

## Aggregating Documents in SWiM

Author: Zdravko Beykov

Supervisors: Prof. Michael Kohlhase and Christoph Lange

Date: 16.05.2008

# Table of contents

- Introduction
- Overview of technologies
- Contextual Modelling
- Implementation
- Critical Evaluation
- Future Work

# Introduction

- Getting information nowadays
  - find it
  - narrow it down
  - see what it is linked to

# Introduction

- Solution: use semantic document aggregation applet
  - navigate to needed document
  - choose it's dependencies
  - get a document that combines all your choices
  - print it

# Overview of technologies

- XML
  - has only text, tags and properties
    - `<tag property="new value">this is text</tag>`
  - each tag can contain others and text or be empty
    - `<tag isEmpty="true"/>`
  - can be validated by custom datatype definitions (DTD)

# Overview of technologies

- XML continued
  - can have special variables called entities used
    - Examples: &gt; &lt;
  - API's exist for parsing – DOM
  - Can be easily converted to another text document via an XSL transformation
  - Used a lot
    - HTML, SVG, RSS, MathML

# Overview of technologies

- HTML
  - based on XML
  - marks formatting of content
  - client-side processed
  - allows for limited scripting ability via JavaScript

# Overview of technologies

- JSP
  - Java Server Pages
  - server processed
  - included via tags in HTML
  - intuitive to work with if you know Java



# Overview of technologies

- OMDoc
  - open markup for mathematical documents
  - based on XML
  - stores semantic meaning and relation to other mathematical objects
  - aims for managing large mathematical collections

# Overview of technologies

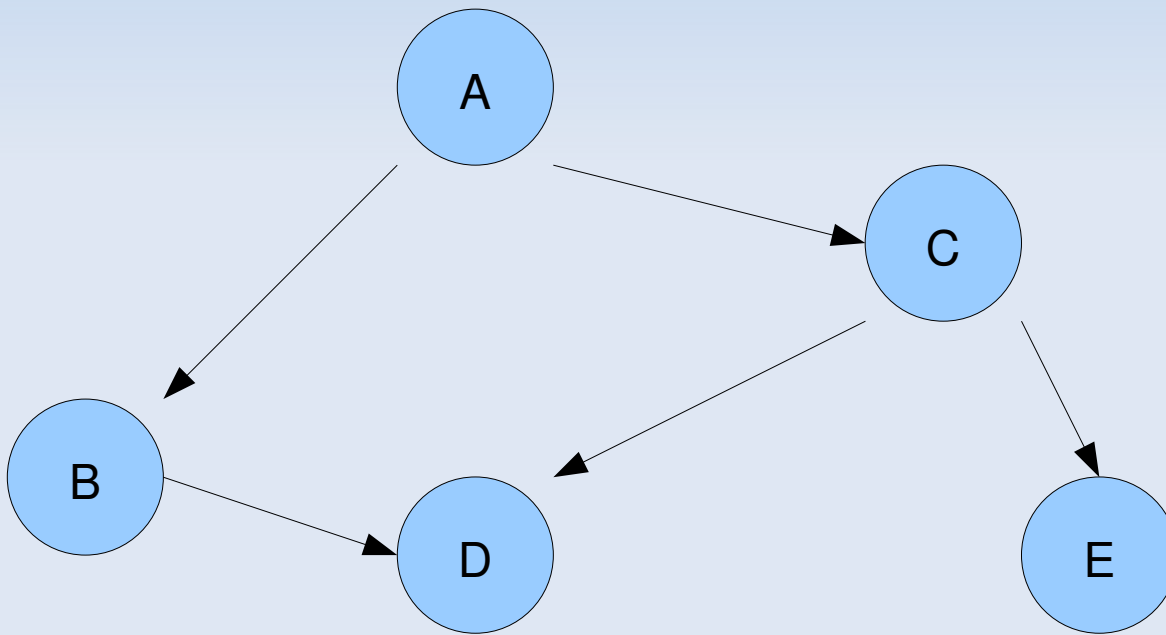
- SWiM
  - semantic wiki for mathematical knowledge management
  - written in JSP and uses an OMDoc database
  - has an applet that visualizes dependencies - IkeWiki
  - demo is available at <http://swim.kwarc.info/>

