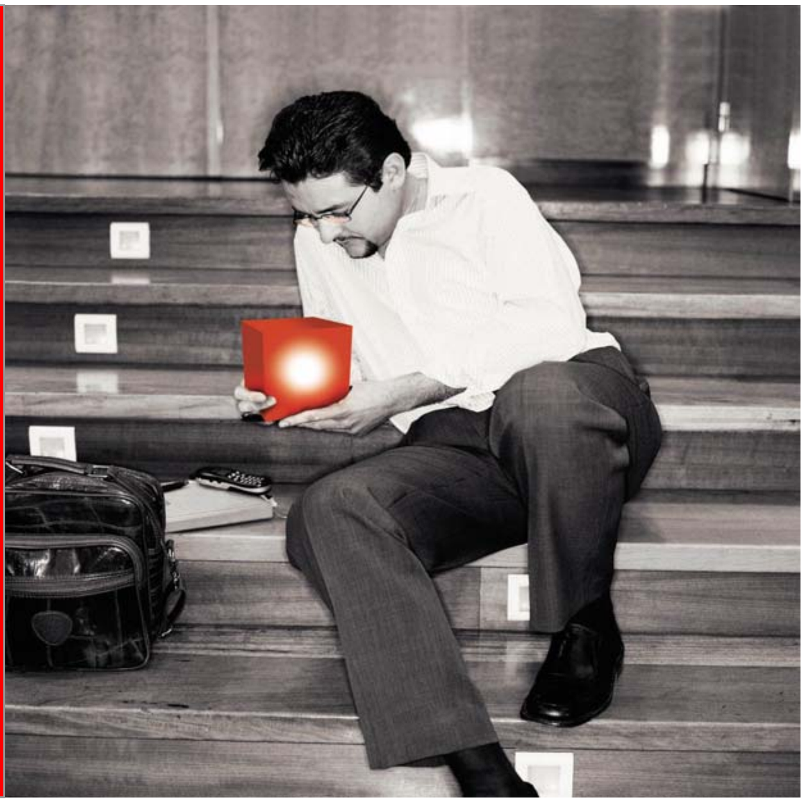
staffware®

THE POWER OF PROCESS



Workflow patterns according to Staffware

WE DELIVER | YOU SUCCEED

Introduction

This document responds to the workflow patterns as described by W.M.P. van der Aalst, A.H.M. ter Hofstede, B. Kiepuszewski, and A.P. Barros and the evaluation of the various available products (http://tmitwww.tm.tue.nl/research/patterns/documentation.htm). It describes how Staffware can be used to comply with the different patterns.

Flexibility

If enterprises follow the old 80/20 rule, then it can be broadly stated that 80% of processes are fixed and can be readily automated. That leaves the 20% that are more fluid and are difficult to manage effectively. It is these 20% of exceptions where an 'ad hoc' approach is required. By allowing sub-processes to branch off and resolve themselves without disruption to the central process functionality, organizations can control sub-processes in a dynamic environment. Business benefits include:

- Greater visibility of the enterprise process map
- Improved productivity
- Reduction in unit costs
- Early visibility of process bottlenecks

Staffware and ad-hoc processing

For most business processes, in addition to the recognized steps or activities that make up the process, there are number of recognized exceptions that may occur along the way. For example in a loan application process a recognized exception will be that the applicant fails the credit check. The point at which these types of exceptions may occur can be anticipated and can be built into the automated business process. On the other hand there are other types of exception that could occur at any point within the process. Examples of these types of exception are: a person moves house, a person gets married and changes their name, a person dies, notification is received that a loan application is being investigated for fraud. In some processes the number of identified exceptions may be numerous. Any attempt to build handling for such exceptions at every point with the process would result in a very large spaghetti-like process that would be almost impossible to enhance or maintain.

The way in which these types of exceptions need to be handled may also vary; some may need separate processes to be initiated, some may need the main process to be suspended while some other activity takes place, some may require that process data is modified to take account of changed circumstances while others may mean that the current state of the process may need to change.

