XoWiki Content Flow:

From a Wiki to a Simple Workflow System

Gustaf.Neumann@wu-wien.ac.at
Vienna University of Economics and Business Administration (WU)

November 18, Valencia

Overview

- Short Update of Learn@WU
  - DotLRN based E-Learing Platform of WU-Vienna
- XoWiki Content Flow
  - Basic Design Principles
  - State Aware Application Objects
  - Generalizing XoWiki Concepts
  - State and Context Dependent Input/Output Behavior
  - Application Examples
  - Summary

Learn@WU

- **Development**
  - □ **2002:** Initial Launch, Content Project, based on OpenACS (Learning Content Management System)
  - □ **2003:** E-Learning became strategic goal of the University
  - □ **2004:** Re-launch, based on LCMS + DotLRN
  - □ **2005:** E-Learning part of general Trainee Programmes
  - □ **2006:** Development of an E-Learning Academy
  - □ **2008:** Full coverage of all courses (~5000/year)
    Improving integration with the Campus Management System

Current Key Figures of Learn@WU

- Broad Acceptance
  - □ More than 60.000 learning resources
  - □ More than 32.000 registered members (mostly students)
  - □ Students solve up to 600.000 interactive exercises per day online
  - □ More than 200.000 class-room exams prepared/corrected via Learn@WU (via mark-reader)
  - □ *"Without Learn@WU, the operations of our university would not have been possible"* (Christoph Badelt, President of WU)

- Technical Figures
  - □ Up to 8 Mio hits and 1.8 Mio views per day from registered users
  - □ Average response time on views less than 0.2 sec
  - □ Up to 1800 concurrent users
  - □ No hardware upgrade in last 3.5 years (IBM p570, 16 cores)
  - □ Up to 65 GB/Day traffic

Current annual growth rate: 30%

One of the most intensively used eLearning platforms world-wide

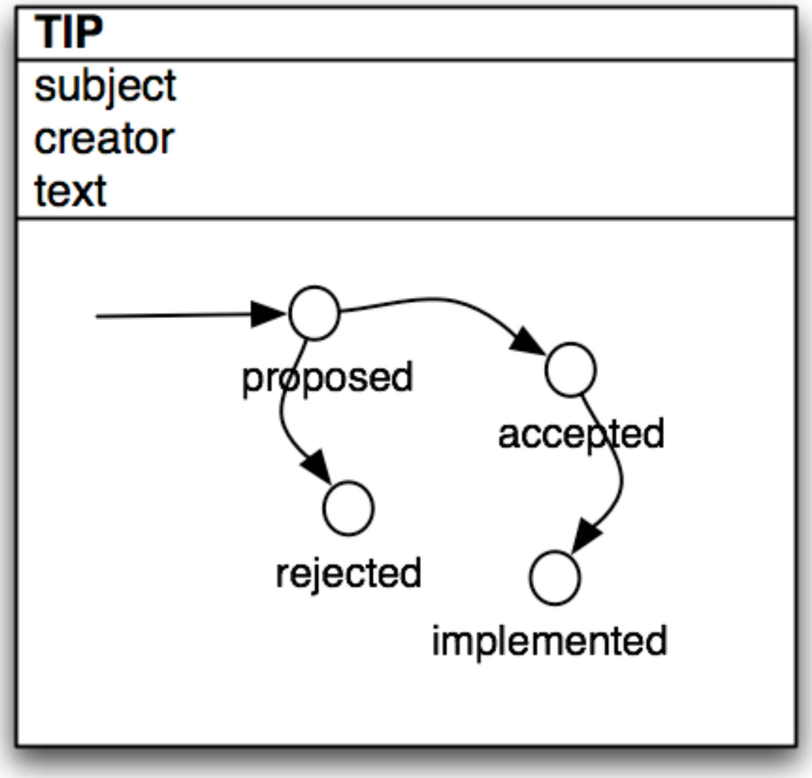
Platform Developments

- Integration with Campus Management System
  - □ Goal: Student should interact for all learning specific activities over a single system
  - □ Done:
    - Course/Course Instance Creation (as minimal communities) for all courses
    - Automatic User Creation, Student and Professor enrollments in communities
    - Automatic management of lecture dates and lecture rooms (subscribable from Calendar software)
- Video-Recording/Podcasting integrated with Course Management
- Increasing usage of XoWiki
  - □ Learning and Intranet applications
  - □ Currently about 300 XoWiki instances on Learn

