Alaska Simulator Toolset for Conducting Controlled Experiments on Process Flexibility

Barbara Weber[1], Jakob Pinggera[1], Stefan Zugal[1], and Werner Wild[2]

[1] Quality Engineering Research Group, University of Innsbruck, Austria
{Barbara.Weber,Jakob.Pinggera,Stefan.Zugal}@uibk.ac.at
[2] Evolution Consulting, Innsbruck, Austria
werner.wild@evolution.at

Abstract. Alaska Simulator Toolset (AST) is a software suite developed at the University of Innsbruck for exploring different approaches to business process flexibility by using a familiar metaphor, i.e., travel planning and execution. For this, AST provides integrated support of several decision deferral based approaches to process flexibility and enables their systematic comparison. Moreover, AST facilitates the design and execution of controlled experiments through experimental workflow support. This paper introduces AST, explains its underlying meta-model, discusses factors which can be investigated using AST and shows how it can be used for designing and conducting experiments.

1 Introduction

Alaska Simulator Toolset (AST) has been developed to address the investigation of strengths and weaknesses of different approaches to flexibility in Process-aware information systems (PAIS). In particular, AST allows a systematic comparison of these approaches through the execution of controlled experiments.

A critical success factor in applying a PAIS is the option to flexibly deal with process changes [3,4] (e.g., to react to changes, like shifts in customers' attitudes or the introduction of new laws [1]). To address the need for flexible PAISs, competing paradigms enabling process changes and process flexibility have been introduced, e.g., adaptive processes [5], case handling [6], declarative processes [7], and late binding and modeling [8] – for an overview see [9]. While adaptive processes provide flexibility by supporting structural *process adaptations*, flexibility can also be achieved through loosely specified process models which enable *deferring decisions* about the exact process structure from modeling time to run-time [9], when more information is available. By deferring decisions to run-time the strict separation of build-time (i.e., modeling or planning) and run-time (i.e., execution), which is typical for traditional workflow management systems, can be relaxed and planning and execution become more closely interwoven. Unfortunately, however, little research exists so far on the suitability of respective approaches depending on the characteristics of business processes to be supported. In particular, there is a lack of empirical insights on the selection of an appropriate approach.

P. Soffer and E. Proper (Eds.): CAiSE Forum 2010, LNBIP 72, pp. 205–221, 2010.

To pick up this need this paper introduces AST, which is used for the systematic comparison of different approaches to decision deferral. Section 2 elaborates on the support for different approaches for deferring decisions and describes the underlying meta-model. Section 3 discusses several factors which have a potential impact on the suitability for these approaches. Section 4 then describes how AST can be used for designing and executing controlled experiments. Related work is listed in Section 5, Section 6 concludes the paper with a summary and an outlook.

2 Alaska Simulator Toolset

To foster the comparison of different approaches for process flexibility Alaska Simulator Toolset[1] has been implemented, which takes a *journey* as metaphor for a *business process*. Section 2.1 summarizes the different approaches for process flexibility and decision deferral supported by Alaska Simulator, while Section 2.2 introduces its meta-model and Section 2.3 discusses the journey metaphor.

2.1 Support for Decision Deferral Patterns in Alaska Simulator

To foster the comparison of different approaches to process flexibility, Alaska Simulator provides integrated support for the decision deferral patterns described in [9] (cf. Fig. 1).

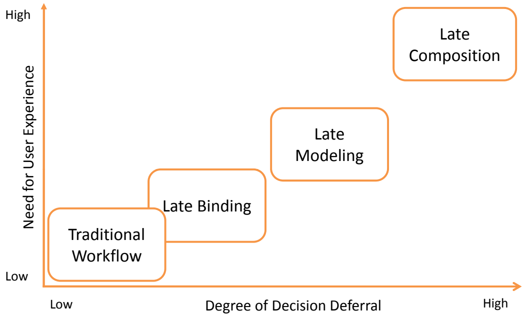


Fig. 1. Different Planning Approaches

Traditional workflow management systems offer little flexibility and require to completely predefine a business process in advance, thus employing a strictly plan-driven approach separating modeling and execution entirely (cf. Fig. 2A). *Plan-driven approaches* try to overcome uncertainty by developing a very detailed plan up-front, which is then regarded as the schema for execution. However, as plans tend to be more imprecise the earlier they are elaborated [18],

[1] Developed at the University of Innsbruck, `http://www.alaskasimulator.org`

concentrating all planning efforts in the planning phase imposes the risk of rendering plans useless due to changing conditions (e.g., changing requirements) [1].

To address this risk both *Late Binding* and *Late Modeling* offer slightly more flexibility by allowing for placeholder activities which can be refined during runtime (cf. Fig. 2B). Using *Late Binding* the selection of the actual content for the placeholder activity (from a set of predefined fragments) can be deferred to run-time. The *Late Modeling* pattern goes one step beyond this and allows for the modeling (not only selection) of the placeholder content at run-time. Using *Late Binding* or *Late Modeling* the strict separation of modeling and execution is slightly relaxed allowing for modeling activities during run-time, but restricted to pre-specified parts in the process, i.e., to the placeholder activities (cf. Fig. 2B).

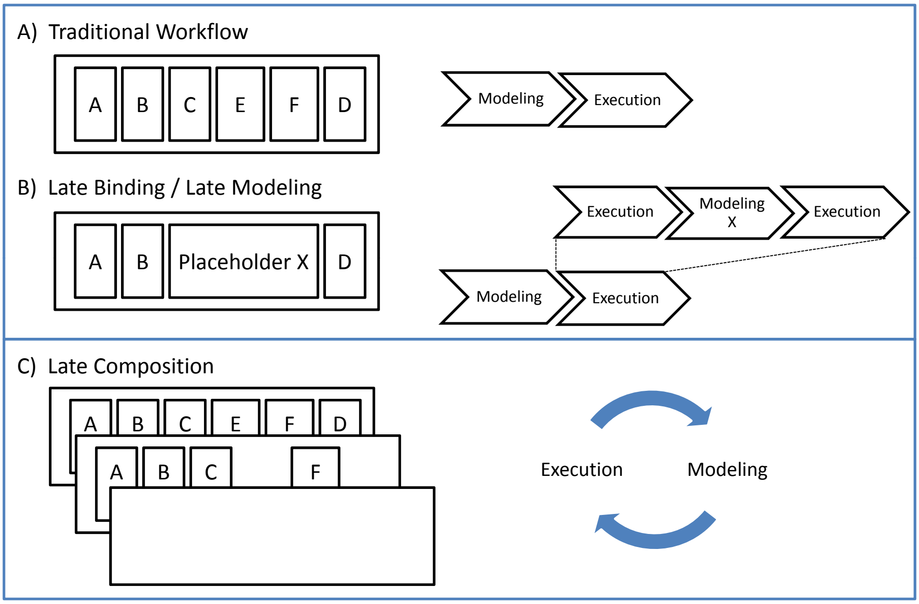


Fig. 2. Planning and Execution Modes

Most flexibility is offered by the *Late Composition* pattern, which allows users to iteratively compose a business process (or a journey respectively) by selecting activities from a repository, while respecting all existing constraints. By employing iterative and continuous planning *Late Composition* mitigates the risk of rendering plans useless due to changing conditions [1,19,20] (cf. Fig. 2C). In contrast to the plan-driven approach (as employed by traditional workflow management systems) no detailed up-front plan is created, the initial plan remains more coarse grained [19]. Over time the initial plan is iteratively refined, allowing for the integration of newly gained knowledge. In contrast to plan-driven approaches, which focus on the plan as an artifact by itself, the emphasis is put on planning as an ongoing activity. Distributing planning effort over the entire duration enables deferring decisions to the last responsible moment [1] and to create opportunities for learning. When this last responsible moment actually will be, highly depends on the characteristics of a business process. Figure 3 classifies activities according to their availability and reliability and shows strategies on how to best deal with particular classes of activities.

