Enterprise Attention Management System

Darko Anicic [1] **Nenad Stojanovic**[1] **Dimitris Apostolou**[2]

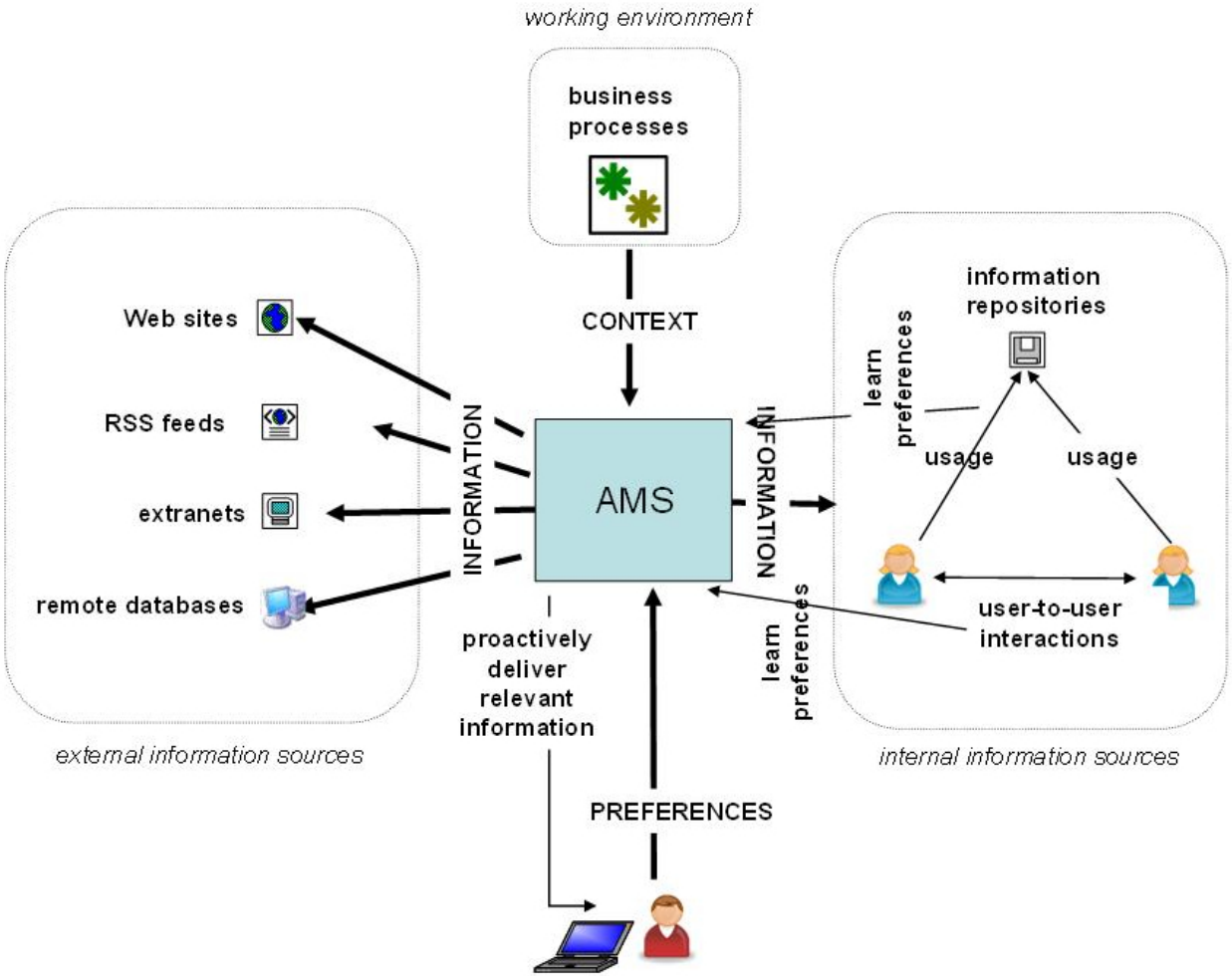
[1] **FZI Forschungszentrum Informatik, Karlsruhe, Germany**
[2] **Department of Informatics Decision Support Systems Lab, University of Piraeus, Greece**

Tenerife – June 2, 2008

Introduction

- How to adapt to knowledge-intensive dynamic business environments including the ability to deal with changing situations and large quantities of information;
- Human attention in knowledge intensive organisations (and on Web, in general) is scarce resource which is difficult to manage and support;
- AMS proactively supports the user in dealing with processes, activities and tasks defined by a semantically-enhanced business workflow.

