSL-Core: standard library with general common functionality
- DSL-Probabilistic: probabilistic functionality
- DSL-Geographic: GIS functionality
- DSL-Geo: geotechnics functionality
- DSL-GeoIO: geotechnics import and database functionality
- DSL-FormsStandard: standard UI functionality
- DSL-FormsMaps: extends FormsStandard with GIS functionality
- DSL-FormsGeo: extends FormsStandard with geotechnical functionality

Other libraries that are used are

- Dot Spatial: GIS library
- Commandline Parser: library for parsing commandline options
- Lumenworks: CSV import library
- SQLite: SQLite database access library
- Firebird: Firebird database access library

4.4 DAM Live

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

- DSL-Core
- DSL-Geographic

4.5 DAM Live Showcase

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

- DSL-Core

4.6 DAM Kernel Comparison Runner

DAM Kernel Comparison Runner only uses part (non-UI modules) of the DSL library

- DSL-Core

4.7 DAM Engine

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

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

Other libraries that are used are

- Math.Net: mathematical library

4.8 Failure mechanisms

The failure mechanisms are completely independent, and will incorportate their own libraries. The only restriction that is imposed on the failure mechanisms is that, if they share the same libraries with DAM, the libraries should have the same version. The following libraries are know to be used by some failure mechanisms:

- DSL-Core
- DSL-Probabilistic
- DSL-Geographic
- DSL-Geo
- DSL-GeoIO
- Math.Net

5 Version control

As version control system Subversion with the Tortoise client will be used. The layout of the SVN repository will reflect the components of DAM as shown in Figure 3.1.

The failure mechanisms are not part of DAM itself and thus not of the DAM repository. Instead, the failure mechanisms will be stored in their own repositories. DAM will refer to the failure mechanisms as external libraries.

5.1 DAM repository main layout

In Figure 5.1 the main layout is shown.

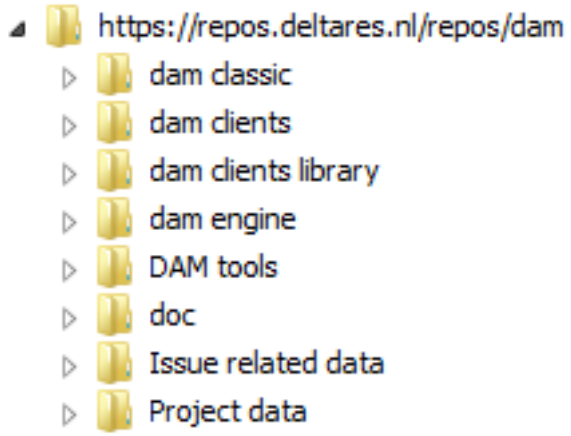

Figure 5.1: *DAM SVN main layout.*

The 4 main parts are

- dam clients - the client applications that use the DAM Engine.
- dam clients library - shared libraries by the DAM clients.
- dam engine - the computational engine of DAM.
- doc - general documentation for DAM (like this document).

The maps "'dam clients library"','dam engine"', and "'doc"' have their own trunk/branches/tags layout.

The map "'DAM tools"' is for storing independent tools that are not part of DAM itself, but support the work processes of DAM (e.g. Dam Edit Design).

The map "'dam classic"' is used by the previous implementation of DAM. The map "'project data"' is for archiving DAM project data. The map "'Issue related data"' is for storing issue related data. DAM uses Jira as issuetracking system, which supports attachments, but DAM attachments can be very large, so it is better to save the attachments in this location. The attachments are stored in a map with the name of the issue number (e.g. MWDAM-982)

5.2 DAM repository clients layout

Each DAM client has its own entry in the clients map. In Figure 5.2 the layout of the DAM clients map is shown.