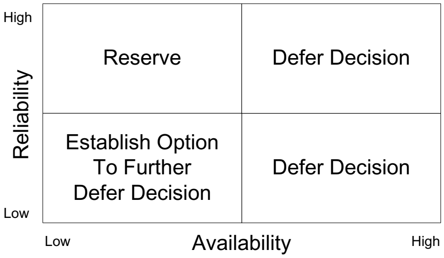


Fig. 3. Deciding at the Last Responsible Moment

Fig. 3 shows that committing to highly available activities (i.e., by making a resource reservation) should be avoided. Reserving respective activities introduces unnecessary early commitment without gaining any benefits. In contrast, activities with high reliability in combination with low availability should be reserved before the reservation deadline to guarantee resource availability (i.e., the last responsible moment would be just before the reservation deadline). Activities with low availability and low reliability are most difficult to handle. Depending on the business value that can be gained and the cost of the respective activity, it might be worth to commit to the activity and to schedule a second more reliable activity in parallel. Therefore, instead of choosing between one of the activities right away an option is build allowing to defer the last responsible moment for deciding between the alternatives.

How much pre-planning is “enough” highly depends on the characteristics of the business process. In the most extreme cases a fully pre-specified plan like in traditional workflow management systems (i.e., if all activities have a low availability and a high reliability) or an empty initial plan (i.e., if the availability is high) might be most appropriate.

2.2 Alaska Simulator Meta-model

This section describes the meta-model underlying Alaska Simulator (cf. Fig. 4).

Goal. When conducting a journey with AST, the goal is to maximize the travel experience (i.e., the overall “business value” of the journey). For optimizing the execution of a journey, information about the benefits (i.e., business value), costs and duration of actions, but also about their availability and reliability is essential.

Journey, Journey Plan, Locations. Alaska Simulator supports users in composing *journeys* based on a pre-specified *journey configuration* (i.e., instance of the meta-model) on a *journey plan.* A journey plan is represented by a calendar

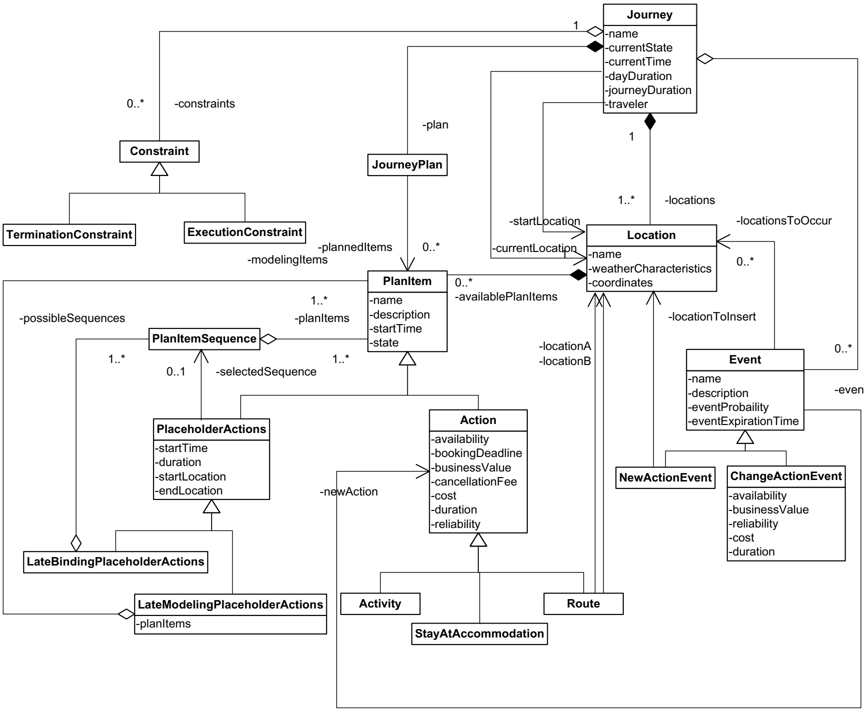


Fig. 4. Alaska Simulator Meta-Model

for scheduling *plan items* (i.e., activities, routes or accommodations). During a journey different *locations* can be visited, each providing distinct *plan items* to select from (e.g., activity `Visit Denali National Park` is only available a location `Denali`).

Plan Items. Plan Items can be *atomic actions* or *complex placeholder actions. Atomic actions* can either be *activities*, *routes* or *stays at accommodations.* For each atomic action its "business value", costs, duration, availability and reliability are specified. Moreover, a booking deadline as well as cancellation fees are known. *Complex placeholder actions*, in turn, provide placeholders which can be refined during the execution of a journey. *Late Binding Placeholder Actions* provide a set of *Plan Item Sequences* from which one can be selected during run-time. *Late Modeling Placeholder Actions*, in turn, allow users to compose a *Plan Item Sequence* during run-time from a set of predefined plan items respecting existing constraints. *Plan items* in general, in addition to their name and a description, have a start time (which is determined by their position in the journey plan) and a particular state (e.g., scheduled, booked, started, executed). A state transition diagram depicting the lifecycle of an action is illustrated in Fig. 5.

Constraints. When composing a journey different constraints have to be considered. Constraints can be classified as execution and termination constraints.