The Staffware Process Suite provides functionality to support these kinds of exceptions. In cases where continuing to work on the business process after an exception has occurred may invalidate the end result of the process, Staffware allows cases of a process to be suspended while the exception is handled. Once the exception has been handled through some external processing such as a dedicated process, then the original process can be reactivated or closed as appropriate. For situations where data relevant to the process has been changed, such as an application changing names, it is fairly straightforward to modify the central case data using Staffware events. Staffware Process Suite v.2 provides functionality that can also propagate these changes into workitems that are already in users' workqueues.

The most complex situation is where the occurrence of an exception means that the current status of the process is invalid. For example:

- The process should now be further advanced than it is - it needs to skip some steps
- The process should be less advanced than it is - it needs to undo some steps
- The process needs to be in a separate, possibly parallel, branch of the process
- The process needs to be repositioned in a specialized piece of exception handling process.

Staffware Process Suite includes "Jump To Step" functionality that allows one or more outstanding steps to be withdrawn and a new set sent out. Since the status of the data at the point where the old steps were withdrawn may be different to that required by the new steps, the Jump To Step functionality allows case data to be updated as part of the operation.

Staffware and prediction

The current business climate is very unpredictable, so those organizations that can anticipate what will happen in the future are going to be better placed to compete than those that cannot. Through its unique Prediction technology, the Staffware Process Suite helps organizations to forecast what will happen and when it will happen. For every live Staffware case Prediction forecasts the expected route through the process, what steps are likely to require processing and when they are expected to occur. Applications and users alike can query the Prediction data at any time to find out what work is expected in the future. Prediction uses the process definition and the existing case data to make as accurate a forecast as possible. The query interface is rich and flexible so that the information can be used for many different purposes.
Within an organization Prediction can be used to forecast whether an individual, team or department is too heavily or lightly loaded with work. Based on this information decisions can be made about how the work is allocated or reallocated. Since it is possible to forecast when work items are expected to be processed, Prediction can be used to inform customers when activities or deliveries might be completed.

In environments where missing deadlines can incur financial penalties, Prediction can be used to forecast whether time critical activities will be completed in time. If Prediction forecasts that some deadlines might be missed, then work could be reallocated to balance the load or could be used to calculate the expected exposure. Prediction provides the necessary information for businesses to react to situations before they occur. In this way they are able to take advantage of forthcoming opportunities and to avoid anticipated problems.

Staffware Process Monitor can help organizations analyze how efficiently processes have been running and hence show where process improvement needs to be targeted. In addition to its work forecasting capabilities Prediction can be used to help optimize processes. When a process has been remodeled as part of process improvement or optimization, Prediction can be used to simulate the running of cases in the remodeled process. To achieve this, instead of using the continuous Prediction functionality described above, 'On Demand' Prediction is used.

‘On Demand’ Prediction takes a specified process definition and uses simulation data provided to it rather than live case data, then generates a list of the expected work steps including their timings. This information can then be used to help determine whether the expected process improvements can be achieved.

What makes the Staffware Process Suite unique?

Staffware has developed the Process Suite with all its recent enhancements culminating in SPS v2, with the following design parameters which, when combined together, deliver a truly unique BPM solution.

Recognizing the importance of people - People do business with people. Staffware has always maintained that people are the heart and soul of an organization. In implementing the Process Suite solution based on how you conduct business with your stakeholder community, your people become inextricably linked to, and supported by, that business backbone.

Recognizing the importance of processes - This design parameter is a natural extension of the previous one as it is people who design your business processes and it is people who execute your processes. If people are the heart and soul of your organization, your business processes become your lifeblood. Your processes represent your distinctive mark that sets you apart from your competition and is a primary reason for your success. The Staffware Process Suite uses your business processes as your process integration roadmap for integrating your underlying application and data infrastructure. Staffware lets you concentrate on the ‘Why’ of your business, not on the ‘What’.

Recognizing the importance of existing IT investments - Your processes do not operate in a vacuum. A variety of other systems may be involved in any one business process, with data needing to be moved between them and with process routing decisions being based on information from them. By supplying powerful integration capabilities, the Staffware Process Suite allows much of this interaction to be ‘hidden’ from the user – leaving them free to concentrate on the task in hand. Staffware can be used for completely automated business processes (Straight Through Processing or STP), with human intervention only being needed for handling approvals or exceptions.