Attention Management Framework



Requirements

Events

Proactive user support through combination of *context-aware ECA* rules with ontologies.

Contexts

It is important to identify the *context* during which active behavior is relevant.

Preferences

Enabling filtering of relevant information according to its importance/relevance to the given user's context.

Requirements

Events

Proactive user support through combination of *context-aware ECA* rules with ontologies.

Contexts

It is important to identify the *context* during which active behavior is relevant.

Preferences

Enabling filtering of relevant information according to its importance/relevance to the given user's context.

Requirements

Events

Proactive user support through combination of *context-aware ECA* rules with ontologies.

Contexts

It is important to identify the *context* during which active behavior is relevant.

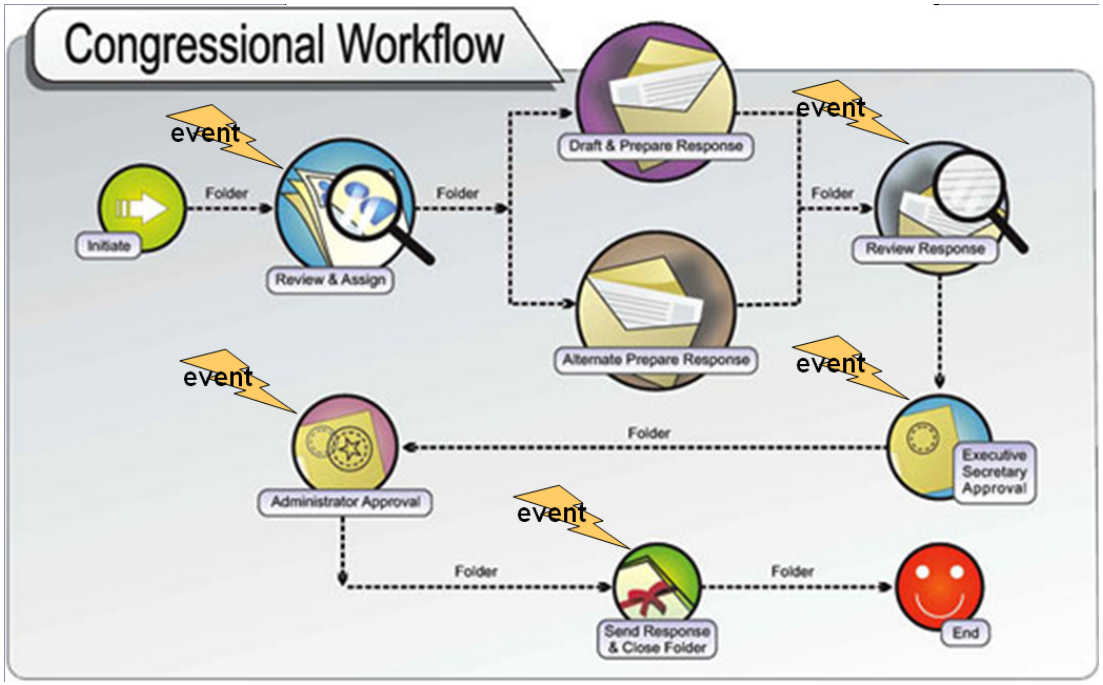
Preferences

Enabling filtering of relevant information according to its importance/relevance to the given user's context.

