



Weakly-structured Workflows for Knowledge-intensive Tasks: An Experimental Evaluation

L. van Elst, F.-R. Aschoff,
A. Bernardi, H. Maus, S. Schwarz

*German Research Center for Artificial Intelligence (DFKI)
- Knowledge Management Department -*

WETICE KMDAP, 10.6. 2003

Agenda

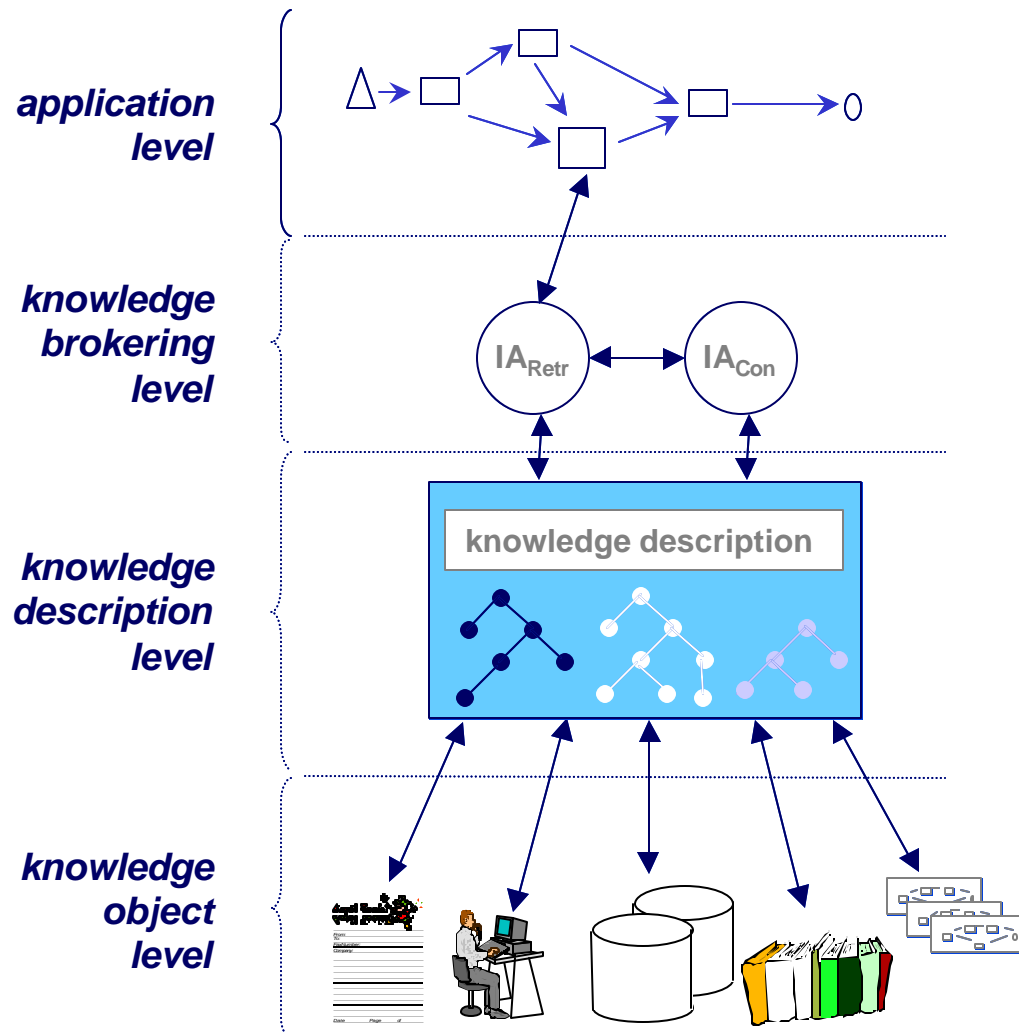
- Concept: Weakly-structured workflows for BPOKM
- Design of experimental evaluation
- Experimental results

Agenda

- Concept: Weakly-structured workflows for BPOKM
- Design of experimental evaluation
- Experimental results

Background/History: KnowMore (Abecker et al., 1998)

Framework for Organizational Memory Information Systems



- Knowledge workers are involved in complex processes
- Process models and their enactment provide context information and facilitate proactivity
- Ontologies are the explicit basis for the knowledge-level description
- Access to various information sources relies on formal knowledge-item descriptions

Knowledge management addresses context-specific, proactive delivery of information

FRODO creates a conceptual and technical framework to build a distributed OM for knowledge-intensive work

