Probabilistic Time Management for Automated Business Processes

Horst Pichler (PhD student), Johann Eder (PhD supervisor)
University of Klagenfurt, Institute for Informatics-Systems, Austria
horst.pichler@uni-klu.ac.at, johann.eder@uni-klu.ac.at

***Abstract*—Empirically learned characteristics of a business process, like branching probabilities and average execution durations of activities, are usually used for process simulation and the creation of scheduling scenarios as a basis for process optimization. We utilize this information by calculating a probabilistic timed process model based on the process structure, time properties and probabilistic information. It yields possible execution intervals for activities in order to meet defined time constraints, where the probabilistic information provides information about the likelihood of a future deadline violation. Another application area is the calculation of the probability for expected (remaining) process execution times. In order to cope with the interoperability aspect of business processes we currently examine the possibilities to map these workflow-based concepts on composite web service (CWS) environments. Slow autonomous web services, invoked by a CWS, can have an disastrous impact on the overall process response time. Thus techniques are needed predict the process duration based on the anticipated response time of participating web services, which enables us too avoid these services or to optimize them for faster execution.**

I. TIME MANAGEMENT

Typically, time violations increase the cost of business processes because they require some type of exception handling. Therefore, the comprehensive treatment of time and time constraints is crucial in designing and managing business processes. Process managers need tools that help them anticipate time problems, pro-actively avoid time constraints violations, and make decisions about the relative process priorities and timing constraints when significant or unexpected delays occur. These are established problems in workflow management, but most of the provided solutions suffer from the uncertainty of time information in processes. This vagueness stems mainly from two aspects: The duration of a task can vary greatly and in a workflow different paths may be chosen with decisions taking place during the execution. Some approaches try to address this problem by introducing time intervals (e.g. best and worst case), or suppose some kind of distribution (e.g. normal distribution) of duration values. The probabilistic approach, presented in this paper, offers a new solution to these shortcomings and enables more precise predictions about the likelihood of the future behavior of a process execution.

II. TIME HISTOGRAMS

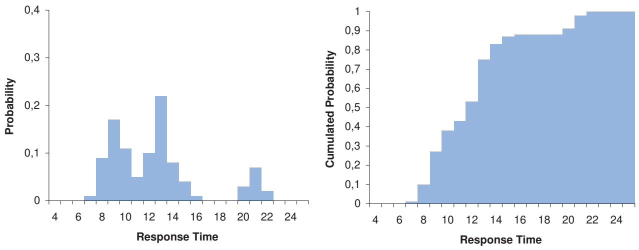

Fig. 1. Response time histogram

Activities as part of workflows are capable of hiding complex business processes with greatly varying durations. Thus simple duration representations, as mentioned above, fall short because they are to imprecise. They all do not take into account that the existence of conditional structures in the control flow may result in a non-evenly distributed duration with multiple peaks. Thus the concept of time histograms (THs) was introduced in [1][2] as structure to represent a distribution function on a time property (e.g. activity duration or process response time). Figure 1 shows an example of the graphical representation of a TH representing the response time of a process. It (on the left hand side) tells us that that the bigger part of future executions will last between 7 and 16 time units with peaks at 9 and 13 plus some additional outliers around 21. The time unit depends on the chosen minimum granularity, e.g. millis, seconds or minutes. According to the associated cumulated representation (on the right hand side) we can for example state that there is a likelihood of 88% that the activity will last 16 time units or less, thus we are enabled to determine the probability for a given threshold duration or vice versa. THs are either generated by expert estimations, the extraction of empirical information from process execution logs or calculated, based on the the structure of an underlying business process. THs are also used to calculate execution intervals for each activity in the process, bounded by earliest possible start times and latest allowed end times (in order to hold process deadlines) for each activity in the process. This information may be used for time constraint validation during build time or pro-active task scheduling during run-time.

III. WORKFLOW GRAPH AND DURATION HISTOGRAM CALCULATION

[1][2] show how different types of time histograms are calculated, based on workflow definitions using sequences, conditional routing (and-splits/joins), parallel routing (x-or-splits/joins) or iterations (loops). As prerequisite for the calculation we define a (directed acyclic) workflow graph holding activities and the control flow dependencies between these activities. Additionally probabilistic information about the branching behavior of a process and the duration of its activities, represented as THs, must be provided. Until now we allowed only well-formed workflow graphs. A well-formed graph has to adhere to certain constraints: a) For every split-node there exists exactly one join-node of the same type, e.g.