# Overview of technologies

- Adobe Flash
  - created by Macromedia for simple animations
  - limited scripting abilities that improved in the future
  - currently aimed not only for designers but also developers
  - multiplatformal

# Contextual Modeling

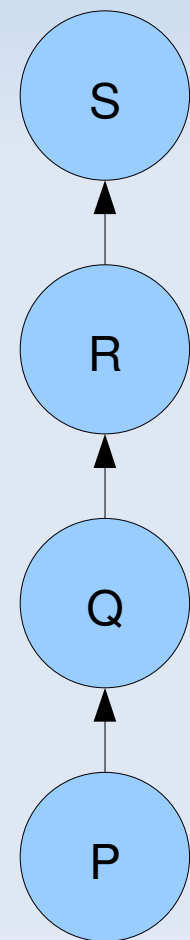
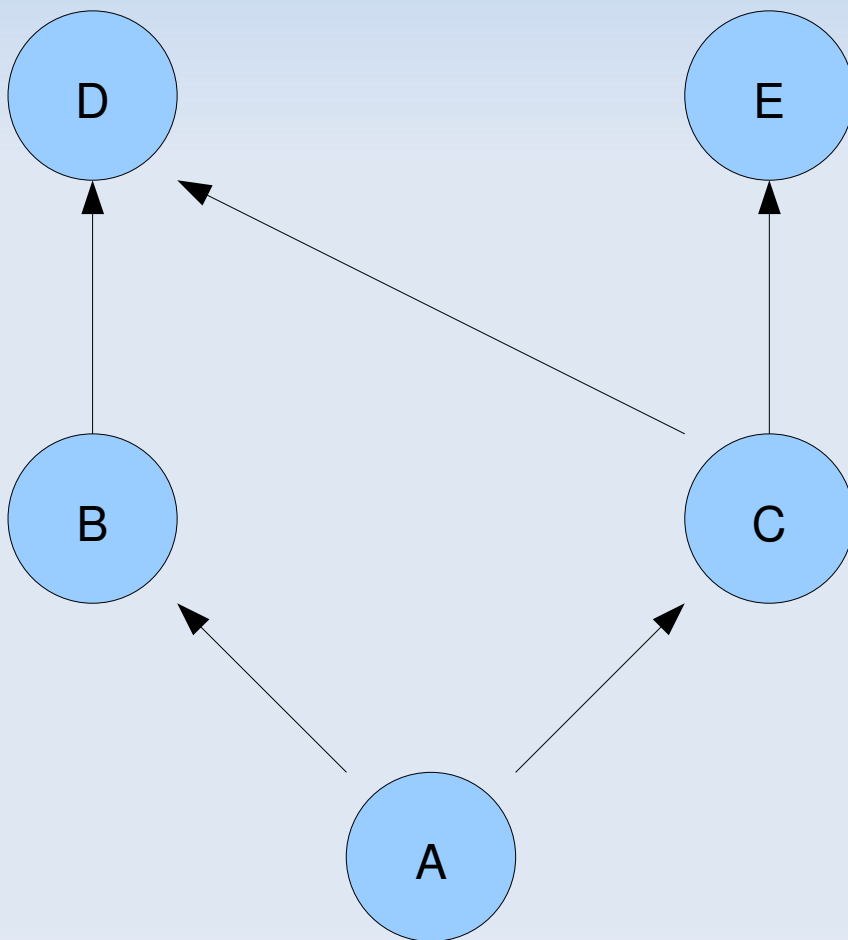
- Representing a graph