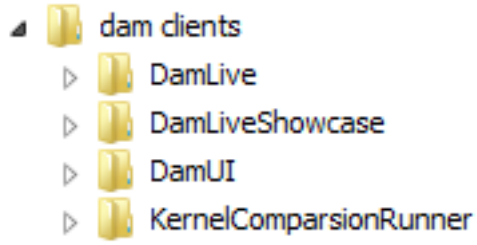

Figure 5.2: *DAM SVN clients layout.*

Each of the "'dam clients"' has its own trunk/branches/tags layout.

The currently known applications are

- Dam UI - the current DAM desktop application.
- DamLive - the current runner for the Fews operational system.
- DamLiveShowcase - a demo application to show DamLive with live sensors.
- KernelComparisonRunner - a commandline utility for comparing the results of different macrostability kernels.

5.3 DAM repository clients library layout

In Figure 5.3 the layout of the DAMUI clients library map is shown.

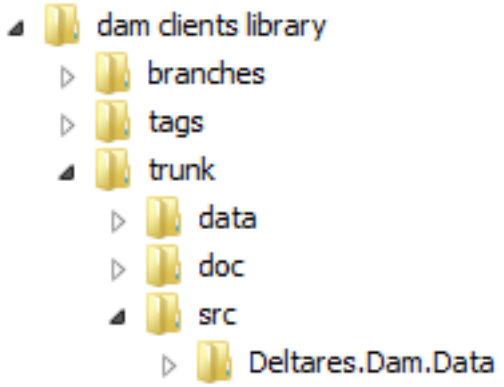

Figure 5.3: *DAM SVN clients library layout.*

At this moment only one project is foreseen to be put in the "'dam clients library"' src map is:

- Deltares.Dam.Data: this map contains the business layer.

5.4 DAM repository DAM Engine

In Figure 5.4 the layout of the DAM Engine map is shown.

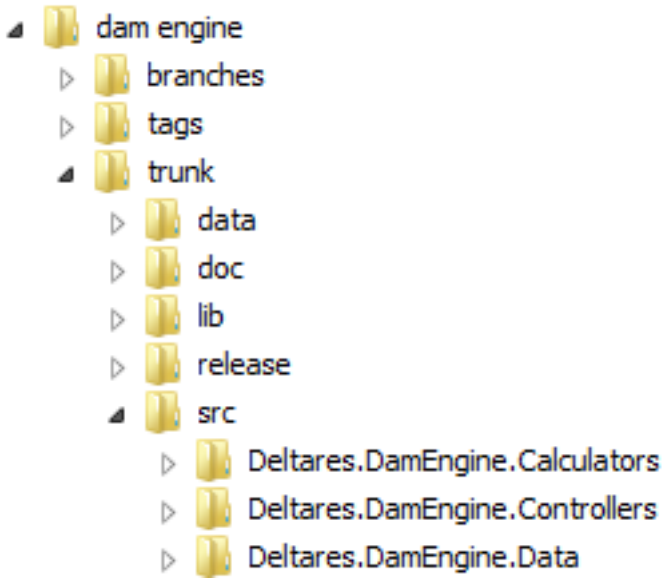

Figure 5.4: *DAM Engine layout.*

At this moment the project foreseen to be put in the "'dam engine"' src map are:

- Deltares.DamEngine.Data: this map contains pure data classes.
- Deltares.DamEngine.Controllers: this map contains classea to manipulate above data classes.
- Deltares.DamEngine.Calculators: this map contains classes to perform the actual calculations for the DAM Engine.

5.5 DAM repository full layout

The full layout of the DAM repository will be as shown in Figure 5.5.

