



MonAtar: The Monash Teaching Avatar

Experiences

John Hurst

Clayton School of Information Technology
Monash University

20051104



Outline

Background and Introduction

The Concept

The Mechanics

The Lessons

The Opportunities

Conclusions



Outline

Background and Introduction

Teaching Quality and Accountability

The Teaching Quality Process at Monash

Evolution

The Concept

The Mechanics

The Lessons



Driving Forces on Teaching Quality

- ▶ The 1993-5 so-called "Quality Rounds" of university review
- ▶ emphases on Research, Teaching, and Community Engagement
- ▶ At Monash, creation of Associate Deans (Teaching)
- ▶ Heavy emphasis upon process, only later upon outcomes.
- ▶ Dramatic shift in attention paid to process: for teaching, it was course and unit approvals.
- ▶ Begs the question: what happens subsequently?



Post Quality Rounds

- ▶ Every faculty has an Associate Dean (Teaching)
- ▶ responsibility is the oversight of Teaching Quality
- ▶ Process is based upon "The Reader Over Your Shoulder"
- ▶ Every ADT reviews **other** faculties new courses, new subjects
- ▶ Process
 - ▶ Course/unit document in proforma *course description/unit description*
 - ▶ contains logistical and pedagogical information: objectives, relationship to others, workloads, academic responsibilities, etc..
 - ▶ roster of ADsT who review **every** proposal from **every** faculty
 - ▶ return approval, or negotiate changes
 - ▶ difficult cases may be decided at EdComm, but usually sent back for revision



Reality Check

- ▶ ADsT review outside their fields of expertise



Reality Check

- ▶ ADsT review outside their fields of expertise – but



Reality Check

- ▶ ADsT review outside their fields of expertise – but
- ▶ review for pedagogy, not content
- ▶ bunfights when demarcation involved as well!
- ▶ “quality” documents – not!
 - ▶ almost universally Word documents
 - ▶ exchanged by email
 - ▶ revised, re-revised, counter-re-revised
 - ▶ no audit trail
 - ▶ many inconsistent copies



The “Lead Site” Program

- ▶ The “Lead Site” program
- ▶ Each faculty asked to develop Teaching Quality theme for cross-dissemination
- ▶ InfoTech: *Exploiting our “Own” Technology in the pursuit of Quality*
- ▶ How can we use IT to support teaching quality?

Outline

Background and Introduction

The Concept

Principles

Opportunity

The Mechanics

The Lessons

The Opportunities



The Concept

- ▶ Need a single repository - database?
- ▶ But need broad access mechanisms
- ▶ But there must also be consistency and a certain amount of structure - a “semi-structured” document – the *unit description*, based upon existing Word document proformas
- ▶ XML was an emerging technology that seemed to be appropriate!
- ▶ Coupled with a web page was the next obvious step



Principles

Two guiding principles:

1. Academics should have the responsibility for, and control of, the content and delivery of all teaching materials.
2. There should be a single, up-to-date, accurate repository of all logistical and quality information relating to the delivery of course content.

Principle 1: Academic Responsibility

- ▶ The best person to judge how a unit is to be delivered is the person who delivers it
- ▶ The nature of the delivery should be transparent to all stakeholders
- ▶ Stakeholders are entitled to accurate and timely information about a unit
- ▶ All of these issues captured in a *unit description document*
- ▶ The quality of the teaching should be discernible from the unit description



Principle 2: Single Repository

- ▶ Far too often, data about units is continually entered and re-entered
- ▶ Example: booklists, software requirements, lecture rooms, tutorials, etc..
- ▶ Every year there is a mad scramble to “update” the handbook.
- ▶ The irony is, the information is usually out-of-date by the time the handbook is published!
- ▶ Want a “cradle-to-grave” management of unit descriptions
- ▶ want single-sourcing of all information – the “one-stop shop”

Principles: More

- ▶ unit descriptions should be publically accessible
- ▶ all *relevant* staff should have edit access to unit descriptions
- ▶ unit descriptions should capture all the relevant details about a unit
- ▶ details should be easily extracted for reports, web pages, etc.



Opportunity

- ▶ Discontent with existing system led to devolution of approval process
- ▶ from 2000, each faculty had authority to approve its own courses and units
- ▶ reporting line to university Education Committee to ensure quarantining of discipline areas
- ▶ Faculty of Information Technology could now run its own unit and course approval process
- ▶ Concept was to do this through an authenticating web page



Outline

Background and Introduction

The Concept

The Mechanics

The Unit Description

Security

The User Interface

Approvals

Paperless Meetings



The Unit Description

- ▶ XML document
- ▶ fields (elements) such as Unit Name, Unit Code, Reasons for Introduction, Unit Objectives and Content, Teaching Strategies, Assessment, Workload, School Responsibilities, Pre- and Co-requisites, Approval History, Edit History, ...
- ▶ many of these have subfields (e.g., Assessment Strategies).
- ▶ at bottom level, elements are unstructured text fields, and can contain HTML - but it must be well-formed!
- ▶