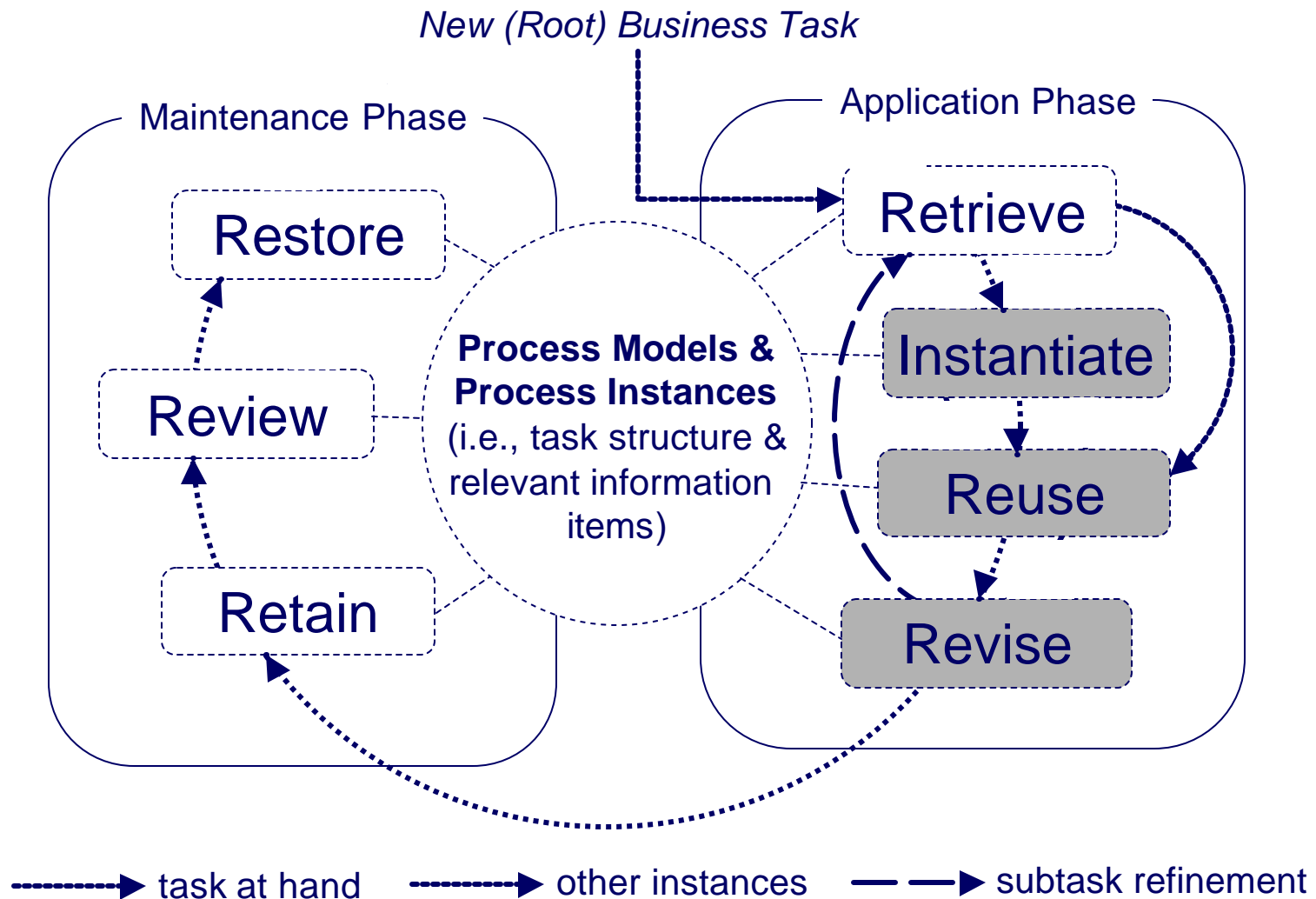
- From **central** to **distributed** Organizational Memories
 - A monolithic, central OM is *seldom feasible*
 - Central OMs neglect the reality (and opportunities) of the *distributed nature of knowledge* in companies
 - ➔ The FRODO framework facilitates *societies of cooperating agents* as a basis for distributed organizational memories, especially
 - distributed ontology management and
 - cooperative specialists for information retrieval
- From **strict** process models to **weakly-structured** workflows
 - Traditional models for business processes are ill-suited for knowledge-intensive work
 - Users are ignored as source for process knowledge
 - ➔ FRODO supports complex, dynamically configured processes

Knowledge-intensive work processes are modelled by weakly-structured workflows

- **A weakly-structured workflow consists of knowledge-intensive tasks**
 - which are not necessarily defined a-priori
 - but abstract structures are usually known
- **Weakly-structured workflows evolve over lifetime**
 - they can - but need not to - follow the structure of underlying models
 - exact structural repetitions are seldom, but there is a multitude of 'similar' tasks
 - workflow instances can be modeled and refined during runtime
i.e., lazy modeling, late modeling
 - hierarchical refinement of tasks leads to structured workflow instances
- **Workflow instances represent valuable process knowledge**
 - tasks are worth to be supported and preserved (e.g., for best/worst practices)
 - typical instances are generalised to 'templates', i.e., task models