Ease of implementation - The Staffware Process Suite effectively conceals the underlying complexity of an integration solution by the ease with which business processes and procedures are defined. In fact, in building your business process blueprint, business specialists who are intimately involved in managing your operations work alongside your IT staff in using the Process Definer tool. The rich graphic output and the accompanying documentation provide a powerful implementation tool to guide your enterprise process integration effort.

Scalability - In implementing a BPM integration solution, your process applications are implemented a layer above your application infrastructure. These new ‘supervisory’ process applications manage the execution of the various applications as required. The applications can be distributed across your network, and just as your process can be sub-divided to encompass additional business requirements, your process applications can be deployed within the context of these additional sub-processes. With the Staffware Process Suite your organization is unlimited to the degree of intra-enterprise and inter-enterprise integration that it can achieve.

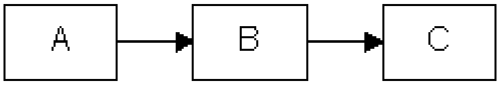
Specialists in BPM - Staffware supplies BPM and workflow software and services - nothing else. Our aim is always to work with the systems and products you have developed or chosen as solutions to specific applications; Staffware’s expertise is in supplying an Independent Process Layer that can join your systems and your people together in one business process.

In summary the combination of people-based process and external application-based process, added to 7*24 processing and transactionality, and operating under a highly scalable architecture, allows the Staffware Process Suite to stand ahead of any other product in existence today. Add to this the continued high levels of investment in research and development to drive the product forward and Staffware can justifiably claim to be the leaders in the BPM market. Independent research and benchmarks by analists like Forrester, Aberdeen, DTI, Yphise and Doculabs support this claim.

Patterns

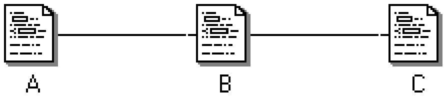
Pattern 1: Sequence

Sequence is the most basic workflow pattern. It is required when there is a dependency between two or more tasks so that one task cannot be started (scheduled) before another task is finished.



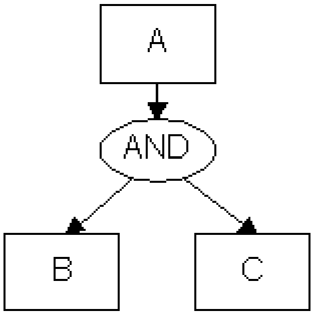
Description:
An activity in a workflow process is enabled after the completion of another activity in the same process.

Staffware Implementation:



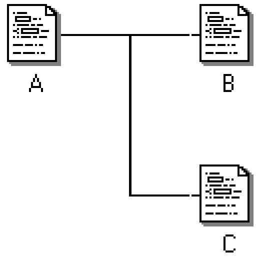
Pattern 2: Parallel split

Parallel split is required when two or more activities need to be executed in parallel. Parallel split is easily supported by most workflow engines except for the most basic scheduling systems that do not require any degree of concurrency.



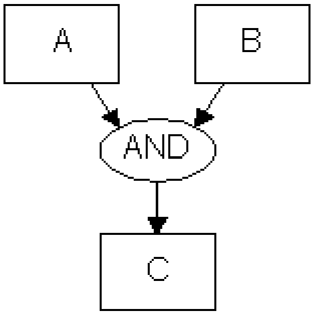
Description:
A point in the workflow process where a single thread of control splits into multiple treads of control which can be executed in parallel, thus allowing activities to be executed simultaneously or in any order.

Staffware implementation:



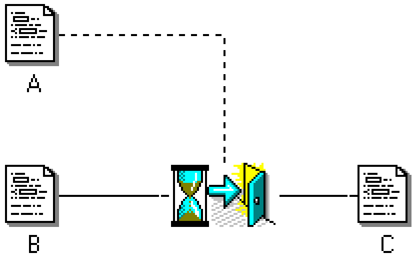
Pattern 3: Synchronization

Synchronization is required when an activity can be started only when two parallel threads complete.