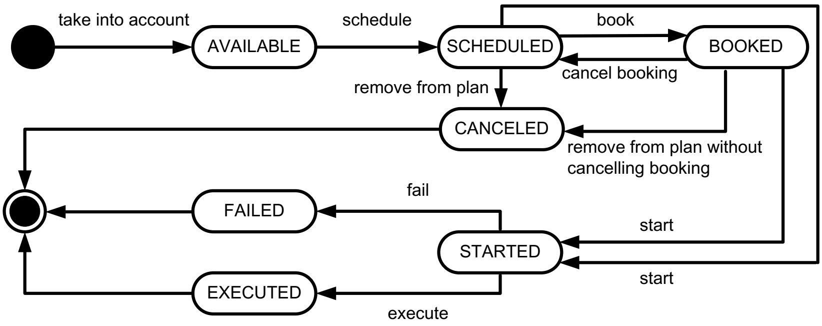


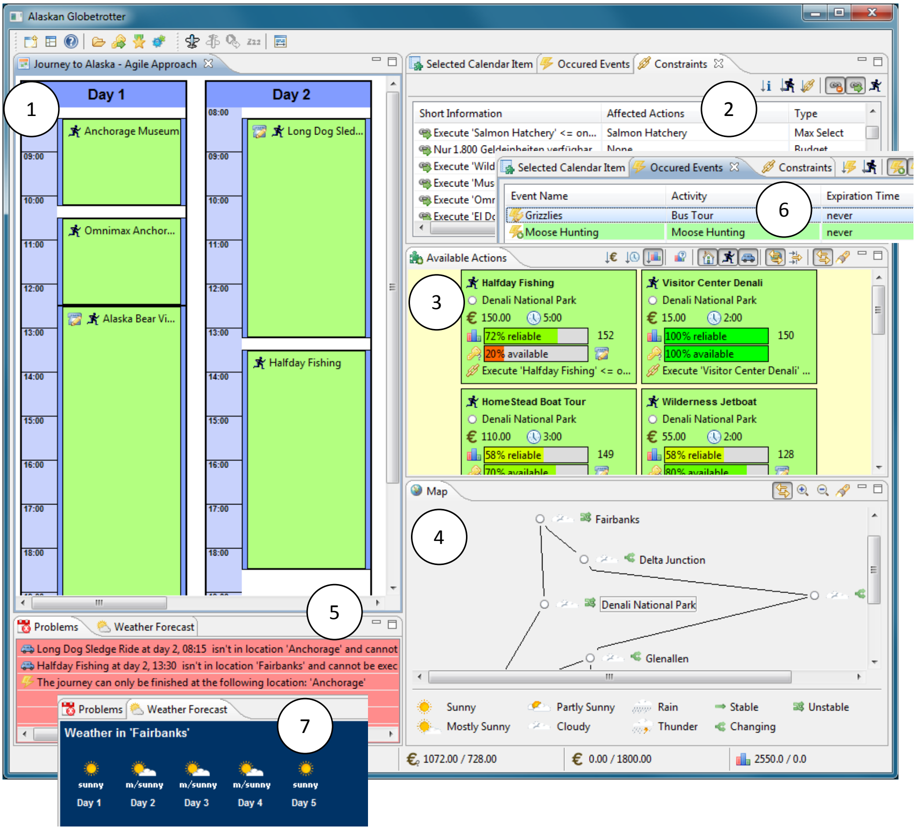
Fig. 5. Action Lifecycle

Execution constraints restrict the execution of activities (e.g., an activity can be executed at most once, or two activities are mutually exclusive). *Termination constraints*, in turn, affect the termination of process instances and specify when process termination is possible. For instance, an activity must be executed at least once before the process can be terminated or a particular end location must be reached for terminating the journey. Alaska Simulator provides support for all control-flow constraints described in [13] (e.g., a *response constraint* requiring that activity `A` is eventually followed by `B` or a *mutual exclusion constraint* prohibiting that activity `A` and `B` co-occur). In addition, several resource and time related constraints are supported (e.g., a *budget constraint* restricting the available travel budget, an *end location constraint* requiring that the journey ends at a particular location, a *completion day constraint* restricting the journey to a particular duration, an *execution time constraint* restricting the time when a particular activity can be executed, and a *time lag constraint* requiring a minimum time-lag between executing two activities).

Uncertainty. Journeys are characterized by incomplete information prior to execution. The outcome of an action within a journey plan is not predefined and varies with the *weather conditions* encountered. The degree of variation is defined by the action's *reliability*, i.e., low reliability indicates that the outcome of an action is highly weather dependent. The overall business value of a journey (i.e., a numeric value representing the travel experience) is calculated as the sum of business values of all performed actions. Prior to performing the journey only the *expected business value* for each action as well as its reliability are known. The expected business value which is calculated based on the action's reliability and the average weather characteristics of the location where the action is available. During the journey the action's *actual business value* is calculated based on the actual weather conditions encountered.

Resource Scarcity. When planning a journey potential resource scarcity has to be considered. By firmly *booking* an action, its *availability* can be guaranteed, but the cost of the action must be paid immediately. If the booking is canceled during the journey, a *cancellation penalty* might apply, thus making too early commitments costly. Furthermore, booking is only possible up to a certain time before executing the action, as specified by the *booking deadline*.

Events. In addition to changing weather conditions, *unforeseen events* (e.g., a traffic jam resulting in delays) create uncertainty in a journey. On the one hand, a *ChangeActionEvent* changes values of existing actions during the journey. For example, a particular event might increase or decrease the business value, the availability, the costs or the duration of an action. On the other hand, a *NewActionEvent* allows to introduce new actions at a particular location during run-time. Moreover, events occur with a predefined probability during run-time, may have an expiration time and can be attached to a particular location (i.e., the event only occurs when the traveler is at a particular location).



Short Information	Affected Actions	Type
Execute 'Salmon Hatchery' <= on...	Salmon Hatchery	Max Select
Nur 1.800 Geldeinheiten verfügbar	None	Budget
Execute 'Wild		
Execute 'Mus		
Execute 'Omr		
Execute 'El Dc		

Event Name	Activity	Expiration Time
Grizzlies	Bus Tour	never
Moose Hunting	Moose Hunting	never

Halfday Fishing
Denali National Park
€ 150.00 5:00
72% reliable 152
20% available
Execute 'Halfday Fishing' <= o...

Visitor Center Denali
Denali National Park
€ 15.00 2:00
100% reliable 150
100% available
Execute 'Visitor Center Denali' ...

HomeStead Boat Tour
Denali National Park
€ 110.00 3:00
58% reliable 149
70% available

Wilderness Jetboat
Denali National Park
€ 55.00 2:00
58% reliable 128
80% available

Long Dog Sledge Ride at day 2, 08:15 isn't in location 'Anchorage' and cannot
Halfday Fishing at day 2, 13:30 isn't in location 'Fairbanks' and cannot be exec
The journey can only be finished at the following location: 'Anchorage'

Weather in 'Fairbanks'

Day 1	Day 2	Day 3	Day 4	Day 5
sunny	m/sunny	m/sunny	m/sunny	sunny

Fig. 6. Screenshot of Alaska Simulator

Fig. 6 depicts the graphical user interface of the Alaska Simulator. Users can compose their individual journey plan by dragging available actions from the Available Actions View (3) onto the travel itinerary (i.e., journey plan) (1). Actions are usually only available at a particular location on the map (4).

Existing constraints are displayed in the Constraint View (2) and have to be considered when composing a concrete journey plan. After each user action, the journey plan is validated and the user is informed about any constraint violations and plan inconsistencies (5). Unforeseen events are shown in the Event View (6). Prior to executing the journey users have to rely on weather statistics derived from the weather characteristics of each location, which can be selected on a map (4). During the journey a weather forecast is provided with the current weather conditions at each location (7).

2.3 The Journey Metaphor

This section discusses the journey metaphor and shows how the concepts described in the previous section relate to business processes.

While the *goal* of a journey is to maximize "travel experience", typical goals for business processes are minimizing costs, cycle times or the optimization of quality or customer satisfaction [14].

Journeys are composed based on *journey configurations*, likewise *process instances* representing concrete business cases based on a predefined *process schema*. Both in journeys and business processes certain activities are only available at certain `locations` (e.g., in a medical process surgeries can only be performed in an operating room). While journeys are composed from activities, routes and accommodations, business processes comprise a set of activities. In addition to *atomic activities*, *complex placeholder activities* (allowing for Late Binding and Late Modeling) can also be found in business processes.

