

Software Requirements Specification for Drasil Matrix, Vector and Tensor Extension

Christopher W. Schankula

January 24, 2025

Contents

1	Reference Material	iv
1.1	Table of Symbols	iv
1.2	Abbreviations and Acronyms	iv
1.3	Mathematical Notation	iv
2	Introduction	1
2.1	Purpose of Document	1
2.2	Scope of Requirements	1
2.3	Characteristics of Intended Reader	1
2.4	Overview of Drasil Project	1
2.5	Organization of Document	1
3	Related Work	2
3.1	Tensors in Haskell	2
3.2	Tensors in Other Languages	2
4	Mathematical Definitions	2
4.1	Tensors	2
4.1.1	Terms	3
4.1.2	Data Storage	3
4.1.3	Einstein Summation Notation	3
4.1.4	Transformation Rules	3
4.1.5	Allowed Operations	3
4.2	Vectors	3
4.2.1	Terms	3
4.2.2	Notation	3
4.2.3	Allowed Operations	3
4.3	Matrices	3
4.3.1	Terms	3
4.3.2	Notation	3
4.3.3	Allowed Operations	3
5	Vectors and Matrices Defined as Tensors	3
5.1	Vectors	3
5.1.1	Tensor Definition of Vectors	3
5.1.2	Allowed Operations as Tensors	3
5.2	Matrices	3
5.2.1	Tensor Definition of Vectors	3
5.2.2	Allowed Operations as Tensors	3

6	Requirements	3
6.1	General Requirements	3
6.2	Test-Driven Requirements	3

Revision History

Date	Version	Notes
January 23rd	1.0	Initial Work on Document for Presentation
Date 2	1.1	Notes

1 Reference Material

This section records information for easy reference.

1.1 Table of Symbols

The table that follows summarizes the symbols used in this document along with their units. The choice of symbols was made to be consistent with the heat transfer literature and with existing documentation for solar water heating systems. The symbols are listed in alphabetical order.

symbol	unit	description
A_C	m^2	coil surface area
A_{in}	m^2	surface area over which heat is transferred in

[Use your problems actual symbols. The si package is a good idea to use for units. —TPLT]

1.2 Abbreviations and Acronyms

symbol	description
A	Assumption
DD	Data Definition
GD	General Definition
GS	Goal Statement
IM	Instance Model
LC	Likely Change
PS	Physical System Description
R	Requirement
SRS	Software Requirements Specification
Drasil Matrix, Vector and Tensor Extension	[put an expanded version of your program name here (as a
TM	Theoretical Model

[Add any other abbreviations or acronyms that you add —TPLT]

1.3 Mathematical Notation

2 Introduction

This introduction section states the purpose of this document, the scope of the requirements, the characteristics of the intended reader, an overview of the Drasil project, and describes the organization of the rest of the document.

2.1 Purpose of Document

The purpose of this document is to document the necessary mathematical background for, and the software requirements of, an extension of the Drasil project encoding tensor, vector, and matrix operations. It is intended to allow the different stakeholders to communicate about and iterate on the project in a formal way. The document will likely be updated throughout the project, and the changes will be recorded in the Revision History table.

2.2 Scope of Requirements

The scope of the requirements will be related to the addition of tensors, vectors, and matrices to the Drasil project.

2.3 Characteristics of Intended Reader

2.4 Overview of Drasil Project

Drasil is “a framework a framework for generating all of the software artifacts from a stable knowledge base, focusing currently on scientific software”. The framework is written in Haskell and allows generation of Software Requirements Specifications, Python, Java, C-Sharp, and C++ code, README files, and Makefiles.

2.5 Organization of Document

The rest of the document is organized as follows: Section 3 presents related work in the area of tensors, Section 4 presents the mathematical definitions, transformation rules and allowed operations on tensors, vectors, and matrices. Then, Section 5 describes how vectors and matrices can be defined as special cases of tensors. Finally, Section 6 presents general requirements for the system extension as well as a set of “test-driven” requirements, denoting some example scientific problems to be encoded and solved using this new system, which will provide oracles with which to test its correctness.

3 Related Work

3.1 Tensors in Haskell

3.2 Tensors in Other Languages

4 Mathematical Definitions

This section provides mathematical definitions of tensors, vectors, and matrices. This includes terminology needed for each, the notation used to describe each one, the transformation rules governing each one, and the allowed operations we are targeting for the software. Note that vectors and matrices will be defined here without relying on their definition as a tensor; Section [5](#) will redefine them using tensors.

4.1 Tensors

A *tensor* is a mathematical

- 4.1.1 Terms
- 4.1.2 Data Storage
- 4.1.3 Einstein Summation Notation
- 4.1.4 Transformation Rules
- 4.1.5 Allowed Operations

4.2 Vectors

- 4.2.1 Terms
- 4.2.2 Notation
- 4.2.3 Allowed Operations

4.3 Matrices

- 4.3.1 Terms
- 4.3.2 Notation
- 4.3.3 Allowed Operations

5 Vectors and Matrices Defined as Tensors

5.1 Vectors

- 5.1.1 Tensor Definition of Vectors
- 5.1.2 Allowed Operations as Tensors

5.2 Matrices

- 5.2.1 Tensor Definition of Vectors
- 5.2.2 Allowed Operations as Tensors

6 Requirements

6.1 General Requirements

6.2 Test-Driven Requirements

References