SAKE Ontologies

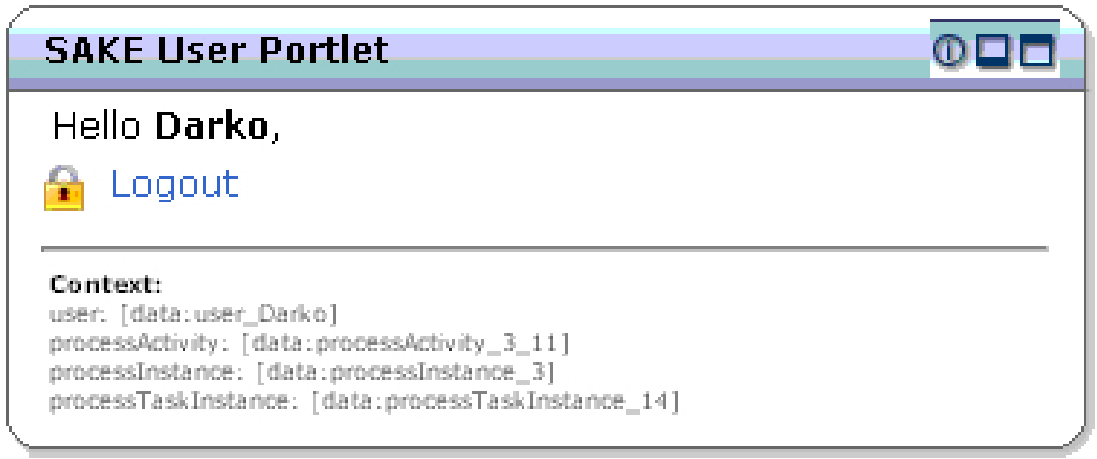
- Information Ontology;
- Process Ontology;
- Preference Ontology;
- Log Ontology;
- SAKE use-case specific ontologies.

Contextual Event Processing



- On an *event* check the context out, and find all relevant preference rules;
- Then execute preference rules in order to proactively deliver relevant information resources.

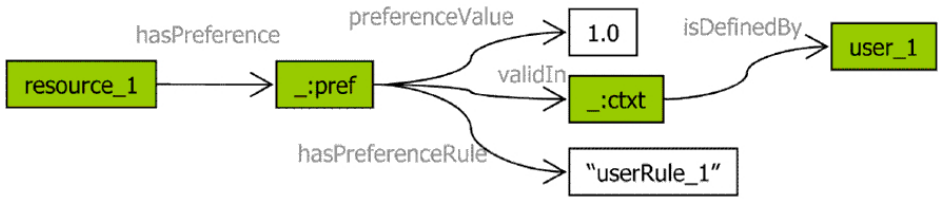
Contextual Event Processing



- Business context is derived using a context observer;
- The context observer triggers an event;
- Which starts reasoner for context evaluation and preference rule execution.

Preference Rules

Preference: n-ary relation between a user, multiple resources, context, and a preference value.



$$\begin{aligned}&\mathrm{Document}(?res) \wedge \mathrm{yearCreated}(?res, \text{"2006"}) \wedge \\ &\mathrm{RuntimeContext}(?ctx) \wedge \mathrm{queryContext}(\mathrm{sakesystem}, ?ctx) \wedge \\ &\mathrm{isDefinedBy}(?ctx, \mathrm{userA}) \wedge \mathrm{isDefinedBy}(?ctx, \mathrm{processZ}) \wedge \\ &\mathrm{swrlx:createIndividual}(?x) \\ \Longrightarrow\ &\mathrm{Preference}(?x) \wedge \mathrm{hasPreferenceRule}(?x, \text{"rule\_2"}) \wedge \\ &\quad \mathrm{preferenceValue}(?x, \text{"1.0"}) \wedge \mathrm{hasPreference}(?res, ?x)\end{aligned}$$

- Information resource may be assigned with different preference values;
- Preference values are not pre-computed and persisted;
- Adding a preference rule may significantly influence the whole preference model.