When composing a journey, or a business process respectively, existing *contraints* need to be considered. Thereby, support provided by Alaska Simulator is comparable to existing declarative process management systems like DECLARE [13].

Like journeys, highly flexible business processes are characterized by incomplete information prior to execution. In a journey, *uncertainty* is, for example, caused by unforeseen weather conditions, while uncertainty in business processes refers to the difficulty to predict the exact activities and resources that are required to perform a particular process (e.g., due to changing requirements caused by dynamic changes of the business environment) [15].

Both, when planning a journey or executing a business process, potential *resource scarcity* has to be considered (e.g., appointments in medical processes to deal with the limited availability of doctors) [16].

Unforeseen events have to be handled when conducting journeys (e.g., a traffic jam resulting in delays), or executing a business process (e.g., changing requirements or an allergic reaction during a medical treatment) [17].

Motivation. The journey metaphor has been chosen since the domain of travel planning is familiar to the majority of experimental subjects. Furthermore, journey planning is an attractive context for many people to become engaged in, which significantly improves their willingness to use the system for experimental purposes and as a consequence increases internal validity of data.

Limitations. Using a metaphor, however, imposes also several limitations. While the concepts underlying Alaska simulator are mostly comparable to the ones in PAISs, the domain under investigation is fixed to travel planning. Generalizability to other domains (which are less familiar to the experimental subjects) has to be investigated. Another limitation relates to the fact that Alaska Simulator, even though it allows scheduling of actions in parallel, currently does not support the parallel execution of actions (since this does not naturally fit the travel metaphor). As a consequence, the findings obtained with Alaska Simulator, should be complemented with field studies for further validation.

3 Factors for Investigating Different Flexibility Approaches

In the previous sections we have introduced the concepts underlying Alaska Simulator (cf. Section 2.2), explained the journey metaphor (cf. Section 2.3) and illustrated how it supports the different decision deferral patterns (cf. Section 2.1). This section discusses different factors having an impact on the suitability of a particular approach for decision deferral, which can be investigated using Alaska Simulator. These factors can be grouped into two categories: configuration characteristics and personal characteristics.

3.1 Configuration Characteristics

Alaska Simulator allows researchers to investigate the impact of different configuration characteristics on the suitability of a particular approach for decision deferral. Furthermore, Alaska Simulator can be used to compare different approaches for particular configuration characteristics.

- **Actions.** The different attributes of the actions in the journey configuration are a potential factor influencing the suitability of a particular approach. For example, planning and executing a business process or a journey respectively and in particular resolving exceptional situations might be easier if the configuration comprises many small actions in contrast to a few very large actions.
- **Constraints.** The suitability of a particular approach for planning and executing a business process or a journey respectively might be affected by the number and type of constraints. Moreover, the presence of global constraints might have a potential negative impact on the applicability of *Late Binding* or *Late Modeling* respectively.
- **Events.** The probability of unforeseen events that can occur during run-time might be another factor affecting suitability of the different approaches.
- **Uncertainty.** The selection of a suitable approach might also depend on the *reliability* of the different activities. With increasing uncertainty decision deferral might become more important.
- **Resource Scarcity.** Suitability might also be affected by the *availability* of activities. While availability of resources is facilitating decision deferral, resource scarcity is creating significant challenges.

3.2 Personal Characteristics

Even though the advantages of applying decision deferral techniques are apparent, especially for highly uncertain environments – they foster both *learning* (i.e., the ability to resolve uncertainty) and *flexibility* (i.e., the ability to exploit learning outcomes) [21], their successful application also requires user experience. In particular, as illustrated in Fig. 1, with increasing flexibility the need for user support increases. To fully exploit the benefits, adopters must not only have the skills to make use of decision deferral techniques, to be able to determine the last responsible moment and to know how and when to build options (i.e., to create opportunities for learning), but also be able to leverage learning results. Literature suggests that talent and skills are among the critical people-factors for agile methods [22,23], which tend to necessitate a richer mix of higher-skilled people than plan-driven approaches [24]. This raises the question of how well inexperienced users can employ respective planning techniques and whether these techniques can outperform plan-driven approaches for a lower level of expertise.

Novices and experts might not only differ in their ability to defer decision to run-time, but also in handling unforeseen events or obeying complex constraints. AST allows, for example, to investigate to what extent novice users and experts differ and how the novice user's modeling approach changes when uncertainty increases.

4 Designing and Executing Experiments with Alaska Simulator Toolset

This section describes details about the functionality AST provides for supporting researchers in designing and conducting experiments. AST consists of three major components: *Alaska Configurator*, *Alaska Simulator* and *Alaska Analyzer*.

- **Alaska Configurator** allows researchers to design journey configuration (including locations, actions, events and constraints as well as the degree of uncertainty and resource scarcity) executable through Alaska Simulator. In addition, Alaska Configurator supports researchers in designing experimental workflows guiding subjects during the experiment.
- **Alaska Simulator** allows to plan and execute journeys making use of different decision deferral approaches. Thereby, each step performed while planning and executing a journey is logged in an MXML log file for later analysis (e.g., using the process mining tool ProM [25]).
- **Alaska Analyzer** allows detailed manual analysis of planning behavior by supporting replay of journeys step by step.

In the following we describe main functionalities of AST and how design and execution of controlled experiments is supported, using the experiment described in [26] as our example (cf. Fig. 7 for the experimental design).

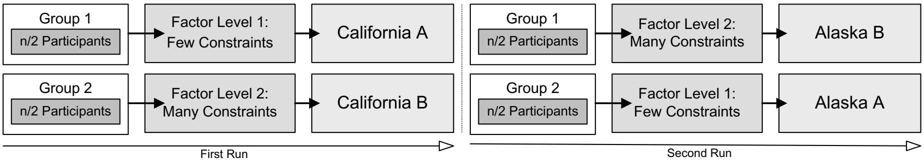
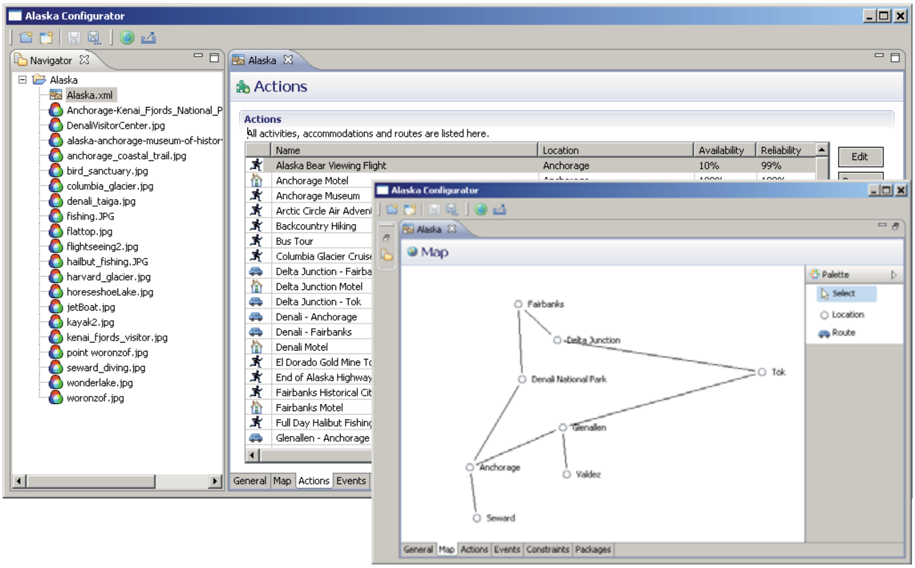


Fig. 7. Experimental Design

Alaska Configurator was used to design two journey configurations (California and Alaska) including locations, actions, events, constraints as well as variability of weather conditions (cf. Fig. 8 and Fig. 9). For each journey configuration two variants were created, one for each factor level (i.e., few and many constraints). To gather the participants' demographic information Alaska Configurator was used for designing a survey (cf. Fig. 10). The journey configurations and the survey were then assembled to an experimental workflow (cf. Fig. 11).