XoWiki Content Flow

- Motivation
  - Business Processes are typically not talking about process in the technical sense
  - Petri-Nets, Process algebras: Instruments for precise concurrency semantics of multiple processes or hardware units
- XoWiki Content Flow
  - Modeling a single Flow
  - Make States of a Flow explicit
  - Modeling Flows as Objects
  - Suitable for Page-Flows as well as for modeling different states of some application objects

TIP Flow



TIP

subject
creator
text

proposed
accepted
rejected
implemented

Relation to XoWiki Forms

- Content Flow extends XoWiki Forms
  - ::xowiki::Form
    - Defines a set of application specific attributes ("instance_attributes")
    - Defines input/output behavior (template and input form)
    - ... Similar to a Class (application classes)
    - Kept in OpenACS Content repository, multiple revisions
  - ::xowiki::FormPage
    - Contains actual values

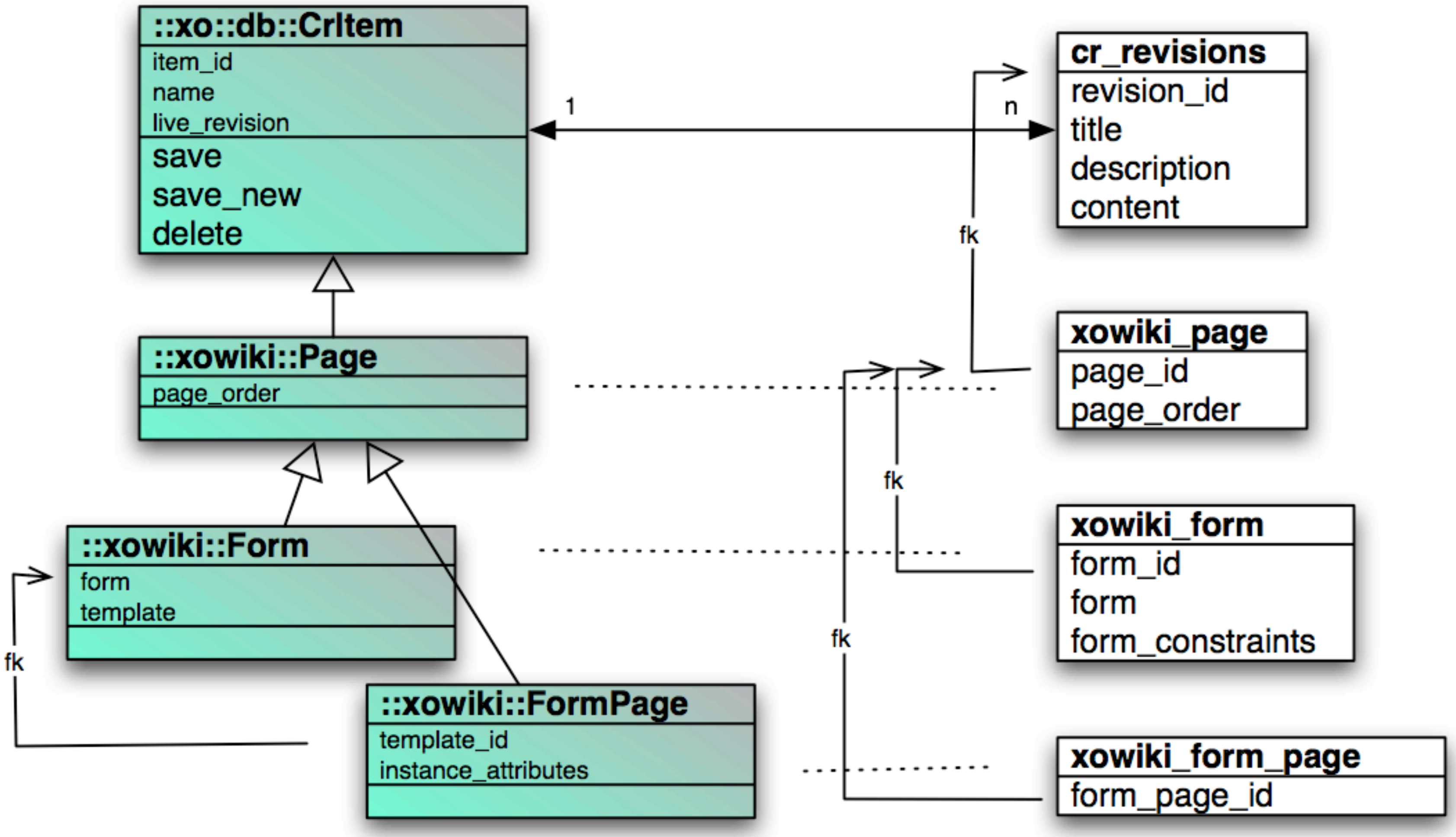
Revisions

$$\begin{pmatrix} a_{1,0} \\ a_{2,0} \\ \ldots \end{pmatrix} \xrightarrow{\alpha_1} \begin{pmatrix} a_{1,1} \\ a_{2,1} \\ \ldots \end{pmatrix} \xrightarrow{\alpha_2} \ldots \xrightarrow{\alpha_n} \begin{pmatrix} a_{1,n} \\ a_{2,n} \\ \ldots \end{pmatrix}$$

Attributes

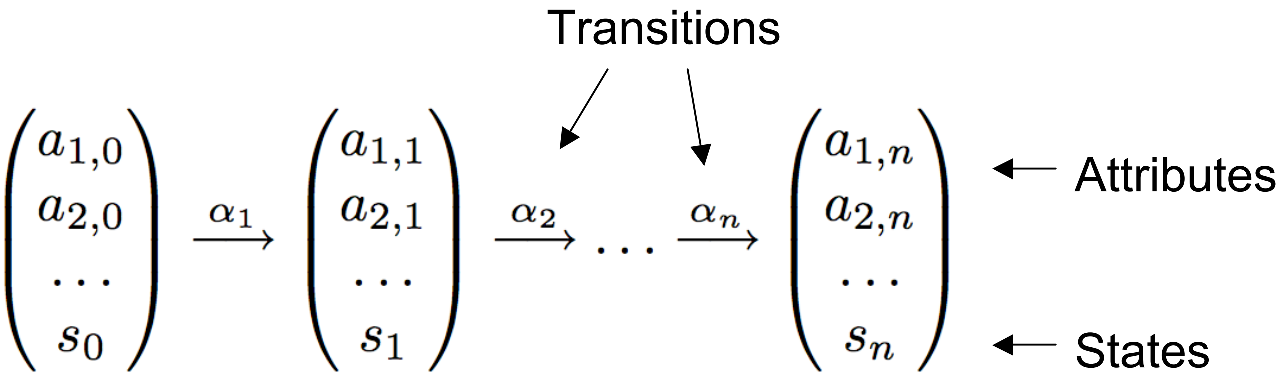
    - ... Similar to an Object (application objects)
    - Kept in OpenACS Content repository, multiple revisions

::xowiki::Form and ::xowiki::FormPage



What’s new in Content Flow (1)

- Explicit State management



$$\begin{pmatrix} a_{1,0} \\ a_{2,0} \\ \dots \\ s_0 \end{pmatrix} \xrightarrow{\alpha_1} \begin{pmatrix} a_{1,1} \\ a_{2,1} \\ \dots \\ s_1 \end{pmatrix} \xrightarrow{\alpha_2} \dots \xrightarrow{\alpha_n} \begin{pmatrix} a_{1,n} \\ a_{2,n} \\ \dots \\ s_n \end{pmatrix}$$