FRODO TaskMan exhibits the core functionalities needed to support knowledge-intensive work

Life- and use cycle of process models can be described as variant of a standard case-based reasoning cycle



Agenda

- Concept: Weakly-structured workflows for BPOKM
- Design of experimental evaluation
- Experimental results

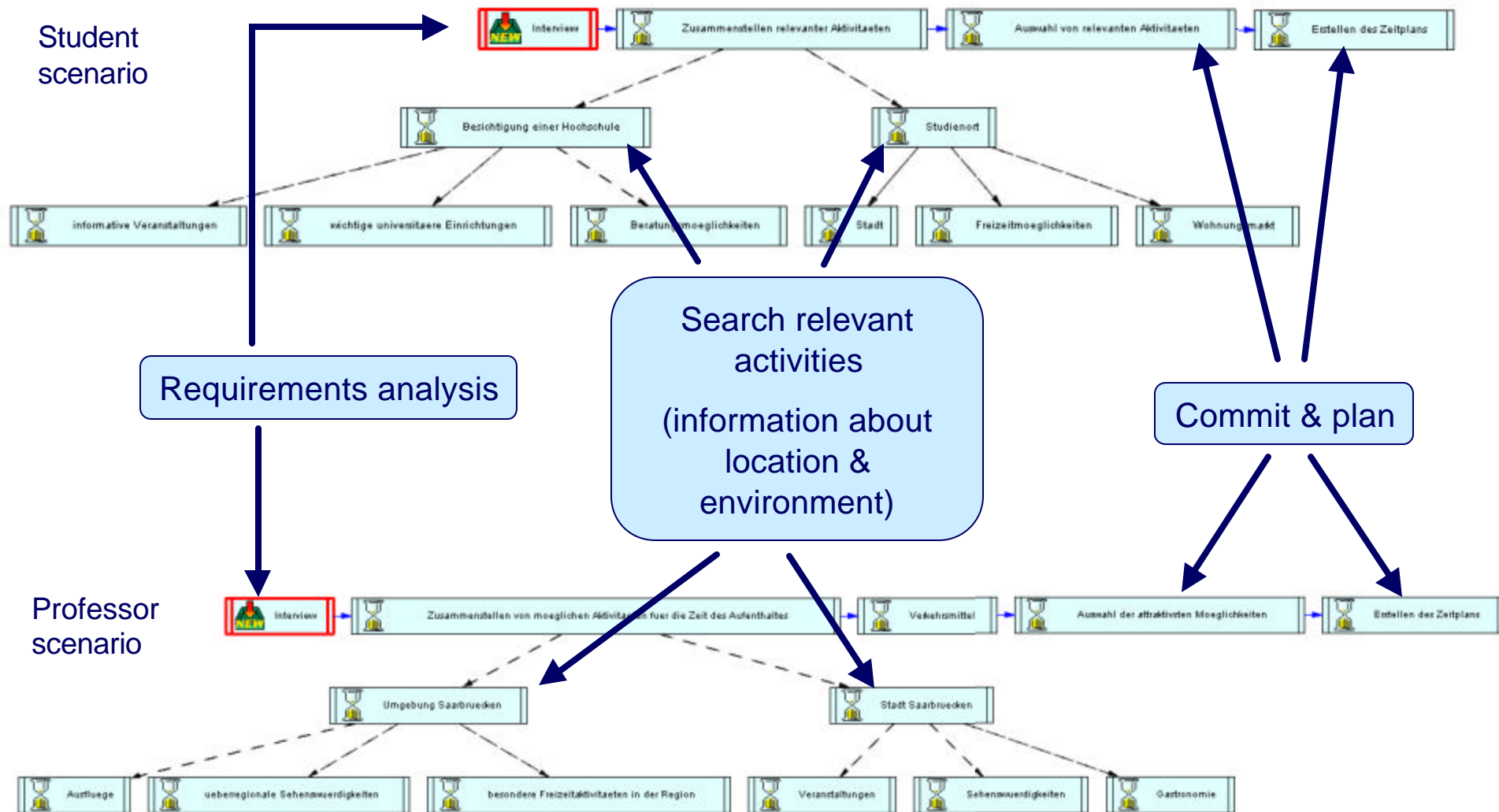
Of course, there are a lot of questions to such an approach

- Are weakly-structured workflows a useful basis for support of knowledge-intensive activities?
- Is integration of process execution and information support accepted and considered of benefit?
- Is process-embedded information a means of knowledge sharing and transfer?