Alaska
Alaska.xml
Anchorage-Kenai_Fjords_National_P
DenaliVisitorCenter.jpg
alaska-anchorage-museum-of-histor
anchorage_coastal_trail.jpg
bird_sanctuary.jpg
columbia_glacier.jpg
denali_taiga.jpg
fishing.JPG
flattop.jpg
flightseeing2.jpg
hailbut_fishing.JPG
harvard_glacier.jpg
horeseshoeLake.jpg
jetBoat.jpg
kayak2.jpg
kenai_fjords_visitor.jpg
point woronzof.jpg
seward_diving.jpg
wonderlake.jpg
woronzof.jpg

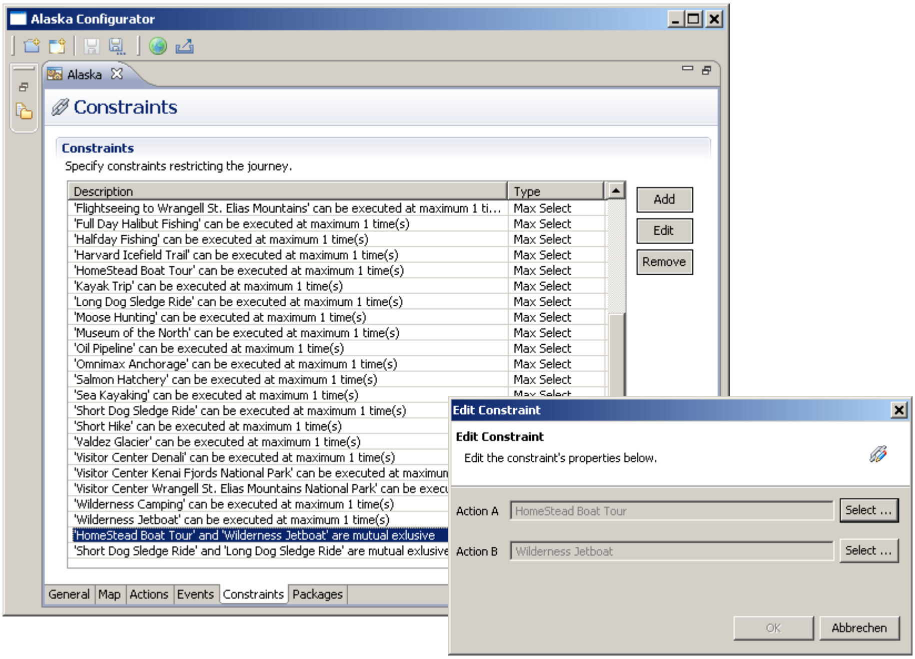
Actions

All activities, accommodations and routes are listed here.

Name	Location	Availability	Reliability
Alaska Bear Viewing Flight	Anchorage	10%	99%
Anchorage Motel			
Anchorage Museum			
Arctic Circle Air Advent			
Backcountry Hiking			
Bus Tour			
Columbia Glacier Cruise			
Delta Junction - Fairba			
Delta Junction Motel			
Delta Junction - Tok			
Denali - Anchorage			
Denali - Fairbanks			
Denali Motel			
El Dorado Gold Mine To			
End of Alaska Highway			
Fairbanks Historical Cit			
Fairbanks Motel			
Full Day Halibut Fishing			
Glenallen - Anchorage			

Fig. 8. Defining Actions and Locations with Alaska Configurator

During experiment execution participants were guided by the experimental workflow. After presenting them with a survey (cf. Fig. 12), half the students obtained configuration California with few constraints, while the other half obtained the same configuration with many constraints. The students then planned and executed a journey to California. Each step that was performed while planning and executing was logged for later investigation and detailed analysis. Having completed their California journeys, subjects planned and executed a journey to Alaska.



Constraints

Constraints

Specify constraints restricting the journey.

Description	Type
'Flightseeing to Wrangell St. Elias Mountains' can be executed at maximum 1 ti...	Max Select
'Full Day Halibut Fishing' can be executed at maximum 1 time(s)	Max Select
'Halfday Fishing' can be executed at maximum 1 time(s)	Max Select
'Harvard Icefield Trail' can be executed at maximum 1 time(s)	Max Select
'HomeStead Boat Tour' can be executed at maximum 1 time(s)	Max Select
'Kayak Trip' can be executed at maximum 1 time(s)	Max Select
'Long Dog Sledge Ride' can be executed at maximum 1 time(s)	Max Select
'Moose Hunting' can be executed at maximum 1 time(s)	Max Select
'Museum of the North' can be executed at maximum 1 time(s)	Max Select
'Oil Pipeline' can be executed at maximum 1 time(s)	Max Select
'Omnimax Anchorage' can be executed at maximum 1 time(s)	Max Select
'Salmon Hatchery' can be executed at maximum 1 time(s)	Max Select
'Sea Kayaking' can be executed at maximum 1 time(s)	Max Select
'Short Dog Sledge Ride' can be executed at maximum 1 time(s)	
'Short Hike' can be executed at maximum 1 time(s)	
'Valdez Glacier' can be executed at maximum 1 time(s)	
'Visitor Center Denali' can be executed at maximum 1 time(s)	
'Visitor Center Kenai Fjords National Park' can be executed at maximum	
'Visitor Center Wrangell St. Elias Mountains National Park' can be execu	
'Wilderness Camping' can be executed at maximum 1 time(s)	
'Wilderness Jetboat' can be executed at maximum 1 time(s)	
'HomeStead Boat Tour' and 'Wilderness Jetboat' are mutual exlusive	
'Short Dog Sledge Ride' and 'Long Dog Sledge Ride' are mutual exlusive	

Edit Constraint

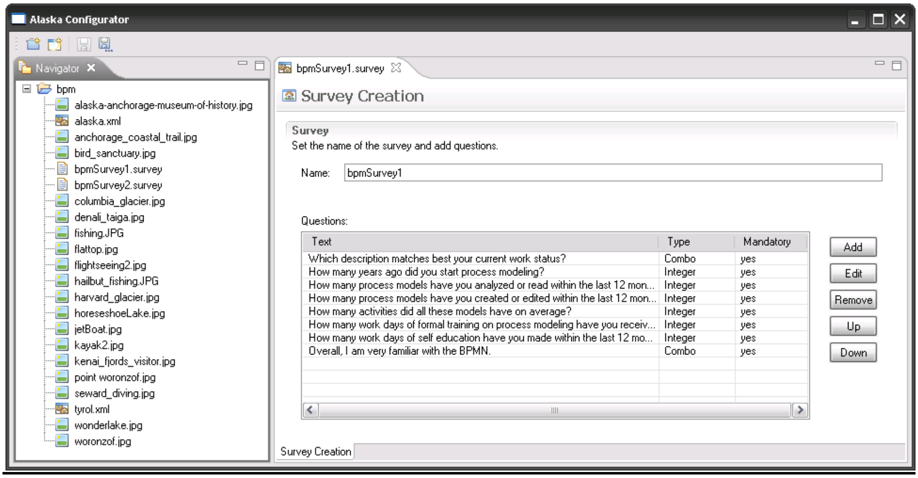
Edit Constraint

Edit the constraint's properties below.

