Literate Scientific Software & The Drasil Framework

Dan Szymczak

Computing and Software Department
Faculty of Engineering
McMaster University

Ernie Mileta Visit, Jan. 24, 2017

Overview

1 Fully Traceable Software

2 Drasil Today

3 Results

4 Next Steps

McMaster University

Slide 3 of 23

FTS
Drasil
Results
Next Steps

Fully Traceable Software

- Motivation
  - Improve verifiability, maintainability and reusability.
  - Save money and time
- One "source," multiple views
  - Requirements
  - Design
  - Test Cases
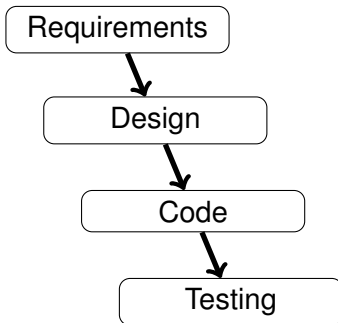 - Build instructions
  - ...

McMaster
University

Slide 4 of 23

FTS

Drasil

Results

Next Steps

Motivation

McMaster
University

Slide 5 of 23

FTS

Drasil

Results

Next Steps

Motivation

McMaster University

Slide 6 of 23

FTS
Drasil
Results
Next Steps

Motivation

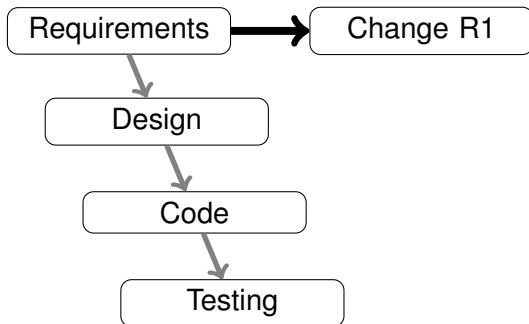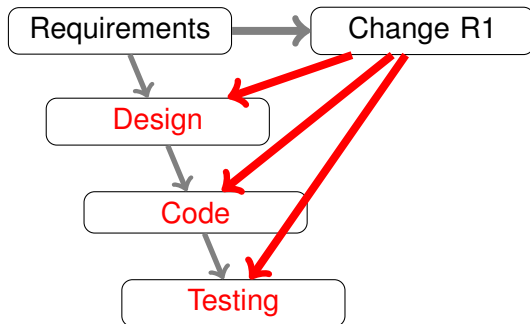McMaster
University

Slide 7 of 23

FTS

Drasil

Results

Next Steps

Last Time

- *Knowledge capture*, *chunks*, and *recipes*
- Common Knowledge Database
- Document Language
- Steve
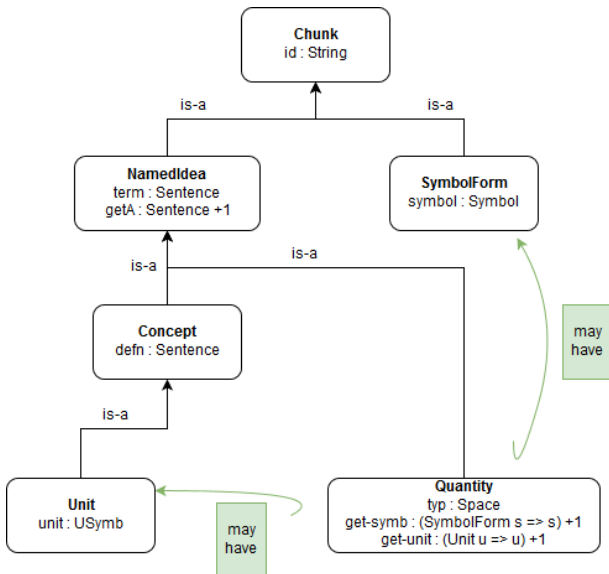- Summer Students Phase 1
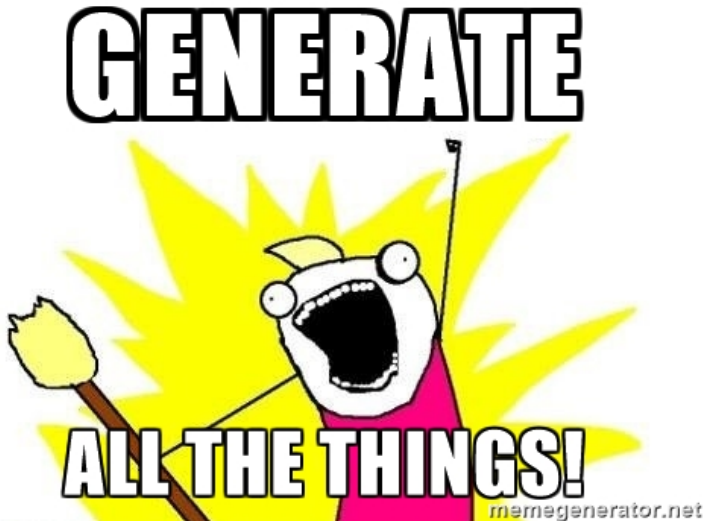
McMaster University