We chose visit planning as the knowledge-intensive activity to be supported

- **Scenario 1:** Prepare a visit for a prospective *student* to get familiar with the future work/living environment
- **Scenario 2:** Prepare a visit for a guest *professor* at the university, including tourism aspects
- Characteristics of the scenarios:
 - They include room for flexible interpretation
 - Ultimately, a time schedule needs to be produced
 - They can be understood and processed in reasonably short time
 - There is plenty of relevant information in the web
 - Both scenarios are “structurally equivalent”

Both scenarios lead to structurally similar initial workflows



In the evaluation, subjects used a restricted version of the FRODO TaskMan

- Expressive power of the workflow language:
 - hierarchical decomposition
 - sequential dependency
- Dynamic refinement of workflows at execution time
 - integration of modeling & enactment
- Enriching workflow tasks with information needs
 - dynamic & static
- Support of task execution by linked information items
 - context-specific information support

The user can add and use relevant information items in the workflow context

- choosing **concepts from domain ontologies**
 - study and tourism ontologies
 - used as task annotation and for searching the web
 - also adding keywords
 - search the web with given concepts/keywords
- adding **relevant web pages**
 - explicitly and by surfing the web
- adding **memos**
 - resemble notes, remarks, results for a specific task
- investigate task information
 - provided in a browser
 - also inherited (i.e. from the task hierarchy) information is shown

Eval - UserManager

Console Worklist Tree **Graph**

Task in progress

Sub-Task (finished)

current task

Work control: model, start & finish

Modelling Put back Commit Revert Finish Assistant Back Fwd

Path: Study arrange relevant activities visit of an university guidance posib

Overview Description **Browser** Information ProcessRole

Search Engines Search

RelevantInformation

Task Info Browser:
Information Items attached to the current task

- [\(HtmlDocument\)](#) Mathematics at Munich University [link](#)
- [\(HtmlDocument\)](#) University Munich [link](#)
- [\(HtmlDocument\)](#) TU Munich [link](#)
- [\(HtmlDocument\)](#) independent information about the university of Munich [link](#)

inherit
visit

arrange relevant activities
Study

- [\(Memo\)](#) schedule for the visit
>result of Eval: The schedule is as follows

Workflow : Study

The Experimental Design Comprised Four Factors

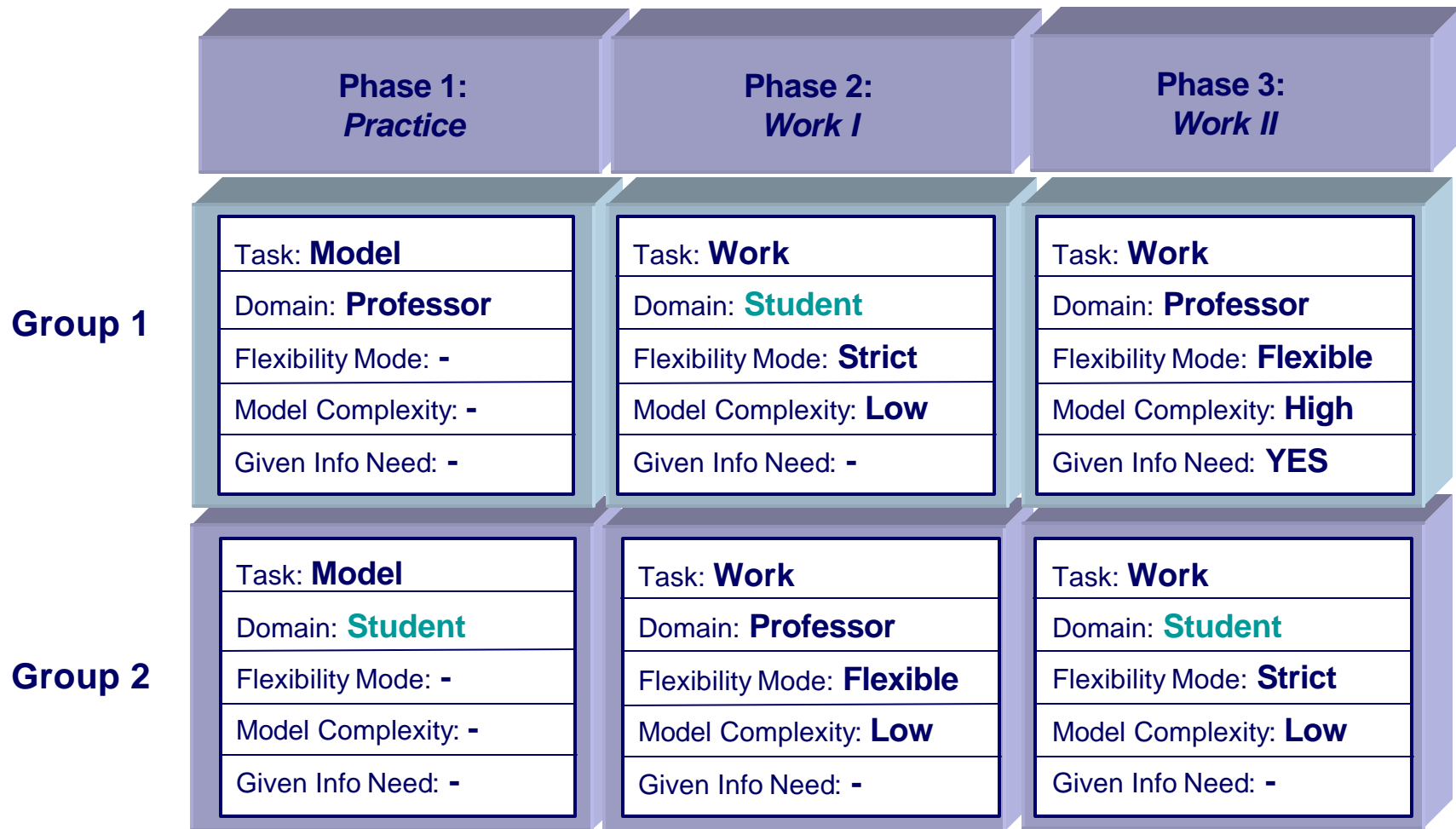
- Domain: Preparing a schedule for a professor in Saarbrücken
vs. for a student in Munich
- Flexibility mode of the workflow: Strict vs. weakly-structured
- Complexity of given workflows: “small” vs. “big” model
- Re-use of information: With/Without Given Information Items