Action A: HomeStead Boat Tour

Action B: Wilderness Jetboat

Fig. 9. Defining Constraints with Alaska Configurator



- bpm
  - alaska-anchorage-museum-of-history.jpg
  - alaska.xml
  - anchorage_coastal_trail.jpg
  - bird_sanctuary.jpg
  - bpmSurvey1.survey
  - bpmSurvey2.survey
  - columbia_glacier.jpg
  - denali_taiga.jpg
  - fishing.JPG
  - flattop.jpg
  - flightseeing2.jpg
  - hailbut_fishing.JPG
  - harvard_glacier.jpg
  - horeseshoeLake.jpg
  - jetBoat.jpg
  - kayak2.jpg
  - kenai_fjords_visitor.jpg
  - point woronzof.jpg
  - seward_diving.jpg
  - tyrol.xml
  - wonderlake.jpg
  - woronzof.jpg

Survey Creation

Survey

Set the name of the survey and add questions.

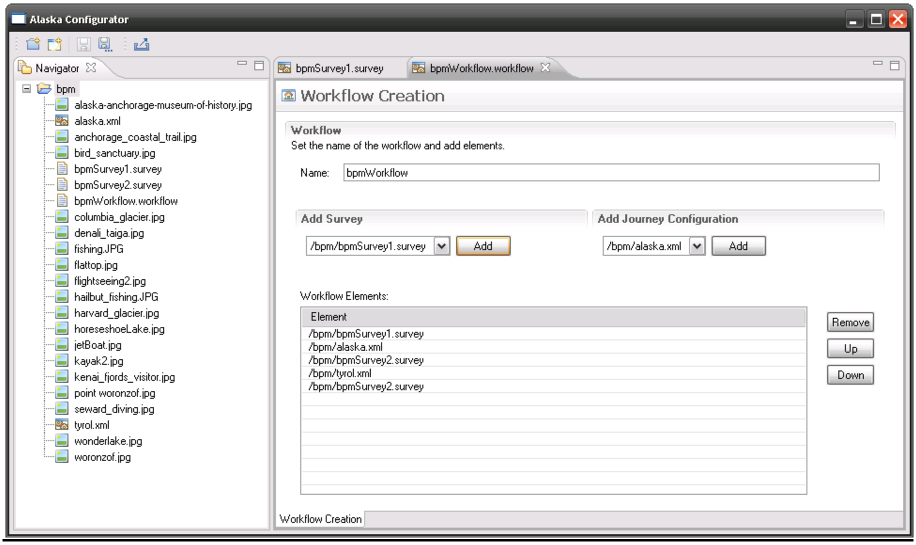
Name: bpmSurvey1

Questions:

Text	Type	Mandatory
Which description matches best your current work status?	Combo	yes
How many years ago did you start process modeling?	Integer	yes
How many process models have you analyzed or read within the last 12 mon...	Integer	yes
How many process models have you created or edited within the last 12 mon...	Integer	yes
How many activities did all these models have on average?	Integer	yes
How many work days of formal training on process modeling have you receiv...	Integer	yes
How many work days of self education have you made within the last 12 mo...	Integer	yes
Overall, I am very familiar with the BPMN.	Combo	yes

Fig. 10. Questionnaire Builder

After the planning session researchers were supported in analyzing the journeys by enabling them to replay each journey step by step, using Alaska Analyzer (cf. Fig. 13).



- bpm
  - alaska-anchorage-museum-of-history.jpg
  - alaska.xml
  - anchorage_coastal_trail.jpg
  - bird_sanctuary.jpg
  - bpmSurvey1.survey
  - bpmSurvey2.survey
  - bpmWorkflow.workflow
  - columbia_glacier.jpg
  - denali_taiga.jpg
  - fishing.JPG
  - flattop.jpg
  - flightseeing2.jpg
  - hailbut_fishing.JPG
  - harvard_glacier.jpg
  - horeseshoeLake.jpg
  - jetBoat.jpg
  - kayak2.jpg
  - kenai_fjords_visitor.jpg
  - point woronzof.jpg
  - seward_diving.jpg
  - tyrol.xml
  - wonderlake.jpg
  - woronzof.jpg

Workflow

Set the name of the workflow and add elements.

Name: bpmWorkflow

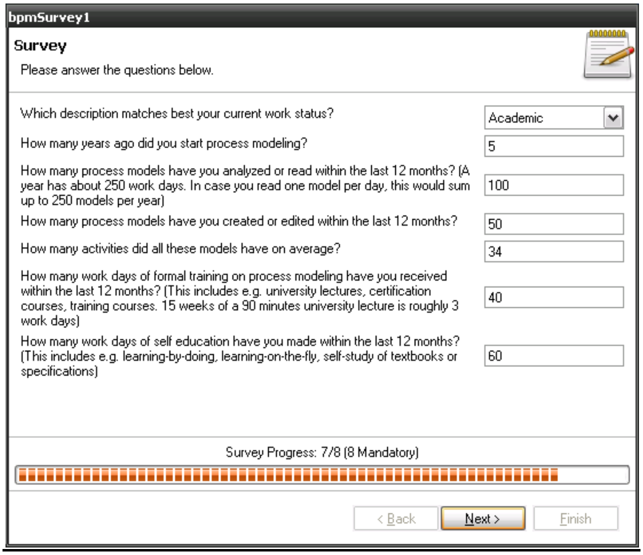
Add Survey: /bpm/bpmSurvey1.survey

Add Journey Configuration: /bpm/alaska.xml

Workflow Elements:

Element
/bpm/bpmSurvey1.survey
/bpm/alaska.xml
/bpm/bpmSurvey2.survey
/bpm/tyrol.xml
/bpm/bpmSurvey2.survey