Transitions

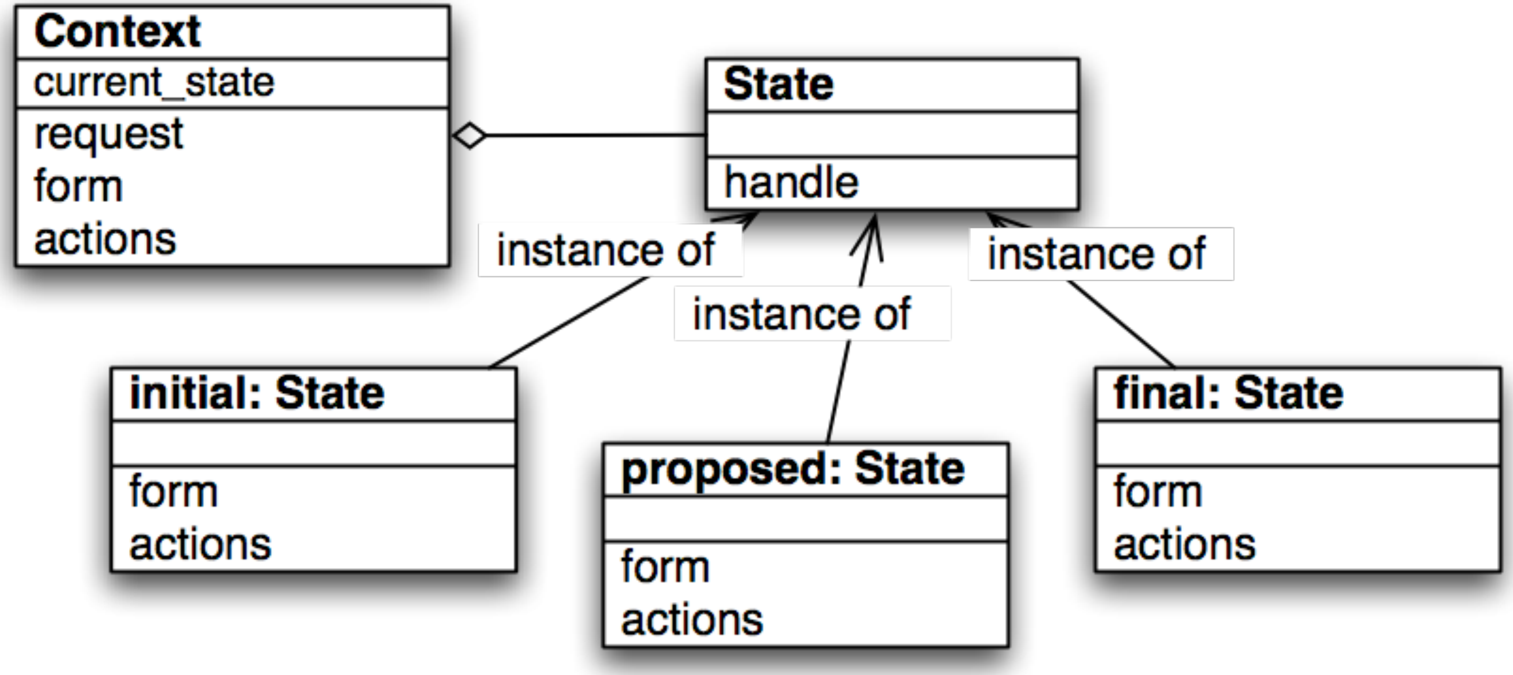
Attributes

States

- State determines
  - □ Applicable Actions
  - □ Forms

- Applicable Actions are presented as multiple HTML FORM buttons.