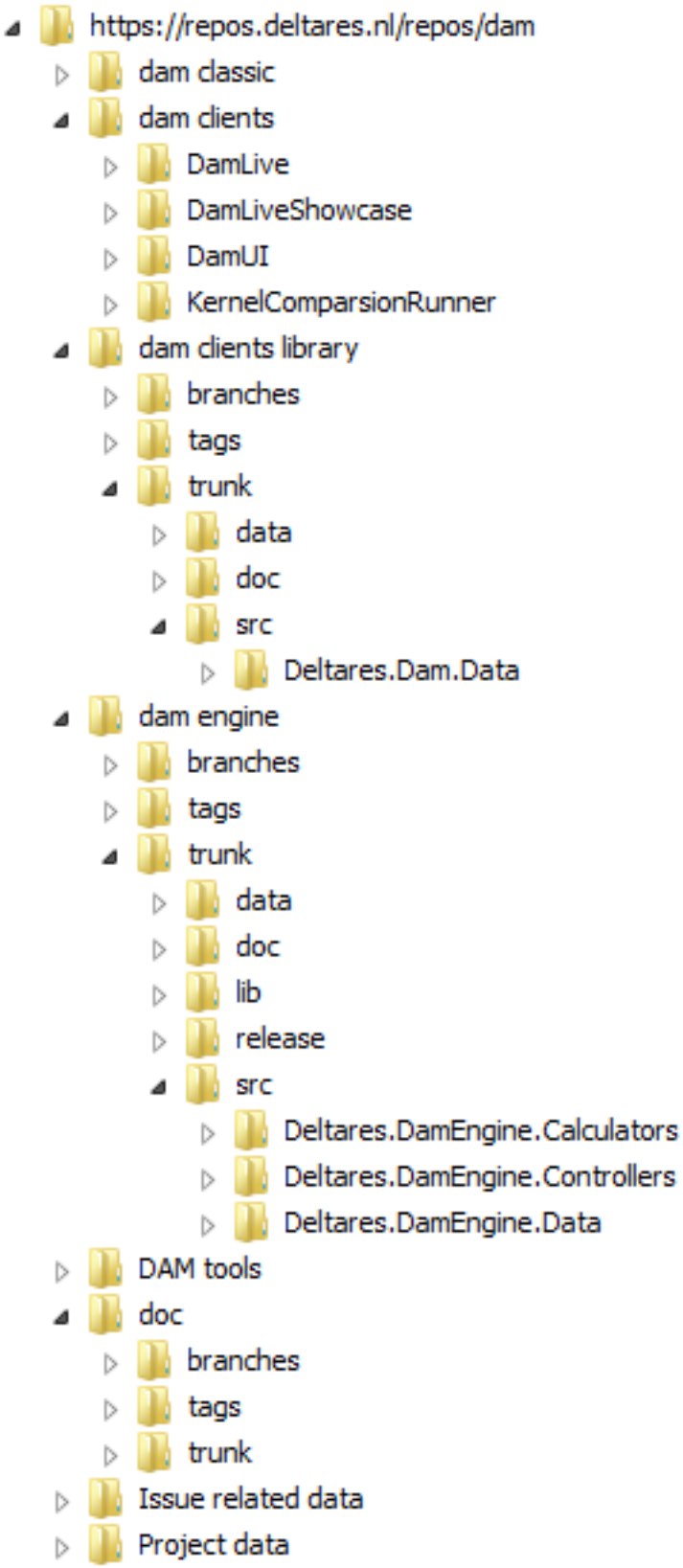

Figure 5.5: *DAM SVN full layout.*

6 Literature

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

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

Erik Vastenburg, I. v. Z., 2013. *DAM UI - Users Manual*. Tech. Rep. 1207094-000-GEO-0002, Versie 2, 11 juni 2013, concept, 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.

The, T., 2017c. *DAM UI - Technical Design*. Tech. Rep. 1210702-000-GEO-0009, version 0.1, jan. 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.

Trompille, V., 2017c. *DAM UI - Test Plan*. Tech. Rep. 1210702-000-GEO-0010, version 0.1, jan. 2017, concept, Deltares.

Trompille, V., 2017d. *DAM UI - Test Report*. Tech. Rep. 1210702-000-GEO-0011, version 0.1, jan. 2017, concept, Deltares.

Zwan, I. v., 2017. *DAM UI - Functional design*. Tech. Rep. 1207094-000-GEO-0005, Versie 2, 11 juni 2013, concept, Deltares.