

Extending OLAT to Support Operative Administrative Processes on the Level of a Department

Talk at the OLAT Conference 2008, Zurich

Hans-Gert Gräbe, Univ. Leipzig, Germany

<http://bis.informatik.uni-leipzig.de/HansGertGraebe>

About the Software Engineering (SWE) Curriculum in Leipzig

- Software Engineering (SWE) projects as integral part of the c.s. Curriculum
- Projects (sem. 4) start if Programming (lecture and labs, sem. 2) and SWE courses (lecture and labs, sem. 3) are completed.
- Groups of 6 .. 8 students work on such a project during the summer term = about 1000 h programming load. Several groups work on the same theme. Projects cover the whole software cycle planning, requirement analysis, design, modelling, implementation – The proof of the pudding is the eating.
- Focus on reusable software to support the needs of the department. The outcome cannot be used directly but requires consolidation (winter term)
- Projects realized so far
 - Uebmanager = standalone web application to support SWE labs
 - Elate-Portal = web portal with main focus on E-Assessment.
Used in online exams for about 800 students within the department of education

SWE and OLAT

- In the summer term 2007 we first time offered an OLAT-based theme: Develop an OLAT extension to handle exam registration and results reporting
 - Involves interaction with the administration (exam office) and thus requires an extension of the roles offered by OLAT
 - Consolidation as **Xman project** in the winter term 2007
- In winter term 2007 we set up (in parallel to our “legacy system”) an OLAT production instance <http://olat.informatik.uni-leipzig.de> as IT-support on department level.
- XMan was integrated upon completion and tested on a variety of exam registrations. Full integration into OLAT head depends upon certain adjustments of extension points in the rights system.
- Another project on the way is the integration of the Elate test engine into OLAT as another course resource.

About LMS Design

How a good LMS should be designed?

- How to get an answer?
 - Start and develop one ...
 - Poirot's method: lay back and think ...
- Software requirements analysis: Identify real life processes, stakeholders, process dimensions and process owners, real need for IT support, address acceptance problems
- IT based? IT driven? IT supported?
 - Techno-social aspects
 - "Code is law" (Lawrence Lessig, www.code-is-law.org)

What's about OLAT?

- OLAT is primarily designed to support creation and delivery of courses. Typical for a today's LMS it unites
 - an authoring tool
 - a delivery engine
 - rudimentary archiving functionality

But these areas are usually separated in business IT systems

(1) Authoring tools:

- Document specific authoring software
- Main focus on the interoperability of documents, not software
- Breakthrough with upcoming XML-based standards

(2) Delivery engine:

- Operative data management: detailed data, no or low time dimension
- Contains the most important part of the business intelligence, focus on technical parameters
- Uses enterprise specific or domain specific solutions

(3) Archiving tools:

- More complex requirement: Dispositional data management
- Consolidated data, deep time dimension, focus on commercial success
- Uses standard data warehousing concepts, architectures and tools, e.g. SAP business warehouse

What's about university IT systems?

(1) Authoring tools:

- There exist plenty of widespread tools, in CS mainly ppt-, pdf- and LaTeX based

(3) Dispositional data management:

- Widespread IT-support of management processes at university level
 - Students' enrolment and achievements profile management
 - Employment management for staff and student assistants
 - Annual reporting to external authorities

(2) Operative data management

- island solutions for IT-supported courses and training
- low level of penetration of IT-support for complex operative administrative tasks
 - even for bundled course assignment with side conditions
- low budget vs. high expectations tend to **centralized solutions**, regardless of process dimensions and process owners (e.g. BPS, OLAT Service at Zurich University)

1. Centralized solutions do not reflect the flexibility of requirements posed by the granularity of process dimensions and owners
 - e.g., the affairs handled in our exam office are 90% standard but 10% require flexibility of decision meeting that could not reliably mapped into algorithms.
 - IT support should not substitute administration but concentrate on the “all day business” to save administration time for the really hard cases that require human interaction.
2. Centralized solutions tend to map a great variety of requirements from different sources into a single piece of software
 - It has to solve conflicts between requirements even if they come from different sources and do not occur in real life
 - **Alternative:** Set of communicating specially configured instances of the same software or special software(s) assembled from a set of components by a common design principle (component framework)
 - Requires clear distinction between software and software instances.

Where to start with such an alternative?