Slide 8 of 23

FTS
Drasil
Results
Next Steps

Recap - Knowledge Capture

Recap - Chunks

McMaster
University

Slide 10 of 23

FTS
Drasil
Results
Next Steps

Recap - Generation

Recap - Drasil

McMaster
University
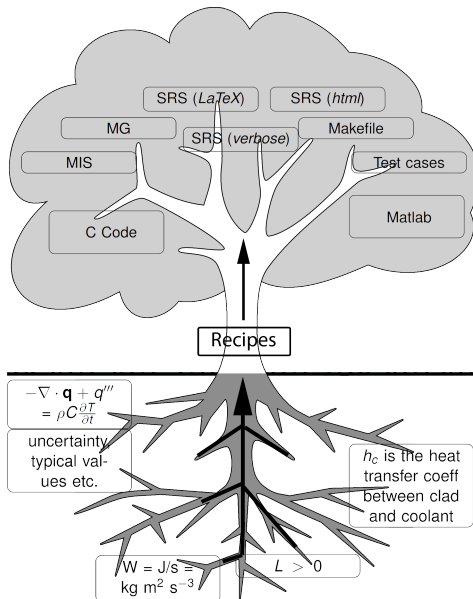
Slide 12 of 23

FTS
Drasil
Results
Next Steps

What's New
Summer Students Phase 2

- Example clean-up
- Knowledge extraction (common + specific)
- Pattern finding & combinator creation

McMaster
University

Slide 13 of 23

FTS
Drasil
Results
Next Steps

What's New
Data.Drasil

Common Knowledge base expanded! Now includes:

1. Documentation
2. Thermodynamics
3. Computation
4. Physics
5. Math
6. Solid Mechanics

and more!

What's New

Yuzhi and Devi

New graduate students as of September 2017.

- Reviewed and updated manual versions of current examples
- Implementing Document Language through examples.
- . . . and more to come!

McMaster University

Slide 15 of 23

FTS

Drasil

Results

Next Steps

Design Changes

New Knowledge Capture Mechanisms

We are able to capture much more information in a 'useful' form

1. Theories
2. Assumptions
3. Requirements
4. Instance Models

...

McMaster
University

Slide 16 of 23

FTS
Drasil
Results
Next Steps

Design Changes

Document Language – Old (Recipes)

```
RefSec ( RefProg intro [
  TUnits ,
  tsymb" s1_2_intro ( TermExcept [ norm_vect ] ),
  TAandA ]) :
map Verbatim [ s2 , s3 , s4 , s5 , s6 , s7 ]

s2 = Section ... ... ...
s3 = Section ... ... ...
s4 = Section ... ... ...
s5 = Section ... ... ...
s6 = Section ... ... ...
s7 = Section ... ... ...
```

McMaster University

Slide 17 of 23

FTS
Drasil
Results
Next Steps

Design Changes
Document Language – New (Recipes)

```
RefSec (RefProg intro
  [TUnits, tsymb [TSPurpose, SymbOrder], TAandA]) :
IntroSec ( IntroProg (startIntro ...) (short gLassBR)
  [IPurpose (s2_1_intro_p1 document gLassBR glaSlab),
   IScope incScoR endScoR,
...
GSDSec (GSDProg2 [UsrChars ...], SystCons [] []]) :
ScpOfProjSec (ScpOfProjProg ...) :
SSDSec (SSDProg
  [SSDProblem (PDProg ... [s6_1_1, s6_1_2, s6_1_3])
  , SSDSolChSpec
    (SCSProg
      [ TMs ([Label] ++ stdFields) [t1IsSafe]
      , GDs [] [] HideDerivation —— No Gen Defs for GlassBR
      , DDs ... , IMs ... ])]) :
ReqrmntSec (ReqsProg [
  FReqsSub s7_1_list,
  NonFReqsSub [performance] (gBRpriorityNFReqs) :
...
AuxConstntSec (AuxConsProg gLassBR auxiliaryConstants) :
Bibliography gbCitations :
AppndxSec (AppndxProg [s12_intro, fig_5, fig_6]) : []
```

McMaster University

Slide 18 of 23

FTS
Drasil
Results
Next Steps

Results
Document Language – New (Recipes)

GlassBR_SRS.html

McMaster University

Slide 19 of 23

FTS
Drasil
Results
Next Steps

Results to Date
Sanity Checking

SSP Example (Issue #348)

$$S_i = \frac{P_i}{FS} \qquad (1) \qquad FS = \frac{S_i}{\tau_i} \qquad (2)$$

Where did τ_i come from?

Were S_i and P_i swapped?

- τ_i was not defined anywhere in the documents
- Found with Drasil – undefined symbols throw errors
- Equation based on concepts – symbols automatically retrieved

McMaster
University

Slide 20 of 23

FTS

Drasil

Results

Next Steps

Results to Date
Sanity Checking Cont'd

- Stricter (more formalized) approach reveals hidden errors.
- Fixing these kinds of errors is easy thanks to tool support from Drasil

McMaster
University

Slide 21 of 23

FTS
Drasil
Results
Next Steps

Results to Date
Conceptual Inconsistencies

What do our documents "mean"?
DD vs IM stuff here. Conceptually confused in the documentation. Understandable by a human, but causes unexpected problems when explained to a machine.

- DD7

McMaster
University

Slide 22 of 23

FTS
Drasil
Results
Next Steps

Next Steps

What next?

- Recipe language
- Clean up examples
- Abstraction
- More examples
- More artifacts – Module Guide
- Design Language

Thank You!