PhD Committee Meeting #4

Dan Szymczak

Computing and Software Department
Faculty of Engineering
McMaster University

June 28, 2018

Overview

1. **Introduction**

2. Current Progress

3. Example.

4. Next Steps.

Overview

1 Introduction

2 Current Progress

3 Example.

4 Next Steps.

Overview

1 Introduction

2 Current Progress

3 Example.

4 Next Steps.

Overview

1 Introduction

2 Current Progress

3 Example.

4 Next Steps.

Current Program Progress

- Completed all necessary graduate courses.
- Completed comprehensive examinations.
- Drasil framework proof-of-concept completed
    - Scoped-down section due to nature of project
    - Generating Software Requirements Specification (SRS) for six case studies + code for one

Current Progress
Cont'd

Currently Writing:

- Journal paper for ACM TOSEM
- Thesis

McMaster University

Slide 8 of 20

Introduction

Progress

Example

Next Steps

Research Topic Recap
Motivation

- Too much duplication!
- (Re-)Certification is expensive
- Ensure inter-/intra-artifact consistency & improve:
    - Verifiability
    - Maintainability
    - Reusability
- Promote reusability
- Design for change – Easily switch between software family members

Drasil Framework – Generating Software Families

- One "source", multiple views
  - Requirements
  - Design
  - Test Cases
  - Build instructions
  - ...

Research Topic
How?

Knowledge Capture!

- Advantages
  - Avoid duplication through chunk reuse.
  - Improve understandability, traceability and reproducibility.
  - Increased flexibility

McMaster
University

Slide 11 of 20

Introduction
Progress
Example
Next Steps

Example: h_g
A simple example taken from the SRS for FP

h_g is a symbol which appears in several locations including:

- The Software Requirements Specification
- The Literate Programmer's Manual
- The Source Code

Let's take a look!

McMaster University

Slide 12 of 20

Introduction
Progress
Example
Next Steps

Example: h_g
SRS Definition for h_g (original)

Number	DD1
Label	h_g
Units	$ML^0 t^{-3} T^{-1}$
SI	$\frac{\text{kW}}{\text{m}^2 ({}^\circ\text{C})}$
Equation	$h_g = \frac{2k_c h_p}{2k_c + \tau_c h_p}$
Description	h_g is the gap conductance τ_c is the clad thickness h_p is initial gap film conductance k_c is the clad conductivity NOTE: Equation taken from the code
Sources	source code

McMaster University

Slide 13 of 20

Introduction
Progress
Example
Next Steps

Example: h_g

LPM Definition for h_g (original)

$$h_g = \frac{2k_c h_p}{2k_c + \tau_c h_p} \tag{1}$$

The corresponding C code is given by:

```
double calc_hg (double k_c, double h_b, double tau_c)
{
 return (2*(k_c)*(h_p)) / ((2*(k_c)) + (tau_c*(h_p)));
}
```

McMaster
University

Slide 14 of 20

Introduction
Progress
Example
Next Steps

Example: h_g

A simple example taken from the SRS for FP

Modifying h_g to reflect changes in requirements is not a simple matter. It involves, at the very least, the following steps:

- Update the definition in the SRS, LPM, and all other documents which reference the symbol
- Modify the source code to reflect the new requirements
- Trace all dependencies
- Modify dependents to accomodate the change
- Ensure each of the documents is now up to date and consistent

McMaster University

Slide 15 of 20

Introduction

Progress

Example

Next Steps

Example: h_g
Simplifying the process

Here is an example of a "chunk" for h_g:

```
--------------- --------------- --------------- ---------------
{--------------- Begin h_g ---------------}
--------------- --------------- --------------- ---------------

h_g :: Chunk
h_g = newChunk $
  [(Symbol, S "h" :-: S "g"),
   (Equation, E h_g_eq),
   (SIU, S "($\\mathrm{\\frac{kW}{m^2C}}$)"),
   (Description, S
    "effective heat transfer coefficient between clad and fuel surface")
  ]

h_g_dep :: Dependency
h_g_dep = get_dep h_g_eq

h_g_eq :: Expr
h_g_eq = ((Int 2):*(C k_c):*(C h_p)) :/ ((Int 2):*(C k_c):+((C tau_c):*(C h_p)))
```

McMaster University

Slide 16 of 20

Introduction
Progress
Example
Next Steps

Example: h_g
How do we generate?

What do we do with the "chunk"?
That depends on the "recipe"!

- To create our SRS we use the following recipe:

```
createSRS :: Doc
createSRS = spre $$ doctitle $$
            author auth $$ srsComms $$
            begin $$ srsBody $$ end
```

- To create our LPM we use the following recipe:

```
createLPM :: Doc
createLPM = lpre $$ doctitle $$
            author auth $$ lpmComms $$
            begin $$ lpmBody $$ endL
```

McMaster University

Slide 17 of 20

Introduction
Progress
Example
Next Steps

Example: h_g
Generated SRS Output

Number	DD2
Label	h_g
Units	$ML^0 t^{-3} T^{-1}$
SI	$\frac{\text{kW}}{\text{m}^{2\circ}\text{C}}$
Equation	$h_g = \frac{2k_c h_p}{2k_c + \tau_c h_p}$
Description	h_g is the effective heat transfer coefficient between clad and fuel surface k_c is the clad conductivity h_p is the initial gap film conductance τ_c is the clad thickness NOTE: Equation taken from the code
Sources	source code

McMaster University

Slide 18 of 20

Introduction
Progress
Example
Next Steps

Example: h_g
Generated LPM Output

$$h_g = \frac{2k_c h_p}{2k_c + \tau_c h_p} \qquad (2)$$

The corresponding C code is given by:

```
double calc_h_g(double k_c, double h_p, double tau_c)
{
        return 2*k_c*h_p/(2*k_c+tau_c*h_p);
}
```

Next Steps
Broad Strokes

What next?

- Comprehensive examination part two.
- Complete final graduate level course.
    - Looking for a category theory course, but open to suggestions.
- Complete thesis.

Thank You!