Operative administrative processes on the level of a department

- (Yet) the main level of operative academic decision meeting
- Sufficiently explicit rules and actors' experience with real life processes
- Sufficiently homogeneous to be modeled consistently

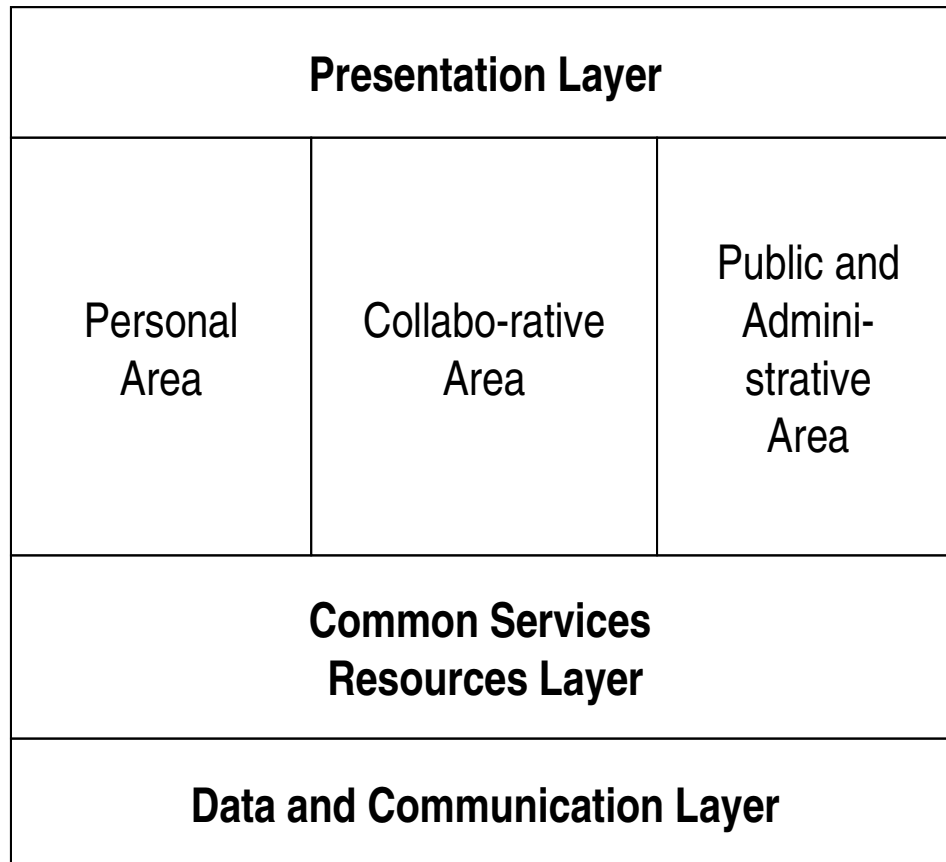
Claim 1: Operative administrative processes on the level of a department are characterized by quickly and unpredictably changing requirements. In particular, the practical consequences from modularization and the Bologna process are hardly to predict in detail yet.

Claim 2: With regard to the financial situation in general and the discussion about financial equipment of academic institutions in particular one cannot expect generous financial support for development and operation of such IT structures.

Design 1: Use cases and actions viewpoint

Learners' Profiles	Course Enrolment and Groups	Admini- stration
Personal Area	Cooperative Area	Public Area

Design 2: Layer and Block Structure



Provides unified access

Business Layer

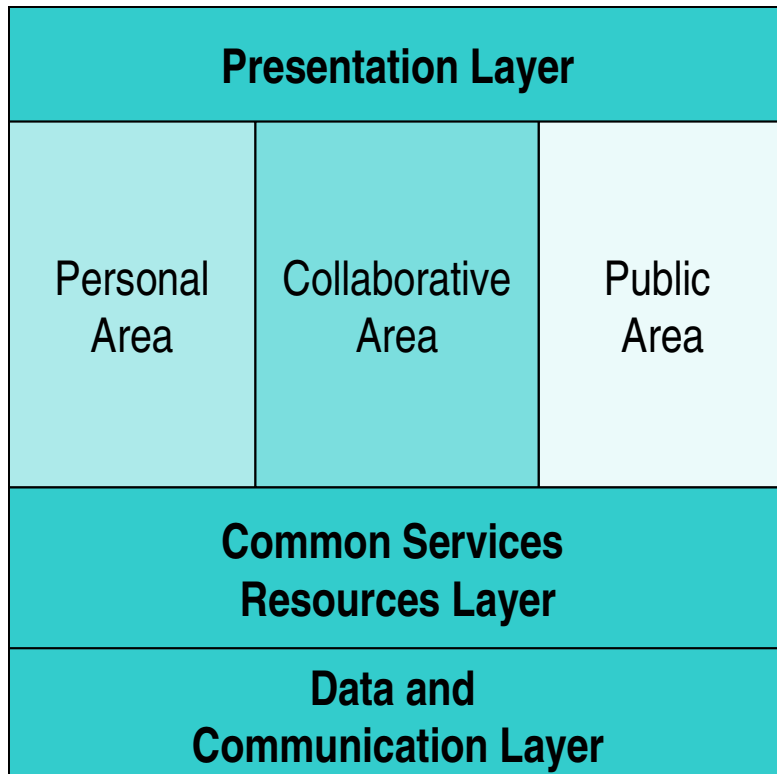
Arrange tools instances according to special needs