Realization (simplified)



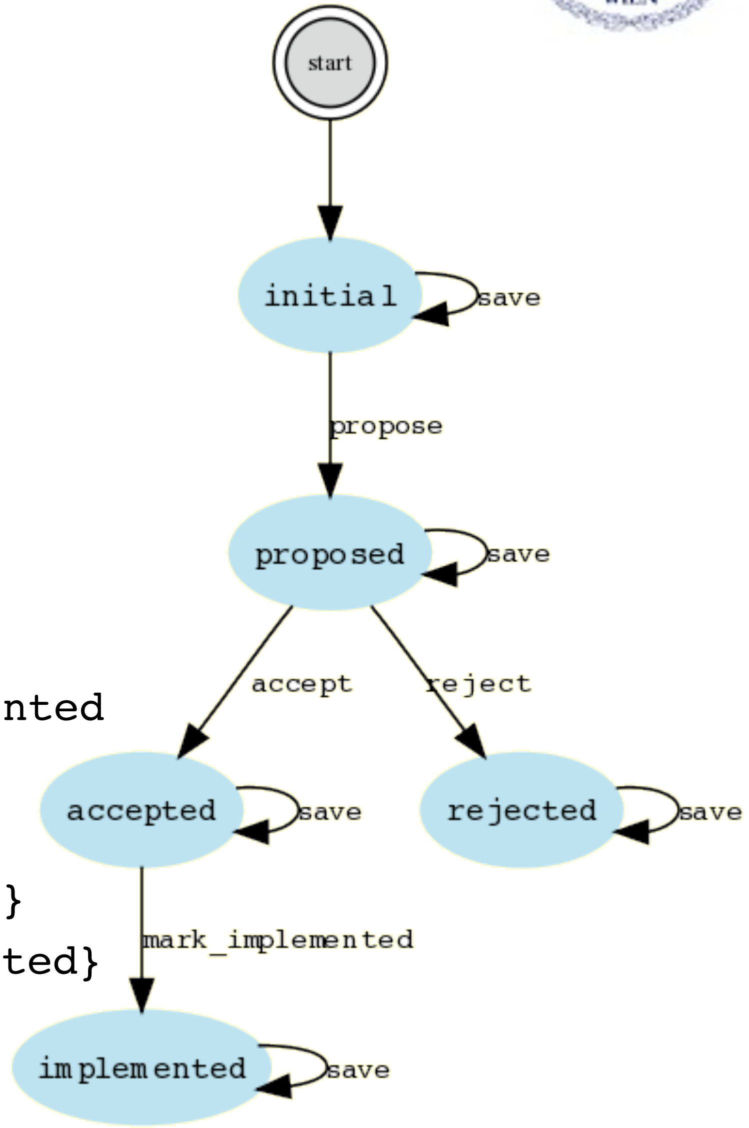
- Based on State Design Pattern
- Extended by “form” and “actions”
- Workflow Context is instantiated for every request

What’s new in Content Flow (2)

- Form selection possible based on state and context (e.g. user roles)
  - Same set of attributes can be presented to different users in different ways
  - Also application Objects with a single (explicit) state are more powerful than ::xowiki::FormPages alone
- Content Flows are defined in terms of
  - States,
  - Actions, and
  - Conditions

TIP Definition

```
set default_form "en:tip-form“

Action save -roles admin
Action propose -next_state proposed
Action accept -next_state accepted
Action reject -next_state rejected
Action mark_implemented -next_state implemented

State initial  -actions {save propose}
State proposed -actions {save accept reject}
State accepted -actions {save mark_implemented}
State rejected -actions {save}
State implemented -actions {save}
```