Fig. 11. Designer for Experimental Workflows



Survey

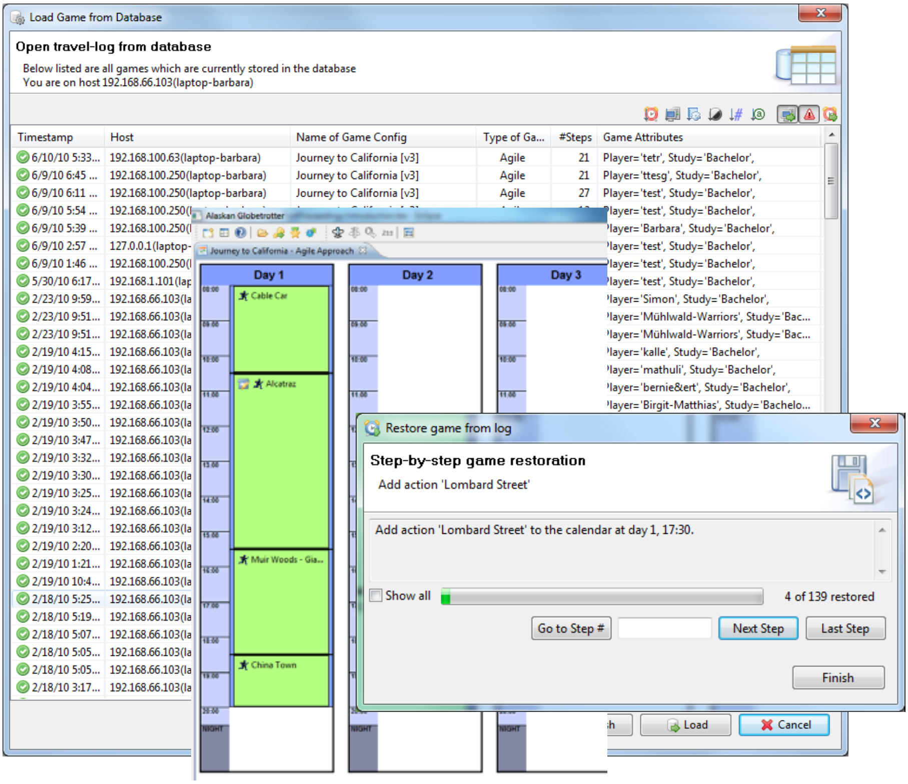
Please answer the questions below.

Which description matches best your current work status?	Academic
How many years ago did you start process modeling?	5
How many process models have you analyzed or read within the last 12 months? (A year has about 250 work days. In case you read one model per day, this would sum up to 250 models per year)	100
How many process models have you created or edited within the last 12 months?	50
How many activities did all these models have on average?	34
How many work days of formal training on process modeling have you received within the last 12 months? (This includes e.g. university lectures, certification courses, training courses. 15 weeks of a 90 minutes university lecture is roughly 3 work days)	40
How many work days of self education have you made within the last 12 months? (This includes e.g. learning-by-doing, learning-on-the-fly, self-study of textbooks or specifications)	60

Survey Progress: 7/8 (8 Mandatory)

Fig. 12. Survey Generated Using Questionnaire Builder

Alaska Simulator, including a test configuration, extensive documentation and screencasts can be downloaded from http://www.alaskasimulator.org. For detailed information on the results of controlled experiments conducted using Alaska Simulator we refer to [26,27].



Load Game from Database

Open travel-log from database

Below listed are all games which are currently stored in the database
You are on host 192.168.66.103(laptop-barbara)

Timestamp	Host	Name of Game Config	Type of Ga...	#Steps	Game Attributes
6/10/10 5:33...	192.168.100.63(laptop-barbara)	Journey to California [v3]	Agile	21	Player='tetr', Study='Bachelor',
6/9/10 6:45 ...	192.168.100.250(laptop-barbara)	Journey to California [v3]	Agile	21	Player='ttesg', Study='Bachelor',
6/9/10 6:11 ...	192.168.100.250(laptop-barbara)	Journey to California [v3]	Agile	27	Player='test', Study='Bachelor',
6/9/10 5:54 ...	192.168.100.250(l				Player='test', Study='Bachelor',
6/9/10 5:39 ...	192.168.100.250(l				'layer='Barbara', Study='Bachelor',
6/9/10 2:57 ...	127.0.0.1(laptop-				'layer='test', Study='Bachelor',
6/9/10 1:46 ...	192.168.100.250(l				'layer='test', Study='Bachelor',
5/30/10 6:17...	192.168.1.101(lap				'layer='test', Study='Bachelor',
2/23/10 9:59...	192.168.66.103(la				'layer='Simon', Study='Bachelor',
2/23/10 9:51...	192.168.66.103(la				'layer='Mühlwald-Warriors', Study='Bac...
2/23/10 9:51...	192.168.66.103(la				'layer='Mühlwald-Warriors', Study='Bac...
2/19/10 4:15...	192.168.66.103(la				'layer='kalle', Study='Bachelor',
2/19/10 4:08...	192.168.66.103(la				'layer='mathuli', Study='Bachelor',
2/19/10 4:04...	192.168.66.103(la				'layer='bernie&ert', Study='Bachelor',
2/19/10 3:55...	192.168.66.103(la				'layer='Birgit-Matthias', Study='Bachelo...
2/19/10 3:50...	192.168.66.103(la				
2/19/10 3:47...	192.168.66.103(la				
2/19/10 3:32...	192.168.66.103(la				
2/19/10 3:30...	192.168.66.103(la				
2/19/10 3:25...	192.168.66.103(la				
2/19/10 3:24...	192.168.66.103(la				
2/19/10 3:12...	192.168.66.103(la				
2/19/10 2:20...	192.168.66.103(la				
2/19/10 1:21...	192.168.66.103(la				
2/19/10 10:4...	192.168.66.103(la				
2/18/10 5:25...	192.168.66.103(la				
2/18/10 5:19...	192.168.66.103(la				
2/18/10 5:07...	192.168.66.103(la				
2/18/10 5:05...	192.168.66.103(la				
2/18/10 5:05...	192.168.66.103(la				
2/18/10 3:17...	192.168.66.103(la				

Alaskan Globetrotter

Journey to California - Agile Approach

Restore game from log

Step-by-step game restoration

Add action 'Lombard Street'

Add action 'Lombard Street' to the calendar at day 1, 17:30.

□ Show all 4 of 139 restored

Fig. 13. Replaying Journeys with Alaska Analyzer

5 Related Work

Most existing work about flexibly dealing with exceptions, changes, and uncertainty in the context of PAISs and related technologies is strongly *design-centered*, i.e., aiming at the development of tools, techniques, and methodologies. For overviews and discussions of these approaches, see [9,28,29].

Only few empirical investigations exist that aim to establish the suitability of the various proposed artifacts. Closely related to this paper is our previous work, which provides empirical insights into the use of declarative processes [26]. In particular, it investigates how well users can cope with the gained flexibility provided by declarative approaches, especially when processes become rather complex in terms of constraints. The effect of events on how end users can cope with declarative approaches, in turn, is investigated in [27]. A theoretical discussion on declarative versus imperative approaches is provided in [30]. In [10], the results of a controlled experiment comparing a traditional workflow management system and case-handling are described. The systems are compared with respect to their associated implementation and maintenance efforts. In turn, the impact of workflow technology on PAIS development and PAIS maintenance is investigated in [31]. Other empirical works with respect to PAISs mainly deal with establishing their contribution to business performance improvement, e.g. [32,33], and the way end-users appreciate such technologies, e.g., [34,35].

Worth mentioning here is a stream of research that relates to so-called *change patterns* [9]. It provides a framework for the qualitative comparison of existing flexibility approaches.