Provide tools

Persist data and provide communication

Some Thoughts about Design

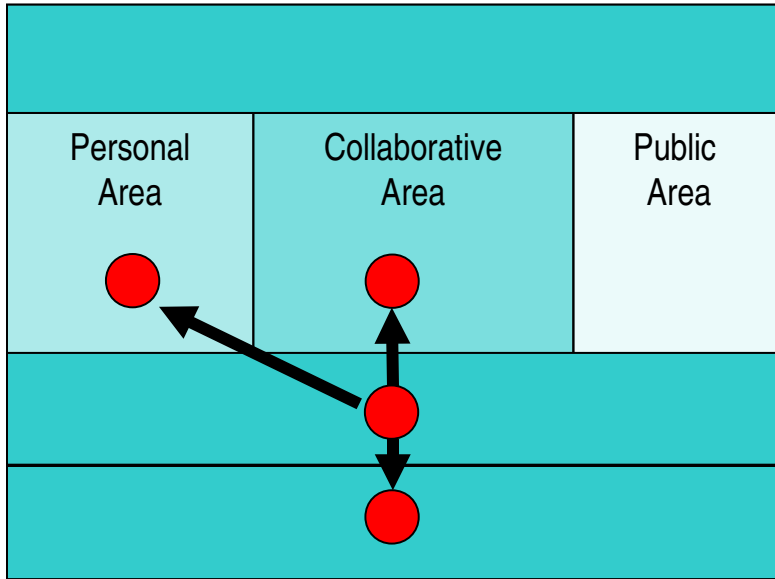
What's about OLAT?



Personal area: contains predefined information with statically fixed role dependent exposure mode to other users.

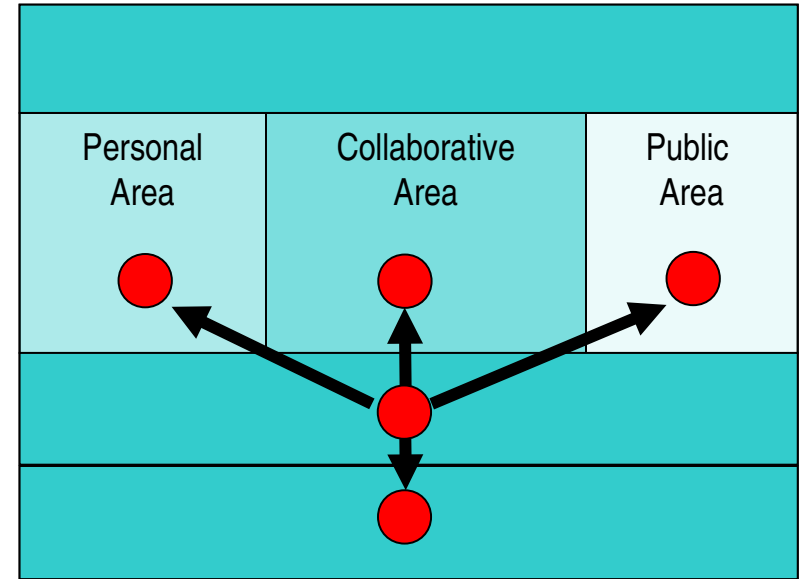
Collaborative area: contains two predefined modes of collaboration (courses, working groups) that can be equipped with different resources. Not (yet) clearly modularized.

Public area: Rudimentary tools for course schedule display (catalogues)



Elate-Project

Add another tool for
E-Testing to OLAT



XMan

Add another collaborative mode for exam
assignment and results reporting to OLAT

Roles or Rights?

Cooperate Scenarios – Roles or Rights?

