

# Units for MathWebSearch\*

## Guided Research Proposal

EdN:1

Tom Wiesing  
Supervisor: Michael Kohlhase  
Jacobs University, Bremen, Germany

December 31, 2014

### Abstract

In this proposal we describe an approach to introduce Units to MathWebSearch<sup>2</sup>

EdN:2

## 1 Introduction

3

EdN:3

MathWebSearch (MWS for short<sup>4</sup>) is a system to search (latex) documents for mathematical formulae. Additionally it can also search for text in the documents<sup>5</sup>. However it not only searches for formulae in a simple-minded string way but also includes simple transformation rules, such as  $a + b = b + a$ . Additionally, it is possible to search with wildcards such as  $\textcolor{red}{x} + \sqrt{\textcolor{red}{x}}$ . In this example MWS delivers results of the given form where  $\textcolor{red}{x}$  is substituted with any sub-formular<sup>6</sup>.

EdN:4

EdN:5

EdN:6

MWS has been shown to be very useful for mathematicians<sup>7</sup>. The transformation system it uses can currently be used only for mathematical formulae which limits its applications. In this paper we propose an extension for physical units<sup>8</sup>. Instead of transforming mathematical formulae, the search

EdN:7

EdN:8

---

\*EdNOTE: Preliminary Title

<sup>2</sup>EdNOTE: Write abstract properly

<sup>3</sup>EdNOTE: Write an introductory sentence / paragraph?

<sup>4</sup>EdNOTE: should I really use abbreviations here?

<sup>5</sup>EdNOTE: Or is this the multi-faceted search that is currently planned? Do we really need this sentence?

<sup>6</sup>EdNOTE: Re-formulate this and link to an example

<sup>7</sup>EdNOTE: Quote needed

<sup>8</sup>EdNOTE: Reformulate this?

engine should transfer physical units. The end-user will search, for example, 100 °C and also get results which show 212°F or 373.15K.

This proposal is organised as follows<sup>9</sup>: In section 2 we shortly describe and discuss the existing MathWebSearch system and then proceed in section 3 to describe in detail the proposed extension. Finally in section 4 we discuss possible problems with this approach and related work. EdN:9

## 2 Background - The existing MathWebSearch system

- searches for math formulars
- has a corpus of ??? <sup>10</sup> documents EdN:10
- currently used by ??? and ??? <sup>11</sup> EdN:11

The frontend

- Written in HTML/CSS/JavaScript
- accessing a REST backend

The backend

- written in <sup>12</sup> EdN:12
- has a big search index
- searches the index only

<sup>13</sup> EdN:13

<sup>14</sup> EdN:14

Advantages of this approach / hwo MWS is received <sup>15</sup> EdN:15

Disadvantages of the current approach

- has to be re-generated each time a document is added

---

<sup>9</sup>EdNOTE: Update this possible if we change the structure

<sup>10</sup>EdNOTE: Get an estimate here

<sup>11</sup>EdNOTE: where?

<sup>12</sup>EdNOTE: find out

<sup>13</sup>EdNOTE: Explain theory graphs here

<sup>14</sup>EdNOTE: Explain how MWS uses those here

<sup>15</sup>EdNOTE: Find points here

### 3 The proposed extension

- want a complete system
- searches a corpus of documents for units
- which is presentable to the end user
- should be extendable with respect to
  - the corpus. Plugging in a new corpus should be as easy as running a script somewhere.
  - the units. Adding new units should be simple by just adding a conversion to one already known unit.

The frontend

- a web page
- should work in modern browsers, preferably mobile-friendly
- should only be a frontend for a REST backend
- has an input for a unit
- has an input for a value
- maybe have facetted search on top

The backend

- REST based
- based on the existing system
- has to have a format of units
- has to receive text queries
- has to receive exact values or ranges or automatically generated ranges

The corpus

- should consist of a lot of tex documents
- should have marked up units

- ideally, if a single document is added, only the new corpus should have to be re-scanned (procedural generation)
- should be easily exchangeable

The unit transition system

- should be a graph
- should have few connected components and each of the components should be sparse (i. e. few connections)
- translation are:
  - either a factor towards a single unit
  - or a composition of a factor together with a product or fraction of units <sup>16</sup>
  - Perhaps include prefixes somehow?

EdN:16

17

EdN:17

## 4 Problems and related Work

The unit input system:

- Entering a single unit and recognising it is simple
- It is not clear how to enter composite units
- the end result delivered to the search engine should either be LaTeX or MathML
- maybe allow different input methods:
  - The output latex
  - AsciiMath (with autocompletion would be nice)
  - MathML?
- system needs to be aware of full unit names as well as abbreviations

Unit translation

---

<sup>16</sup>EdNOTE: Figure out more details about this

<sup>17</sup>EdNOTE: Write this

- Should just be rational factors
- might give a problem with rounding
- maybe have ranges instead

– this has just been implemented by Radu <sup>18</sup>

EdN:18

- support for composite units:  $a \cdot b$  and  $\frac{a}{b}$ .

Finding a corpus

- We need to have a suitably large corpus of documents to test this properly
- the units need to be marked up in the corpus
- actually finding them is done by ??? <sup>19</sup>
- The results should show which unit is originally in the text and also show the value in the unit searched for.

EdN:19

<sup>20</sup>

EdN:20

---

<sup>18</sup>EdNOTE: Quote needed

<sup>19</sup>EdNOTE: Who is doing the unit finding?

<sup>20</sup>EdNOTE: Write this