6 Conclusion

Alaska Simulator Toolset supports researchers in systematically comparing different approaches for decision deferral by means of controlled experiments. This paper describes the concepts underlying Alaska Simulator and shows how AST supports the different patterns for decision deferral. In addition, the paper discusses factors having a potential impact on the suitability of a particular approach for decision deferral, which can be investigated using AST. Moreover, the paper demonstrates how AST supports researchers in conducting controlled experiments. In particular, *Alaska Configurator* allows researchers to design journey configuration executable with Alaska Simulator and supports them in designing experimental workflows guiding users during the experiment. *Alaska Simulator* allows to plan and execute journeys making use of different approaches for decision deferral. *Alaska Analyzer*, in turn, supports detailed analysis of data and replaying journeys step by step.

Future work includes the planning and execution of additional controlled experiment with AST. On the tool side, the development of a dashboard simplifying the supervision of experiments is planned.

Acknowledgements. We thank Felix Schöpf for his work on the experimental workflow support, and Michael Schier on the Late Binding and Late Modeling Support in Alaska Simulator.

References

1. Poppendieck, M., Poppendieck, T.: Implementing Lean Software Development. Addison Wesley Longman, Amsterdam (2006)
2. Weske, M.: Business Process Management: Concepts, Methods, Technology. Springer, Heidelberg (2007)
3. Van der Aalst, W., Jablonski, S.: Dealing with workflow change: Identification of issues an solutions. Int'l Journal of Comp. Systems, Science and Engineering 15, 267–276 (2000)
4. Mutschler, B., Reichert, M., Bumiller, J.: Unleashing the effectiveness of process-oriented information systems: Problem analysis, critical success factors and implications. IEEE Trans. on Systems, Man, and Cybernetics 38, 280–291 (2008)
5. Reichert, M., Dadam, P.: ADEPT$_{flex}$ – Supporting Dynamic Changes of Workflows Without Losing Control. JIIS 10, 93–129 (1998)

6. Van der Aalst, W., Weske, M., Grünbauer, D.: Case handling: A new paradigm for business process support. Data and Knowledge Engineering 53, 129–162 (2005)
7. Pesic, M., Schonenberg, M., Sidorova, N., van der Aalst, W.: Constraint-Based Workflow Models: Change Made Easy. In: Proc. CoopIS 2007, pp. 77–94 (2007)
8. Sadiq, S., Sadiq, W., Orlowska, M.: A Framework for Constraint Specification and Validation in Flexible Workflows. Information Systems 30, 349–378 (2005)
9. Weber, B., Reichert, M., Rinderle-Ma, S.: Change patterns and change support features -enhancing flexibility in process-aware information systems. In: Data and Knoweldge Engineering, pp. 438–466 (2008)
10. Weber, B., Mutschler, B., Reichert, M.: Investigating the Effort of Using Business Process Management Technology: Results from a Controlled Experiment. Science of Computer Programming 75, 292–310 (2009)
11. Myers, G.J.: A Controlled Experiment in Program Testing and Code Walkthroughs/Inspections. ACM Commun. 21, 760–768 (1978)
12. Lott, C.M., Rombach, H.D.: Repeatable Software Engineering Experiments for Comparing Defect-Detection Techniques. Emp. Software Eng. 1, 241–277 (1996)
13. van der Aalst, W.M.P., Pesic, M.: DecSerFlow: Towards a truly declarative service flow language. In: Bravetti, M., Núñez, M., Tennenholtz, M. (eds.) WS-FM 2006. LNCS, vol. 4184, pp. 1–23. Springer, Heidelberg (2006)
14. Reijers, H.: Design and Control of Workflow Processes. LNCS, vol. 2617. Springer, Heidelberg (2003)
15. Gebauer, J., Schober, F.: Information system flexibility and the cost efficiency of business processes. J. of the Assoc. for Information Systems 7, 122–147 (2006)
16. Mans, R.S., Russell, N.C., van der Aalst, W.M.P., Bakker, P.: Schedule-aware workflow management systems. In: Proc. PNSE 2009, pp. 81–96 (2009)
17. Strong, D.M., Miller, S.M.: Exceptions and exception handling in computerized information processes. ACM Trans. Inf. Syst. 13, 206–233 (1995)
18. Boehm, B.W.: Software Engineering Economics. Prentice Hall, Englewood Cliffs (1981)
19. Cohn, M.: Agile Estimating and Planning. Prentice Hall Professional, Englewood Cliffs (2006)
20. Beck, K.: eXtreme Programming Explained. Addison-Wesley Longman, Amsterdam (2000)
21. Erdogmus, H.: The Economic Impact of Learning and Flexibility on Process Decisions. IEEE Software 22, 76–83 (2005)
22. Highsmith, J., Cockburn, A.: Agile Software Development: The Business of Innovation. Computer 34, 120–122 (2001)
23. Lindvall, M.: Empirical Findings in Agile Methods. In: Proc. XP/Agile Universe 2002, pp. 197–207 (2002)
24. Boehm, B.W., Turner, R.: Balancing Agility and Discipline: A Guide for the Perplexed. Addison Wesley Professional, Reading (2003)
25. van der Aalst, W.M.P., Reijers, H.A., Weijters, A.J.M.M., van Dongen, B.F., de Medeiros, A.K.A., Song, M., Verbeek, H.M.W.E.: Business process mining: An industrial application. Inf. Syst. 32, 713–732 (2007)
26. Weber, B., Reijers, H.A., Zugal, S., Wild, W.: The declarative approach to business process execution: An empirical test. In: van Eck, P., Gordijn, J., Wieringa, R. (eds.) CAiSE 2009. LNCS, vol. 5565, pp. 470–485. Springer, Heidelberg (2009)
27. Weber, B., Pinggera, J., Zugal, S., Wild, W.: Events during business process execution: An empirical test. In: Proc. ER-POIS 2010 (2010)
28. Kammer, P., Bolcer, G., Taylor, R., Hitomi, A., Bergman, M.: Techniques for Supporting Dynamic and Adaptive Workflow. CSCW 9, 269–292 (2000)

29. Reijers, H., Rigter, J., van der Aalst, W.: The case handling case. Int'l J. of Cooperative Information Systems 12, 365–391 (2003)
30. Fahland, D., Lübke, D., Mendling, J., Reijers, H.A., Weber, B., Weidlich, M., Zugal, S.: Declarative versus imperative process modeling languages: The issue of understandability. In: Proc. BPMDS 2009 and EMMSAD 2009. LNBIP, vol. 29, pp. 353–366 (2009)
31. Kleiner, N.: Supporting usage-centered workflow design: Why and how? In: Desel, J., Pernici, B., Weske, M. (eds.) BPM 2004. LNCS, vol. 3080, pp. 227–243. Springer, Heidelberg (2004)
32. Oba, M., Onoda, S., Komoda, N.: Evaluating the quantitative effects of workflow systems based on real case. In: Proc. HICSS 2000 (2000)
33. Reijers, H., van der Aalst, W.: The effectiveness of workflow management systems: Predictions and lessons learned. Int'l J. of Inf. Management 25, 458–472 (2005)
34. Bowers, J., Button, G., Sharrock, W.: Workflow from within and without: technology and cooperative work on the print industry shopfloor. In: Proc. CSCW 1995, pp. 51–66 (1995)
35. Poelmans, S.: Workarounds and distributed viscosity in a workflow system: a case study. ACM SIGGROUP Bulletin 20, 11–12 (1999)