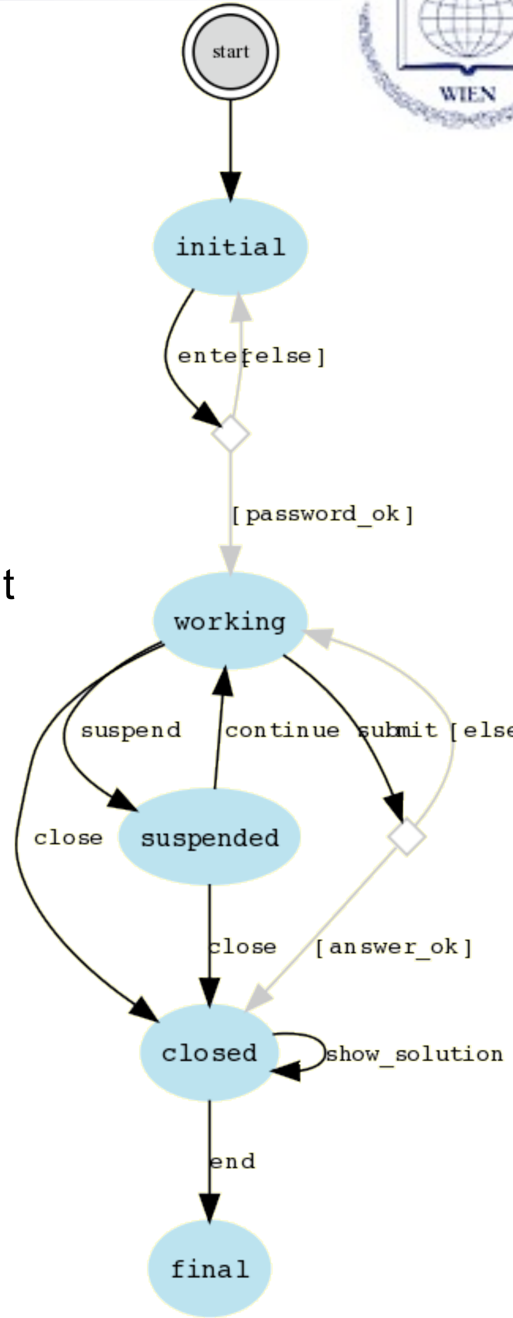


Content Flow with Conditions

- Workflow as defined by IMS QTI 2.0
  - Entering password,
  - allowing to suspend,
  - provide feedback

- Graph is automatically rendered via dot (graphviz) from Content Flow definition using named conditions