+A  
| +B  
| --D  
-+C  
| -D  
--E

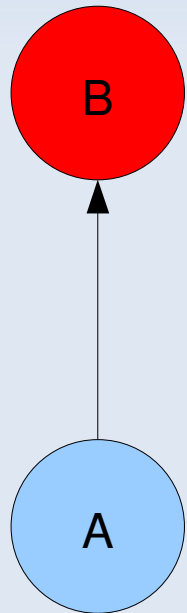
# Contextual Modeling

- Greedy Approach

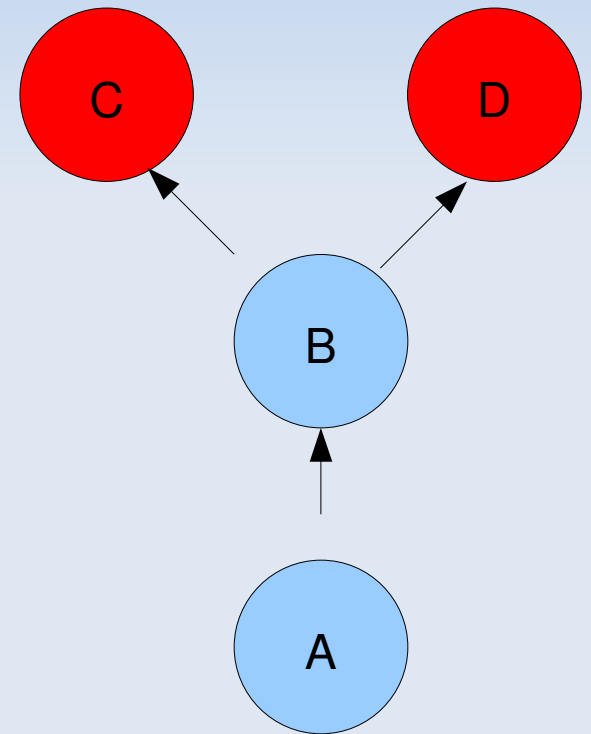


# Contextual Modeling

- Simple Input

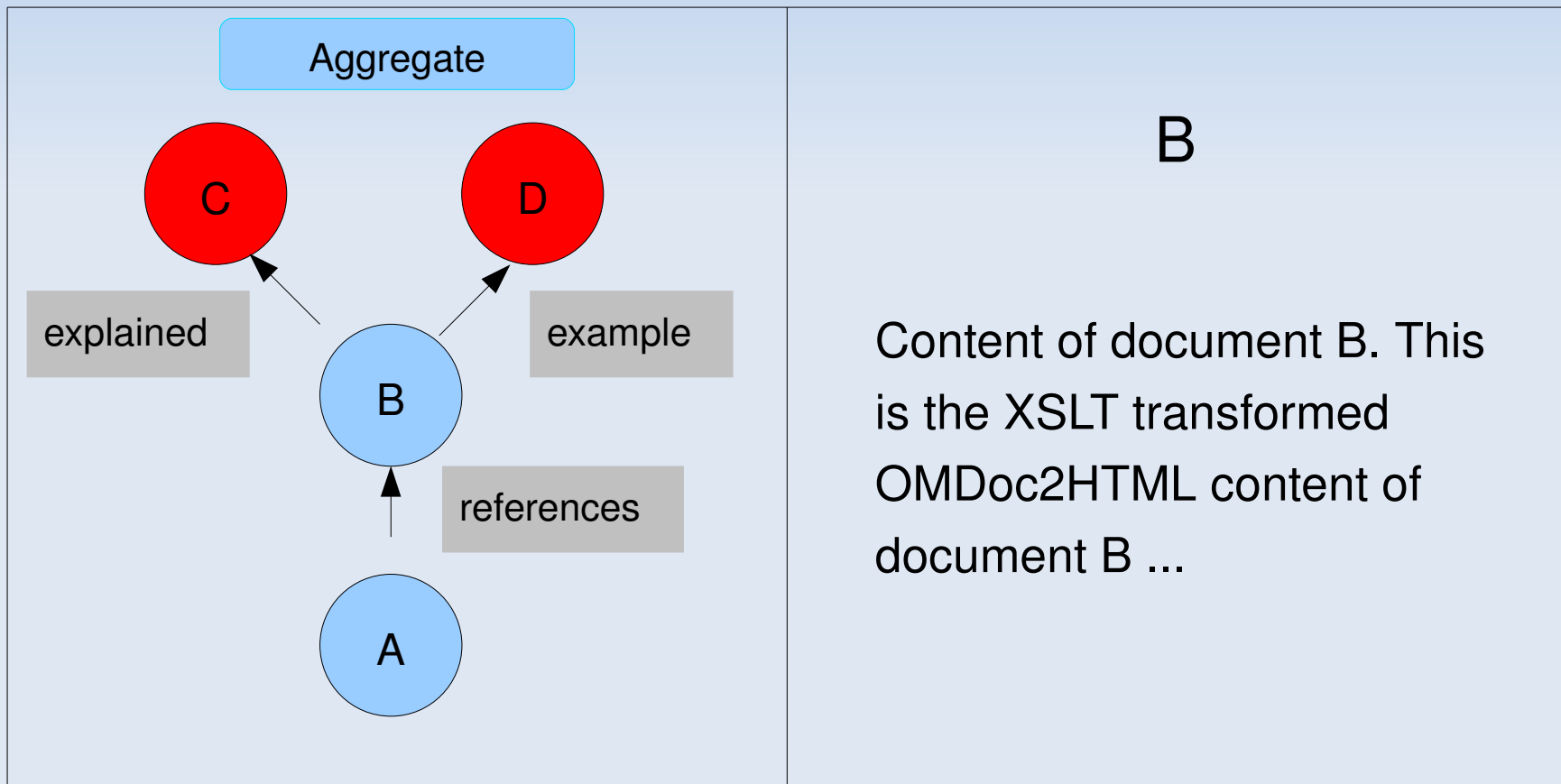


←→  
Left Click On "B"