A full (2x2x2x2)-factorial design was not feasible.

This led to a restricted design with two experimental groups and the following features:

- Both groups process both domains and both flexibility modes.
- Complexity and re-use of information is tested only in one domain and with flexible workflows (between groups).

The experiment comprises the model-work-refine phases of the workflow lifecycle



Four hypotheses are investigated in the experiment

- Knowledge workers feel better supported with late/lazy modeling facilities
- Lazy/late modeling in weakly-structured workflows leads to a more precise classification of information items than strict workflows
- Proactive information support is (demonstrably) useful
- Weakly-structured workflows are better suited than strict ones to deal with unexpected task situations

The evaluation trials gathered direct and indirect measurements

- **direct measurements:** The test persons were asked for subjective assessments via questionnaires
- **indirect measurements:** data collected during the experiment were evaluated
 - modified workflow models
 - attached information items
 - web logs, representing search activities

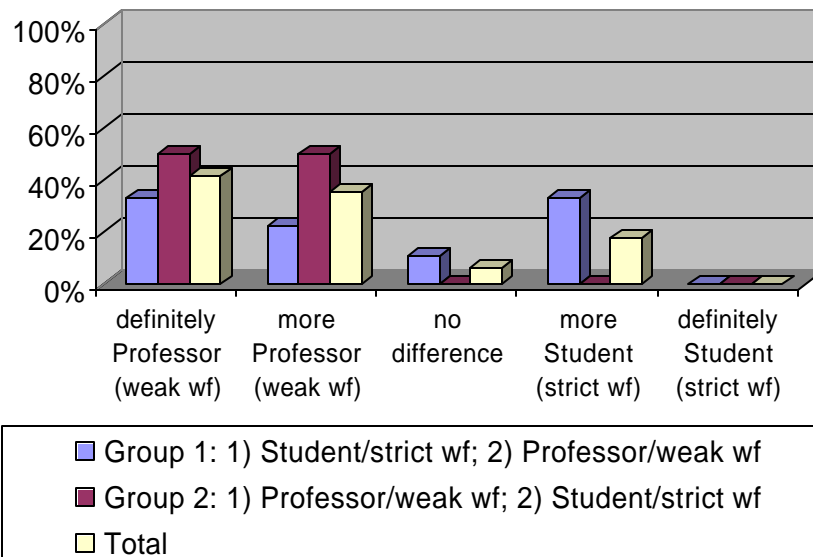
We performed 5 trial runs with 25 students in total.
The first run was considered a pre-study.

Agenda

- Concept: Weakly-structured workflows for BPOKM
- Design of experimental evaluation
- Experimental results

Evaluation of the questionnaires proves: Knowledge workers feel better supported with late/lazy modeling facilities

In which domain did you feel more flexibly supported?