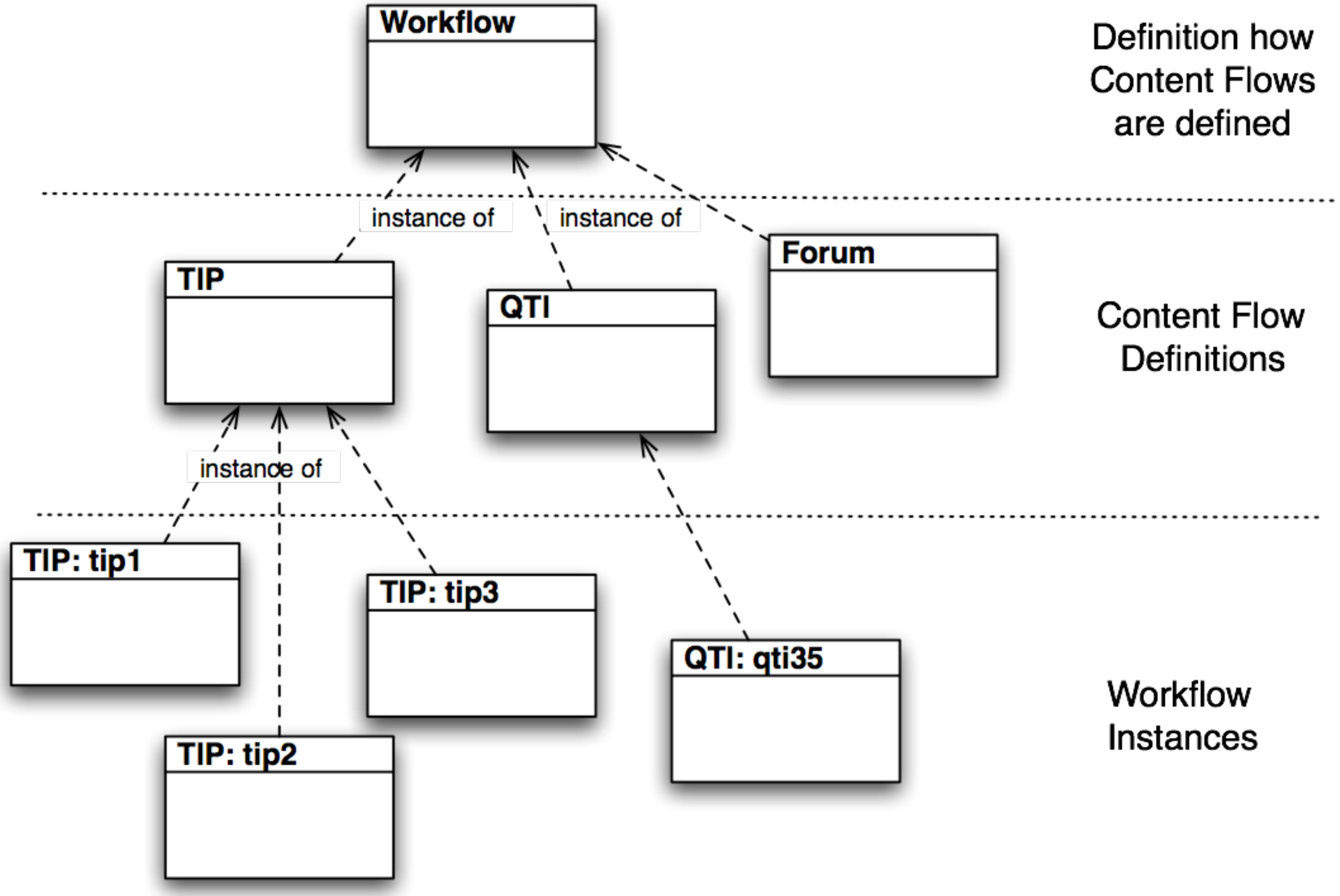
```
...
Condition password_ok \
   -expr {[my property password] eq "123"}
Condition answer_ok \
   -expr {[my answer_is_correct]}

Action enter \
   -next_state {? password_ok working else initial}
Action submit \
   -next_state {? answer_ok closed else working}

...
```