# Contextual Modeling

- Summing it up

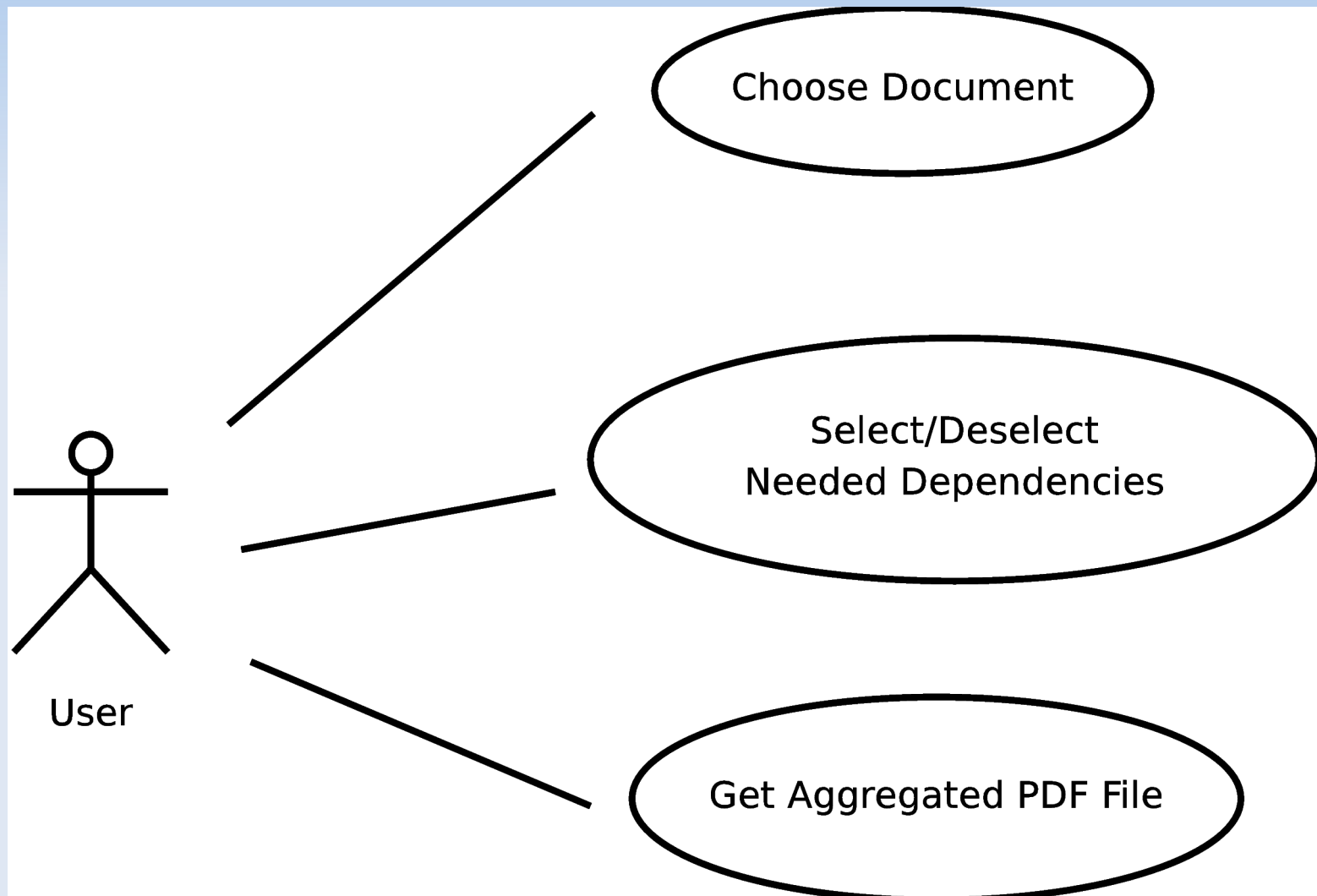


# Implementation

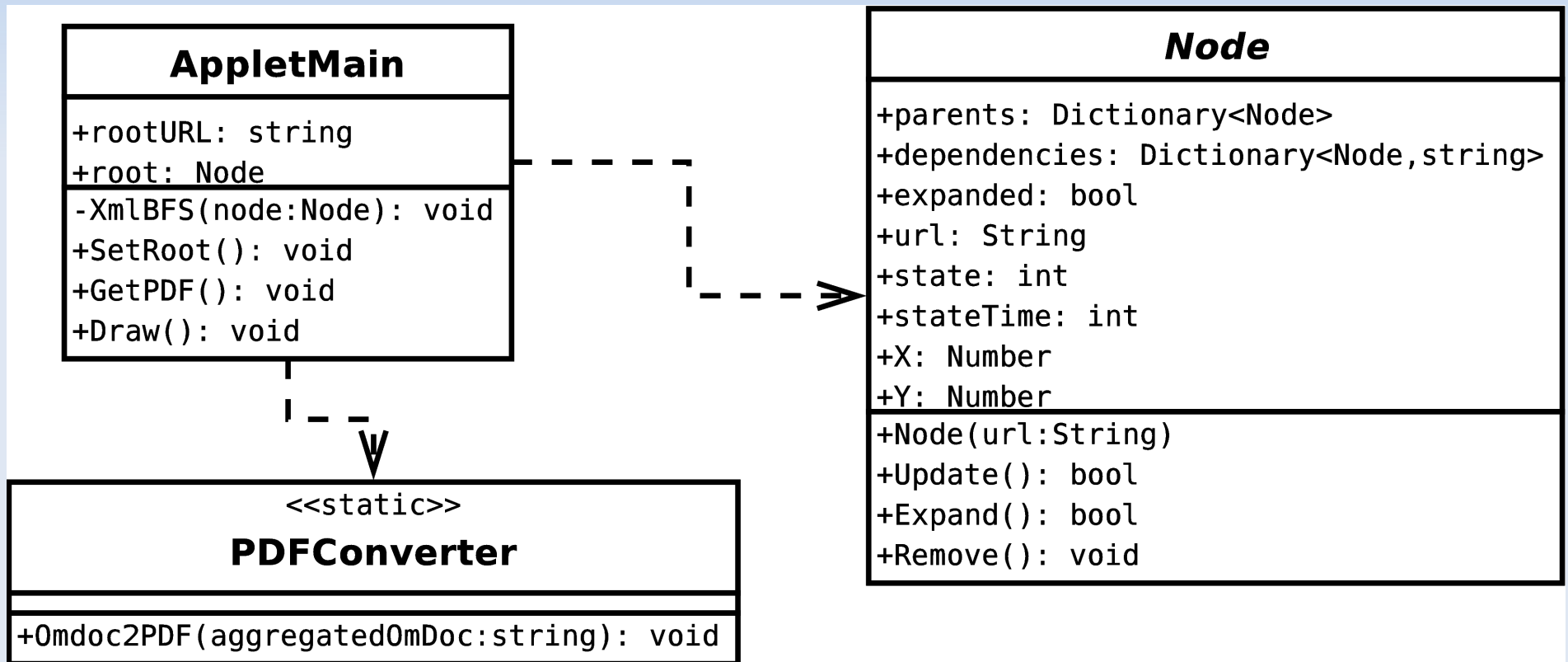
- Use of graph algorithms
  - Breadth First Search
    - Expand nodes by their level
  - Depth First Search
    - Find dependencies and order nodes
- XML communication between JSP and Flash