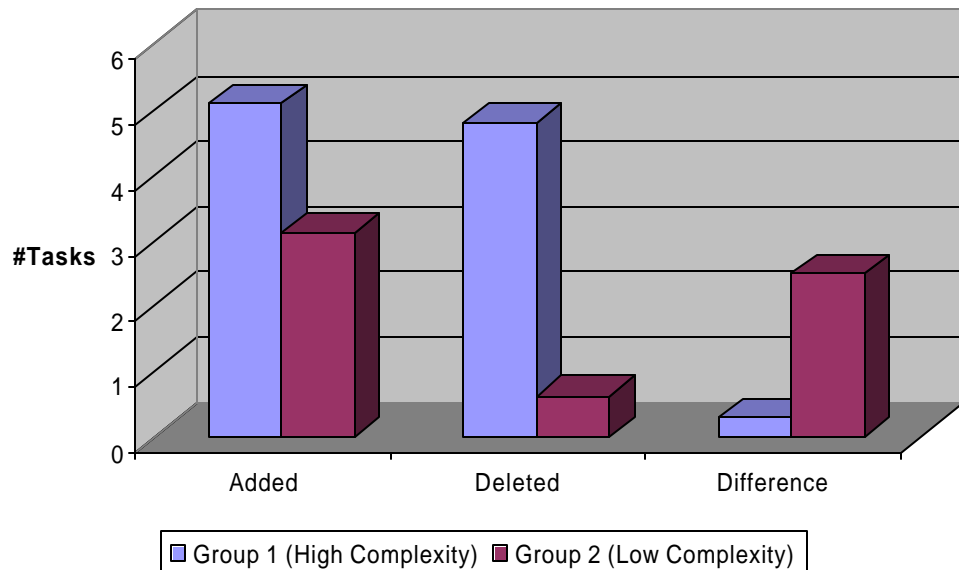


- All groups appreciate the flexibility of weakly-structured workflows
- Appreciation is significantly higher in the groups who started with the weak workflows
 - people appreciate the benefits after they lose them

The inverse question gave an identical result.

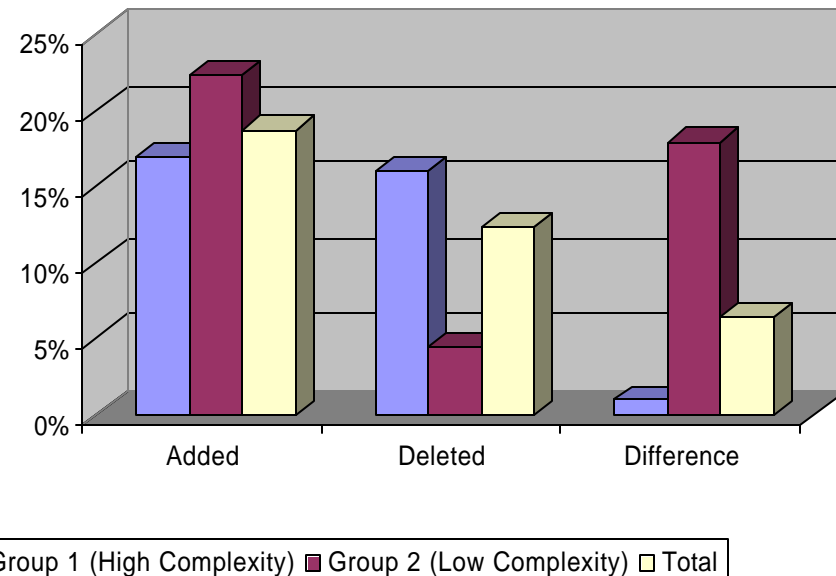
Analysis of the modified workflow models shows: Dynamic modeling is used intensively.

