**ABSTRACT**

In several applications like healthcare, time in workflow execution is critical. Several control and data dependencies arise that must be specified, validated as conflict free, and maintained during workflow execution. The author models these kinds of dependencies as constraints that impose temporal restrictions on the relative order of execution of the activities. Hence, a finer granularity of activity execution with respect to time is introduced. The author incorporates a subset of interval algebra in the workflow specification model and the author proposes the T-WfMc specification model. The author examines the consistency issues that arise, and the author proposes different correctness criteria.

**KEYWORDS**

Business Process, Scheduling, Temporal Consistency, Temporal Constraints

**INTRODUCTION**

In several fields workflow applications require co-ordination and synchronisation among activities. Several control and data dependencies arise that must be specified, validated as conflict free, and maintained during workflow execution. These kinds of dependencies can be viewed as constraints that impose temporal restrictions on the relative order of execution of the activities. In this context the concept of time is dominant in workflow execution, because time constraints impose the activity synchronisation requirements, while their fulfilment or the extent of their fulfilment, judges the execution correctness. As a result, temporal synchronisation can be no longer be limited to a consequence of value dependencies among activities.

One of the limitations of workflow systems is that they lack the expressiveness for specifying temporal relationships across activities. Therefore, they fail to provide adequate timing control during the execution. The importance of time-related issues and their corresponding semantics have been identified in the literature (Liu et al., 2011).

We motivate and present the applicability of our approach with a health care example based on the scenario presented in Haimowitz et al. (1996) regarding the emergency room procedure. When a patient enters the emergency room he has to be evaluated by nurses in “triage”. The nurse takes patient history, treats pressing problems and takes a few preliminary tests such as temperature, blood
pressure, etc. At this point there are several alternatives: the patient might be re-examined and sent home, might be sent for further tests like x-rays, blood test, etc., or he might be sent for surgery. After that the patient is discharged.

In the diagrammatic notation (see Figure 1), tasks are represented as rectangles while constraints as directed arcs labelled by expressions in italic that denote the name of the constraint. The OR constructs are represented as circles, denoting OR-split and OR-join. We can see that activities $a_6$ and $a_1$ have to start and complete at the same time (are constrained by “equals”). Therefore, traditional synchronisation techniques that allow only precedence/parallel execution cannot support advanced temporal requirements.

In this article, we develop a model for temporal workflow, a mechanism for consistency checking, and algorithms for consistent execution of activities with respect to temporally constrained workflow applications. Specifically, we addressed the following problems:

- In temporal workflow applications, the user wants the ability to specify temporal constraints on the execution of tasks. How can we model such a workflow process that allows for temporal constraint specification? We have developed a model where temporal workflow processes are represented through a time-constrained graph that extends the existing workflow model;
- A temporal process specification is meaningful only if there is at least one execution that satisfies all the constraints. What are the consistency checking tests that need to be performed at build time, and what are the corresponding algorithms that will verify the consistency of the specification? We have developed a consistency checking mechanism that verifies consistency of the specification with respect to the temporal constraints and the duration of tasks. The consistency checking algorithms produce for every task a time interval such that there is a consistent execution if tasks are initiated in their assigned time intervals. Another issue that has not been addressed in the literature is: how can the user specify the consistency goals? (i.e., what is the consistency level that the process has to maintain?). In most applications, we are not interested in rigid constraint maintenance - the user can accept some degree of inconsistency. This has to be defined and measured. In our model, the correctness requirement specification provides a measure for counting the degree of inconsistency in the presence of constraints’ violation;

**Figure 1. Emergency room example**

![Emergency room example diagram]
Related Content

Estimation of Factor Scores of Impressions of Question and Answer Statements

[www.igi-global.com/article/estimation-of-factor-scores-of-impressions-of-question-and-answer-statements/89775?camid=4v1a](www.igi-global.com/article/estimation-of-factor-scores-of-impressions-of-question-and-answer-statements/89775?camid=4v1a)

Knowledge Management

[www.igi-global.com/chapter/knowledge-management/75755?camid=4v1a](www.igi-global.com/chapter/knowledge-management/75755?camid=4v1a)
Integration of Human Factors to Safety Assessments by Human Barrier Interaction

Enterprise Integration: Challenges and Solution Architecture
Leo Shuster (2013). Service-Driven Approaches to Architecture and Enterprise Integration (pp. 43-66). www.igi-global.com/chapter/enterprise-integration-challenges-solution-architecture/77944?camid=4v1a