Preference Editor

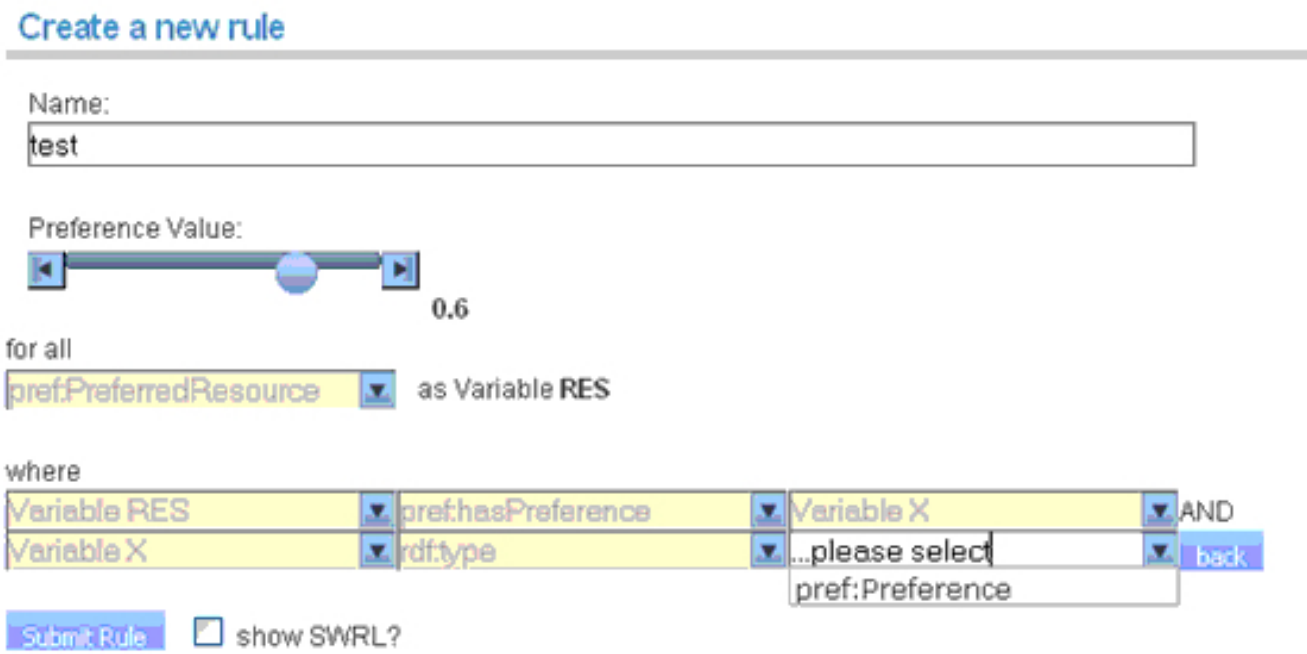
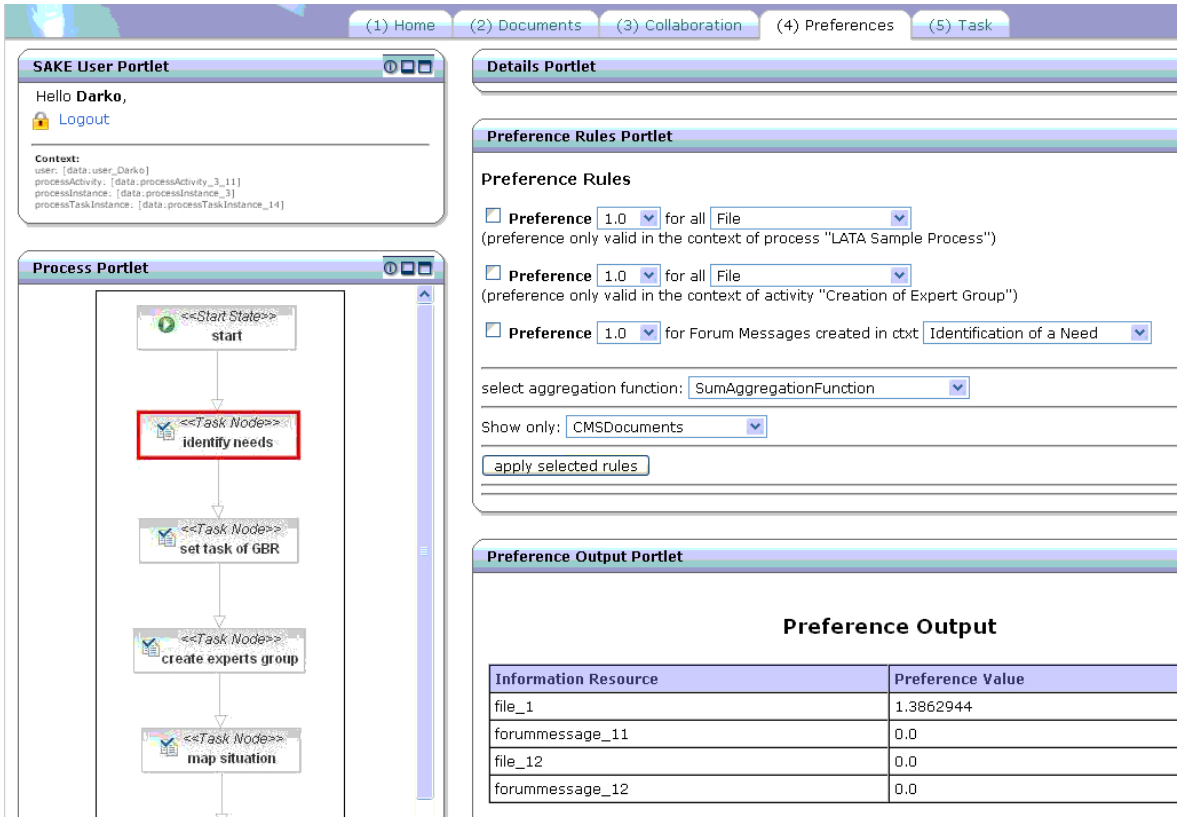
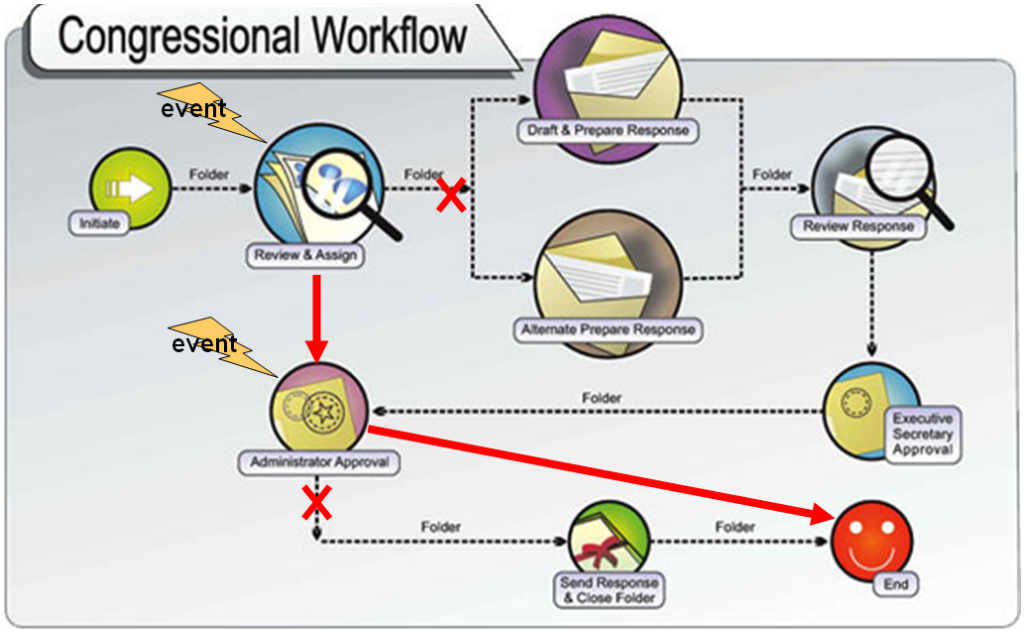


Figure: Preference Editor: Step-wise, interactive rule development

Results of Rule Execution



Future Work



- Synchronization of work, cooperation between concurrent workflows, access to shared resources;
- Collaboration in *ad-hoc* workflows: all this, but not in a completely *predefined* manner.
- Intelligent event processing (logic-based CEP with reasoning capabilities over situations).

Discussion

Thank you!
Questions...