Late Modeling Behaviour



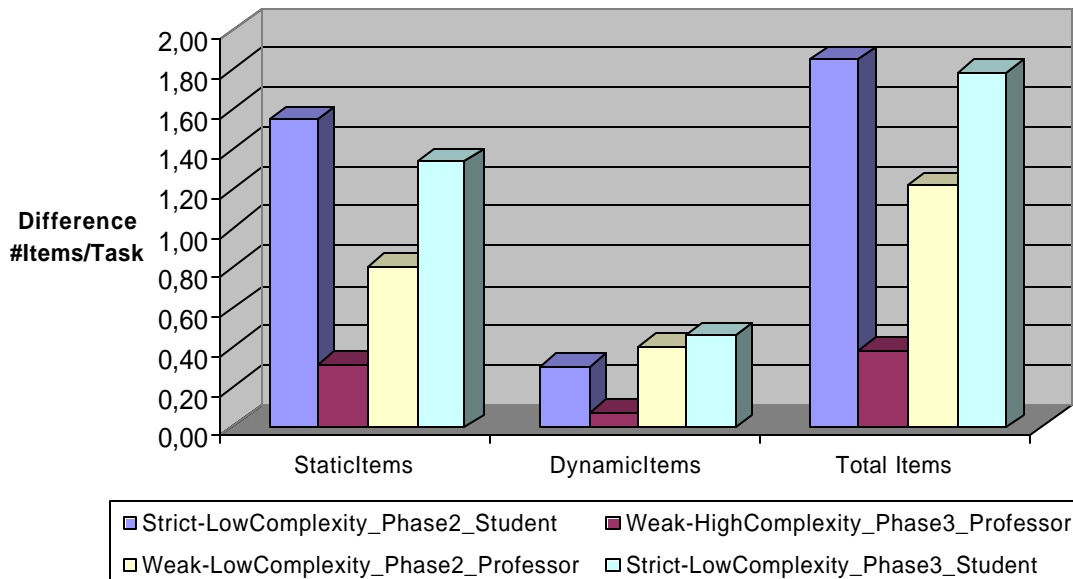
#Tasks/
Given Model
Size

Late Modeling Behaviour



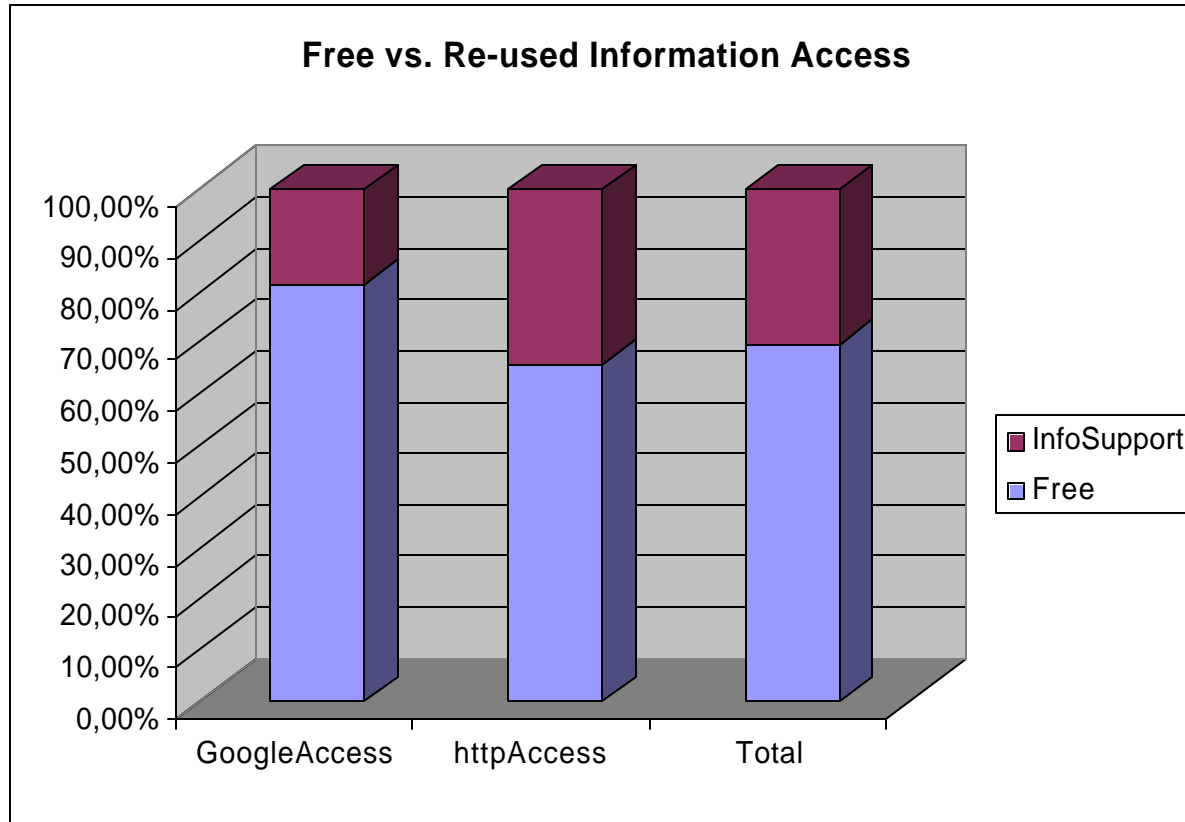
Analysis of enriched workflow models demonstrates: Lazy/late modeling in weakly-structured workflows leads to a more precise classification of information items than strict workflows

Difference InformationItems per Task



- Weak workflows result in less information items per task
- this is interpreted as a more precise classification

Proactive information support is (demonstrably) useful



Evaluation of web access logs shows:

- About 30% of all information access result from pre-given information
- ➔ User profit from pre-given information elements
- ➔ Nevertheless, additional information sources are visited

Weakly-structured workflows are better suited than strict ones to deal with unexpected task situations

- After 1 hour's work, an additional task was introduced in each condition:
 - Weakly-structured workflow: “The professor indicates that his wife intends to do sightseeing & wellness. Check possibilities and make relevant suggestions”
 - strict workflow: “The student intends to earn money by giving music instructions. Check possibilities and contacts”
- The analysis of the work results shows the advantage of weakly-structured workflows:
 - Weakly-structured workflow: The task is integrated into the process
 - Strict workflow: The task is
 - partially ignored
 - wrongly classified

Unexpected task in a weakly-structured workflow: Sound integration of the additional task and information related to it.