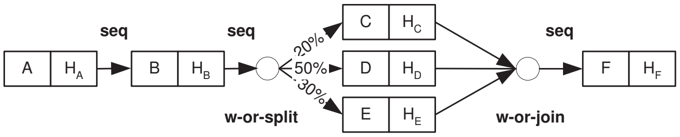

Fig. 2. Well-formed workflow graph

an or-split must have an associated or-join, b) every path that originates from a split-node must merge in the associated join-node, and c) dependencies originating from activities between split/join-pairs must not leave their paths or connect to activities outside of these pairs. Figure 2) shows a simple graph containing activities (A,B,C, etc.), probabilistic duration information (THs named H_A,H_B,H_C, etc.) and two different types of control flow structures: sequence and weighted x-or, which demands that between split an join exactly one out of many paths must be executed, depending on a given branching condition or choice. The weight represents the expected branching probability of each path. To calculate for instance the TH for the workflows response time it is necessary to traverse the graph from the first to last activity while summing up and weighting the activity-DHs according to control structures and branching probabilities. The result of the calculation is a DH for the workflow consisting of tuples for every possible overall duration with its according probability. For further details see [1][2].

IV. MAPPING THE CONCEPTS ON COMPOSITE WEB SERVICE ENVIRONMENTS

The next step in the evolution of business process automation are composite web services (CWSs) to support business processes within organizations as well as business processes spanning several organizations like supply chains. Thus the most critical need in companies will be to provide services with a better quality than their competitors. To assess the quality of service (QoS) it is necessary to define measures which are significant indicators for certain quality aspects, where e.g. expected or guaranteed process duration ranks among the most important dimensions. Slow autonomous web services, invoked by a CWS, can have an disastrous impact on the overall process response time. Thus techniques are needed predict the process duration based on the anticipated response time of participating web services, which enables us too avoid these services or to optimize them for faster execution. Due to the resemblance between the concepts workflow and CWSs, an adoption of the above presented seems reasonable. The following subsections outlines our current and future research projects and arising problems for a solution using executable processes in BPEL4WS.

A. *Response time histograms*

Again the response time of a process is either detected empirically or calculated. Workflow activities are mapped to invoked web services and the duration is mapped to response times of these services. The histogram concept will prove valuable as infrequent response time outliers, caused by delays stemming from communication problems, will be located in the upper regions of the cumulated histogram. This enables us to cut them of by defining a threshold less than 100%. E.g. the response time of a service with a likelihood of 98% will not include very infrequent delays but still yield acceptable forecasts.

B. *Adapting the workflow graph for BPEL4WS definitions*

The approach explained above was designed for graphs representing well-formed workflow definitions. These constraints are too rigid for possible compositions defined with different types of primitive activities (invoke, receive, reply, etc.) and structured activities (sequence, link, flow, etc.) available in BPEL4WS, which may result in non-well-formed workflows. Therefore an extended graph-representation which allows non-well-formed structures must be defined and the calculation algorithms must be adapted. Future research will build upon pattern based analysis of BPEL4WS. Another research topic will be the implementation of a parser to extract a non-well-formed graph from a BPEL-definition.

C. *Gathering data for response time histograms*

In workflow systems empirical data can easily be extracted from the workflow log. In BPEL-environments this can either be done by the process-execution-engine or, if the engine has no logging-feature, be implemented as part of the process itself. However, to generate response time histograms and branching probabilities it is necessary to log the system times of each request and each response as well as the control flow of all instances. This issue tends to be a major problem, especially in flexible environments where autonomous web services, accessed by the composition, are frequently changing. For these cases web service response times could be stored and administered by a trusted third party, which offers an interface to access response time statistics, similar to or as an extension of a Web Service Level Aggreement-architecture.

D. *Possible application scenario*

Equipped with the prerequisites explained above a possible application scenario consists of: a) Web service compositions published as web services to the outside world. b) These web services will have additional interfaces yielding the expected service response time as TH. c) The TH can either be generated by accessing statistical data or calculated by applying the above mentioned algorithms on the structure of the composition. The services invoked by the composition are themselves equipped with these interfaces. d) Dependent on the resulting TH the requestor is enabled to make several decisions (like using this service or selecting an alternative one or for providers: optimizing the process for faster execution, etc.).

REFERENCES

[1] J. Eder, H. Pichler, W. Gruber, M. Ninaus. *Personal Schedules for Workflow Systems*. In Proceedings of the International Conference on Business Process Management, Eindhoven, The Netherlands, Springer Verlag (LNCS 2678), 2003.

[2] J. Eder and H. Pichler. Duration Histograms for Workflow Systems. In Proceedings of the Working Conference on Engineering Information Systems in the Internet Context, 2002, Kanazawa, Japan, Kluwer Academic Publishers, page 239-253.