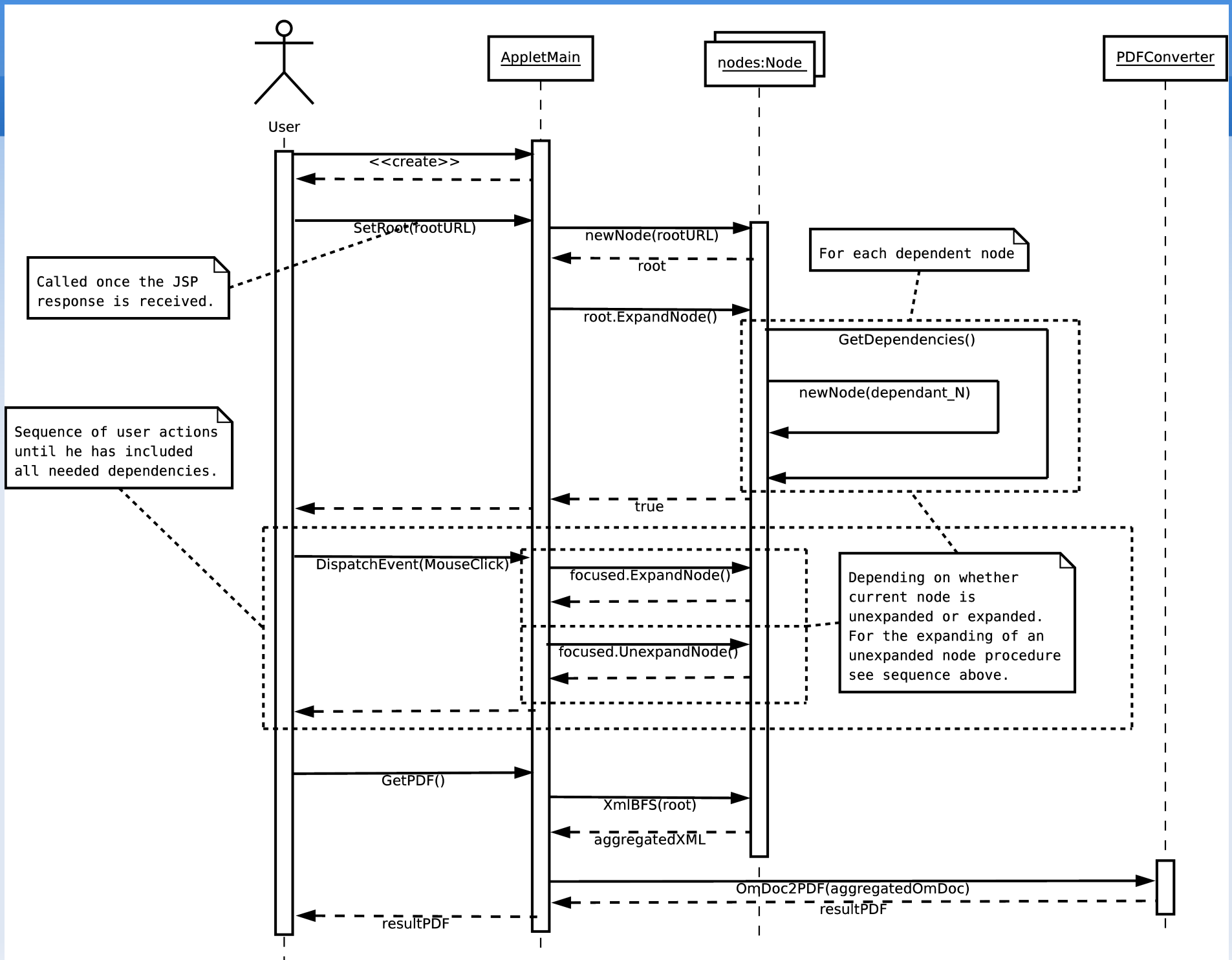


# Implementation



# Implementation





# Critical Evaluation

- Design issues
- Possible improvement in performance
- Shift to .NET

# Demo

- available soon at <http://swim.kwarc.info/>

# Future Work

- open project
- modified on request
- reasonably long time of support

# Conclusion

- Significance