The screenshot displays the 'Eval - UserManager' application interface. The main window is divided into two panes. The left pane shows a workflow diagram for 'Tourismus_Vorgabe'. The diagram starts with a task 'von moeglichen Aktivitaeten fuer die Zeit des Aufenthaltes' (from possible activities for the duration of the stay), which leads to a 'Verkehrsmittel' (vehicle) task. Below this, the workflow branches into 'Saarbruecken', 'Stadt Saarbruecken', 'Gastronomie', 'Wellness', and 'Sehenswuerdigkeiten'. The 'Wellness' task is highlighted with a red box, and a sub-task 'Erlebnisbad' (experience bath) is shown below it. The right pane contains a toolbar with icons for 'Modelling', 'Put back', 'Commit', 'Revert', 'Finish', 'Assistant', 'Back', and 'Fwd'. Below the toolbar is a 'Path' bar showing the current path: 'Tourismus_Vorgabe' > 'Zusammenstellen von moeglichen Aktivitaeten fuer die Zeit des Aufenthaltes' > 'Stadt'. The right pane also has tabs for 'Overview', 'TaskConcepts', 'Description', 'Browser', 'Information', and 'ProcessRole'. The 'Information' tab is active, showing a 'View InfoSupport' button. Below this is a section for 'relevant information' with a red header 'Memo (remark) adresse'. A 'Memo' window is open, showing the title 'adresse', the author 'author of the memo', and the content 'Calypso', 'Deutschlental 7', and '66117 Saarbrücken'. The 'remark' field is empty. The bottom status bar shows 'Workflow : Tourismus_Vorgabe'.

Workflow : Tourismus_Vorgabe

Unexpected task in a strict workflow: Relevant information is (inadequately) linked to 'town'

The screenshot displays the 'Eval - UserManager' application interface. The main window is divided into several sections:

- Top Bar:** Contains tabs for 'Console', 'Worklist', 'Tree', and 'Graph'. The 'Graph' tab is active.
- Workflow Graph:** A flowchart showing tasks. The 'Stadt' task is highlighted with a red border. The graph shows a sequence of tasks: 'Zusammenstellen relevanter Aktivitaeten' (checked) leads to 'Auswahl von relevanten Aktivitaeten' (checked). Below these, 'Besichtigung einer Hochschule' leads to 'Studienort' (checked). 'Studienort' then branches into 'Stadt' (checked and highlighted), 'Freizeitmoeglichkeiten' (checked), and 'Wohnungsmarkt' (checked).
- Right Panel:** Contains a toolbar with icons for 'Model', 'Work', 'Commit', 'Revert', 'Finish', 'Assistant', 'Back', and 'Fwd'. Below the toolbar is a 'Path' bar showing 'Studium_Vorgabe' > 'Zusammenstellen relevanter Aktivitaeten' > 'Studienort' > 'Stadt'. The 'Information' tab is selected, showing a list of 'relevant information' items: 'Studentenwerk München', 'Münchner Nahverkehr', 'Tourismus München Info', and 'Musikunterricht geben' (highlighted in red).
- Bottom Panel:** A 'Browser for Task Stadt' window is open, showing a search bar with the URL 'http://www.vek.de/unterricht.htm'. Below the search bar, there is a list of tasks: 'Gitarre' (Dominik Hanel, klassischer und moderner Gitarrenunterricht in München, Tel.: 089 - 74 37 37 07, <http://www.gitarrenkurse.de>) and 'Violine' (Unterricht von hochqualifiziertem Geiger in Würzburg und Herrsching bei München. Tel.: 0177-5613826 e- mail: violinist@patrickschleuter.de).

Summary & Conclusion

- **Concept of weakly-structured workflows for information support for knowledge-intensive work**
 - Late/lazy modeling deal with the need for flexibility of knowledge workers
 - Attached information needs provide the context for precise information delivery (what & when)
- **Experimental evaluation**
 - sound experimental design was difficult
 - sound example was hard to construct
 - design, performance and evaluation were time-consuming
 - first experiment runs discovered previously unknown deficiencies in various tool implementations
 - But:
 - approach was supported by data (wrt. original hypotheses)
 - interesting additional insights and aspects
 - software was consolidated
- **Hope: More “comparable” work in KM community**



Thank you for your attention!

<http://www.dfki.uni-kl.de/frodo>