Security

- ▶ Initially, every document open to editing by any person
- ▶ All users must authenticate before editing, through university LDAP system
- ▶ audit trail identifies who edited what
- ▶



The Edit Process

- ▶ Two classes of editor: novice and expert
 1. novice edits structured text
 2. expert edits full XML
- ▶ system automatically renders between structured text (as displayed), and XML (as stored)
- ▶ structured text allows paras, bullet and numbered lists, bold, italic, and a few other simple markups
- ▶ inner content XML elements passed through unchanged



Approvals

- ▶ Each new or revised unit must be approved
- ▶ Levels were (since changed):
 1. School
 2. Faculty Education Committee
 3. Faculty Board
- ▶ Approval at all levels handled semi-automatically
- ▶ Each level has a sign-off by an authorized and authenticated user
- ▶ report available to show where in approval process a proposal lies



Paperless Meetings

- ▶ Since all documents, in all versions, are kept on-line, on-line access in meetings is an obvious choice
- ▶ FEC went on-line in mid-2000, FB in late 2001
- ▶ easy transition to generate HTML web-accessible agendas and minutes



Outline

Background and Introduction

The Concept

The Mechanics

The Lessons

Open Access

Approvals

Report Generation



Open Access

- ▶ Some user resistance to everyone being able to edit
- ▶ Concept of school ownership, rather than faculty
- ▶ Incident where offshore academics edited one of their offerings - clashed with originating school!
- ▶ open access changed to restrict edits to members of originating school
- ▶ (since changed back!)



Approvals

- ▶ Hard-coded: not a good idea!
- ▶ Interest in deployment by other faculties foundered on approvals model
- ▶ Wide variation in approvals processes
- ▶ suggestion that the approvals work-flow be decoupled
- ▶ need more “round-tuits”!



Report Generation

- ▶ semi-structured XML documents allow report generation from arbitrary elements
- ▶ examples:
<http://bendigo.csse.monash.edu.au/~ajh/adt/resources/lablist.html>
<http://bendigo.csse.monash.edu.au/~ajh/adt/resources/booklist.html>
- ▶ However, these are not currently generated on-line on demand - should be!



Reengineering?

- ▶ System has already been re-engineered once (to solve big security hole!)
- ▶ Is it time to do it again?
- ▶ Moved from personally maintained system to official faculty web site
- ▶ deployment in other contexts would certainly warrant this effort



Outline

Background and Introduction

The Concept

The Mechanics

The Lessons

The Opportunities

Serendipity



Serendipity?

The advent of the avatar has prompted some interesting ideas:

- ▶ Interfacing to other university systems: Callista, CUPID
 - ▶ Have an API model
 - ▶ needs more “round-tuits”!
- ▶ Automatic teaching portfolios
 - ▶ Data entered by individuals is authenticated
 - ▶ Data represents teaching initiatives by these individuals
 - ▶ generate *authenticated teaching portfolio* reports on these initiatives
- ▶ Course maintenance: auto-admin support for students
 - ▶ Honours student: XML framework for course specification
 - ▶ gives avatar style development for courses
 - ▶ Auto-tailored Prolog programs to assist students with course choices and enrollment



Outline

Background and Introduction

The Concept

The Mechanics

The Lessons

The Opportunities

Conclusions



Conclusions

- ▶ Certainly transformed the way in which InfoTech now processes teaching developments
- ▶ Crunch was the transition from the old faculty structure to the new one (2005).
- ▶ Much of the process and context has changed, but the avatar has been retained.
- ▶ Teaching Quality: is it really improved?
 - ▶ I think **yes**, but I am biased!
 - ▶ Administrators very supportive
 - ▶ Reason avatar was retained in new faculty structure was by popular demand of the academics who were using it!



And the name *avatar*?

- ▶ Comes from the Hindu, meaning *an incarnation, embodiment, or manifestation of a person or idea*.
- ▶ In computing, means *a movable icon representing a person in cyberspace or virtual reality graphics*.
- ▶ Some people have criticised the term, presumably because of the lack of an “icon”.
- ▶ My original joke was that it (the program) was the manifestation of the ADT, and his power of controlling teaching quality.
- ▶ The irony is that the ADT has gone, but his manifestation lives on!



And the name *avatar*?

- ▶ Postscript: someone (unknown?) claimed that the word *avatar* is offensive to Muslims, and the program has been (re)named to *MonAtar* (contraction of Monash Avatar).
- ▶ I could find no evidence of this ...

These Slides ...

Are available from:

`http://www.csse.monash.edu.au/
~ajh/research/seminars/avatar-swin.pdf`