What’s new in Content Flow (3)

- Action can be individually programmed
  - ... within the content flow definition
- Actions can be called
  - ... via HTML FORM buttons, or
  - ... via method “call_action“ (e.g. via SOAP), or
  - ... via time scheduled activations
- Workflows are defined by an xowiki::Form with a form-field of type “workflow”
  - Definition of Content Flow and data model solely via Web
  - Content flow definitions are stored as ::xowiki::FormPages
  - Content flow applications can be exported via standard XoWiki export interface

Multi-Layered Definitions

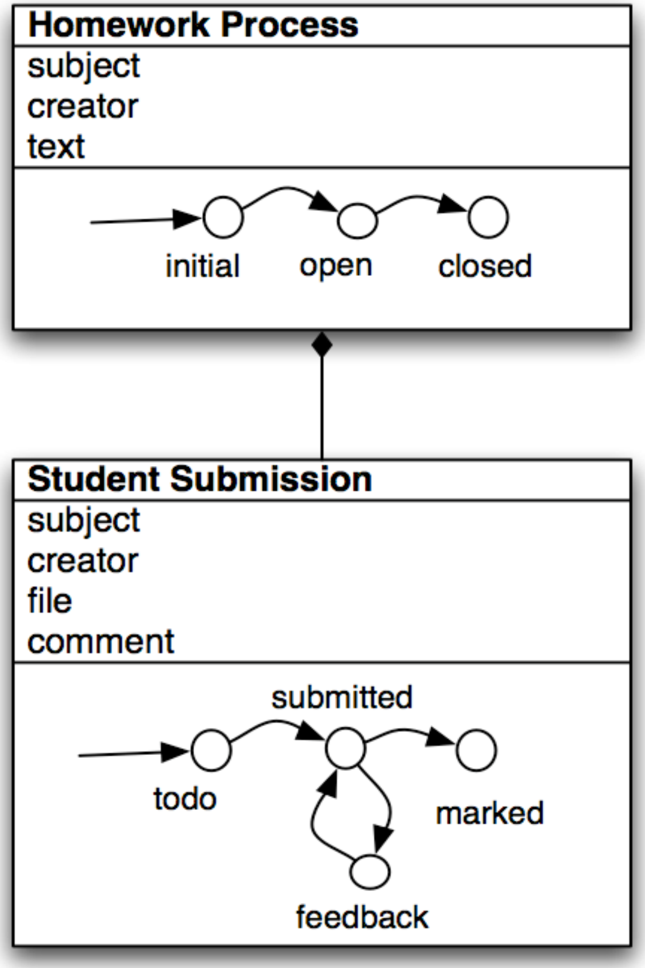


Definition how Content Flows are defined

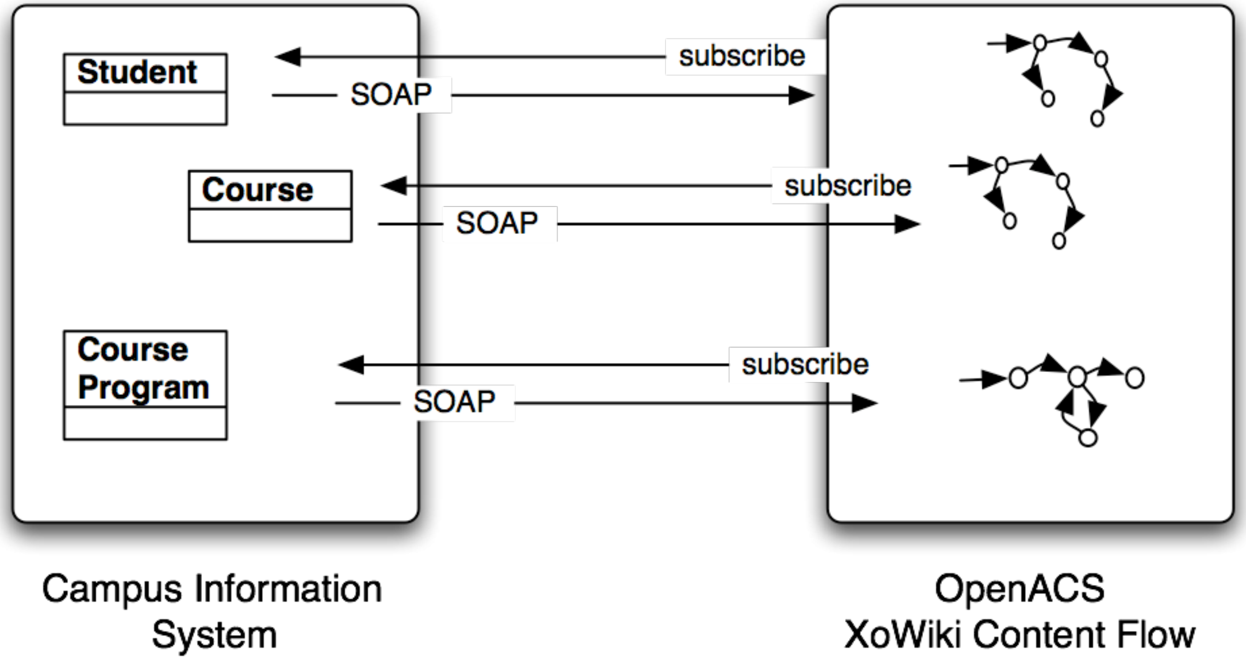
Content Flow Definitions

Workflow Instances

Nested Flows



Integration with Backend System
(Nima Mazloumi, Univ. Mannheim)



- OpenACS/DotLRN subscribes for “Events” (Changes)
- XoWiki Content Flow gets invoked via SOAP
- Workflows describe/document what happens in OpenACS/DotLRN, when a Course/... Is changed

Summary

- XoWiki Content Flow is a straightforward extension of ::xowiki::Form + ::xowiki::FormPage
- Revisions are used for storing workflow traces (possible to go back to an earlier state)
- XoWiki Content Flow inherits XoWiki properties
  - tagging, search, notifications, ..., collaboration graphs
- XoWiki Content Flow is implemented
  - as a sub-package of xowiki,
  - via a mix-in class (workflow behavior is mixed in)