

Architecture of a Light-weight Non-threaded Event Oriented Workflow Engine

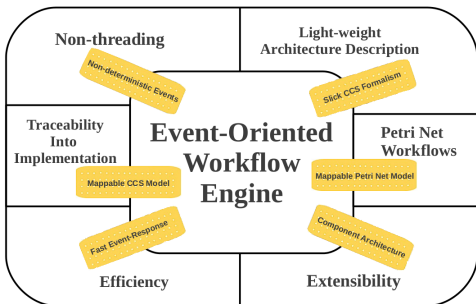
Ahana Pradhan & Rushikesh K. Joshi

Department of Computer Science & Engineering
Indian Institute of Technology Bombay
Powai, Mumbai-400076, India



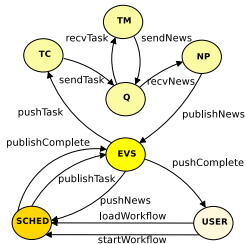
DEBS '14 (Poster & Demo Track)

A Multi-faceted Approach



DEBS '14 (Poster & Demo Track)

CCS Component Architecture



- Scheduler
- Application Server
- Event Server

Composition: System as a Whole

$$\text{SYSTEM} = \text{SCHED} | \text{EVS} | \text{APP} \setminus \{ \text{publishTask}, \text{pushTask}, \text{pushNews}, \text{publishNews}, \text{publishComplete} \}$$

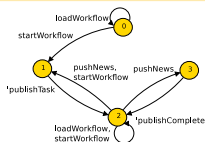

DEBS '14 (Poster & Demo Track)

CCS Model of the Architecture Components

Scheduler and Service Loop Dynamics

$$\begin{aligned} \text{SCHED} &= \text{loadWorkflow} . \text{SCHED} \\ &+ \text{startWorkflow} . ' \text{publishTask} . \text{SLOOP} \\ \text{SLOOP} &= \text{loadWorkflow} . \text{SLOOP} \\ &+ \text{startWorkflow} . \text{SLOOP} \\ &+ \text{startWorkflow} . ' \text{publishTask} . \text{SLOOP} \\ &+ \text{pushNews} . ' \text{publishTask} . \text{SLOOP} \\ &+ \text{pushNews} . ' \text{publishComplete} . \text{SLOOP} \end{aligned}$$

State Machine of SCHED



Event Server Dynamics

$$\begin{aligned} \text{EVS} &= \text{publishTask} . ' \text{pushTask} . \text{EVS} \\ &+ \text{publishNews} . ' \text{pushNews} . \text{EVS} \\ &+ \text{publishComplete} . ' \text{pushComplete} . \text{EVS} \end{aligned}$$

State Machine of EVS



DEBS '14 (Poster & Demo Track)

Application Server Dynamics

$$\begin{aligned} TC &= \text{pushTask}.'\text{sendTask}.TC & Q &= \text{sendTask}.Q + '\text{recvTask}.Q \\ & & & + \text{sendNews}.Q + '\text{recvNews}.Q \\ TM &= \text{recvTask}.'\text{sendNews}.TM & NP &= \text{recvNews}.'\text{publishNews}.NP \\ APP &= TC|TM|Q|NP \setminus \{ \text{sendTask}, \\ & \text{recvTask}, \text{sendNews}, \text{recvNews} \} \end{aligned}$$

State Machine of TC



State Machine of TM



State Machine of NP



State Machine of Q



DEBS'14 (Poster & Demo Track)

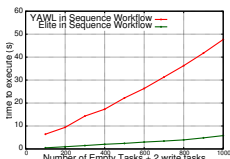
Mapping Petri Net to Engine Scheduler Events

Mapping of Petri net events to workflow scheduling:

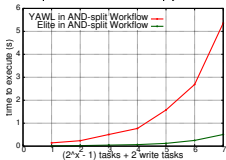
- Marked source place → Workflow uploaded, to be started
- Input token consumption → Schedule of task for execution
- Output token production → Task execution completion
- Marked sink place → Workflow completed



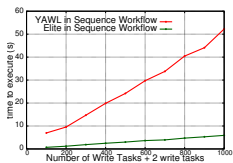
DEBS'14 (Poster & Demo Track)



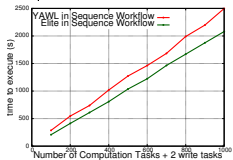
Sequence Workflow of Empty Tasks



Tree Workflow of Empty Tasks



Sequence Workflow of File-write Tasks



Sequence Workflow of Computation Tasks



See You at the Open Session!