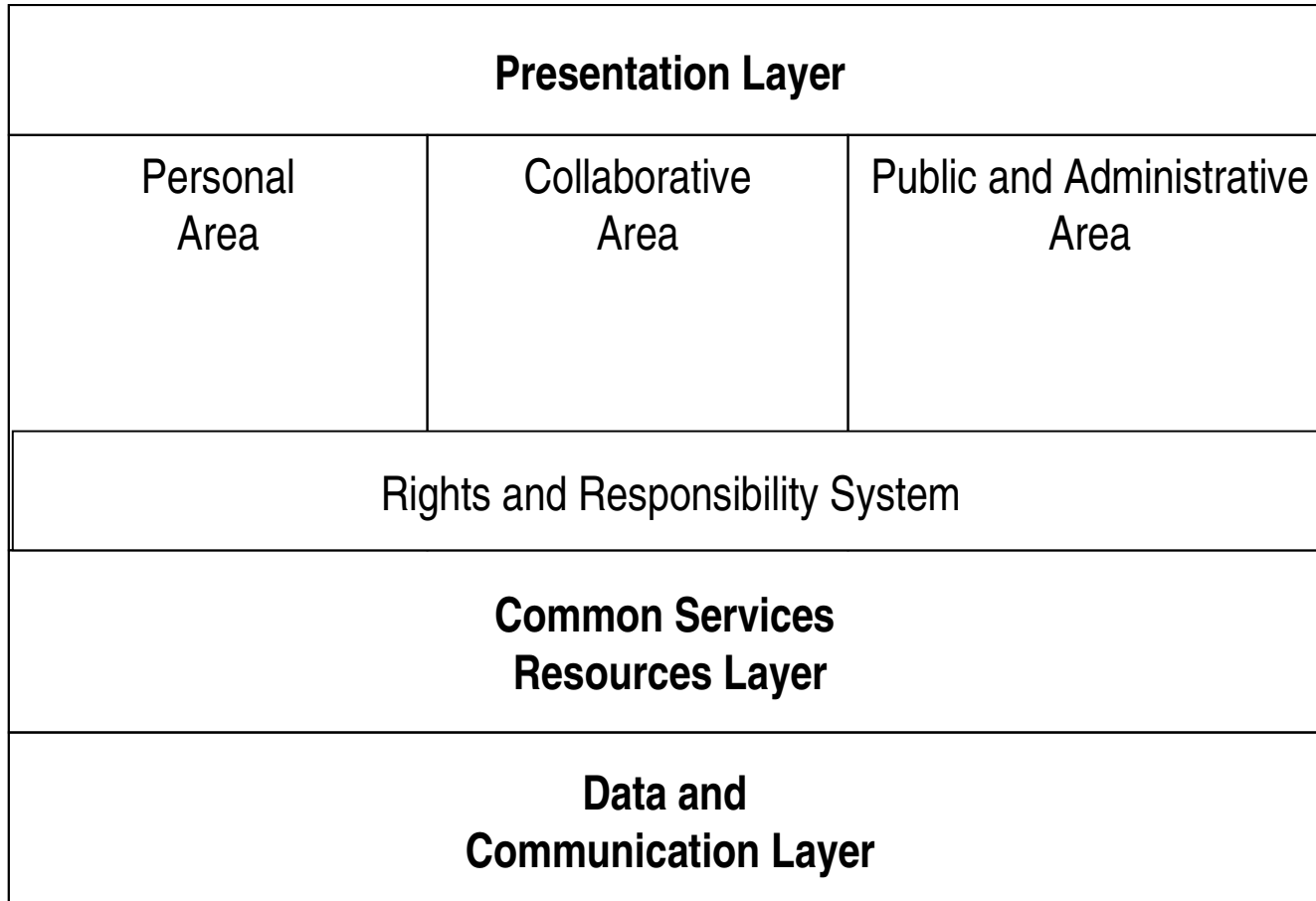
old OLAT is role based

- Classical approach, static rights concept, hard coded
- Easy implementation, roles can be modeled, hard to extend

new OLAT is rights and responsibility based (SecurityGroups)

- Dynamic approach to set up collaborative contexts at run time
- Roles can be designed and added within extensions or even during runtime
- Distinguish between **rights design** (at development time), **rights definition** (at project setup time) and **rights execution** (at project realization time)
- Contractual structure of our responsibility based society:
rights design = legislator, rights definition = contract agreement, rights execution = contract fulfillment
- Execution of rights is encompassed with taking responsibility, e.g.,
 - A student enrolled into a course has to take the exam
 - A staff member offering a course has to provide exams
- Best suited for operative processes with rapidly changing requirements

Design 2a: Special Role of a Rights and Responsibility System



Requirements

Personal area provides

- Flexible extension concept for profiles
- Secure identity management including a private data store
- Flexible concept to expose part of the private data to collaborators and the public

Collaborative area provides

- Setup for different types of collaboration contexts
- Scalable rights based access to its resources, including a project data store
- Flexible concept to expose part of the project data to the public

Public and administrative area provides

- organized public display of categorized information extracted from the public part of the operational data store
- data exchange interface to external applications, in particular to dispositional data stores