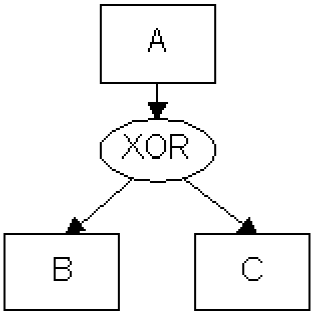
Description:
A point in the workflow process where multiple parallel sub processes/activities converge into one single thread of control, thus synchronizing multiple threads.

Staffware Implementation:

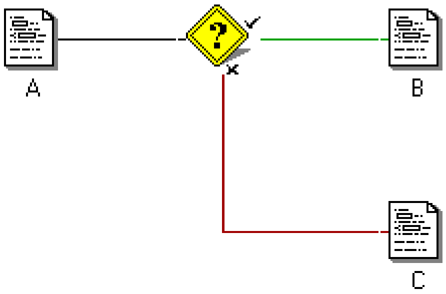


Pattern 4: Exclusive choice

A point in the workflow process where, based on a decision or workflow control data, one of several branches is chosen.

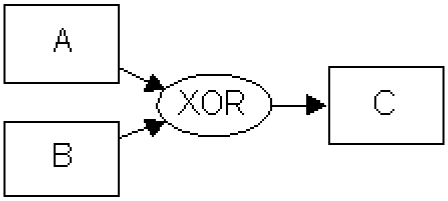


Staffware implementation:



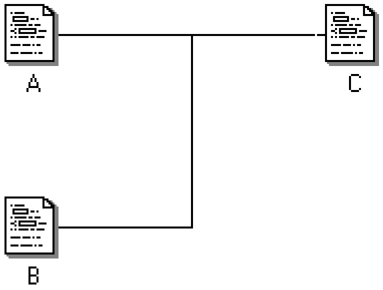
Pattern 5: Simple merge

Merge is required if we want to merge to alternative execution paths into one.



Description:
A point in the workflow process where two or more alternative branches come together without synchronization. In other words the merge will be triggered once any of the incoming transitions are triggered.

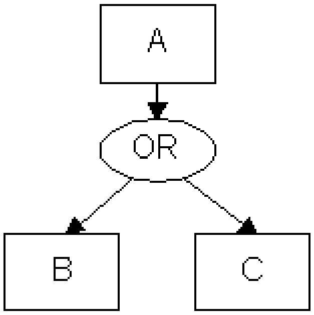
Staffware implementation:



N.B. Only one instance of "C" can be active.

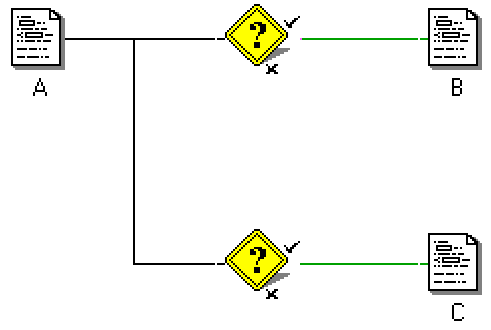
Pattern 6: Multi-choice

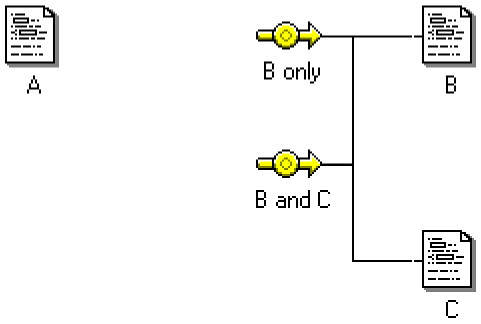
Pattern Exclusive Choice assumes that exactly one of the alternatives is selected and executed, i.e. it corresponds to an exclusive OR. Sometimes it is useful to deploy a construct, which can choose multiple alternatives from a given set of alternatives. Therefore, we introduce the (inclusive) multi-choice.



Description:
A point in the workflow process where, based on a decision or workflow control data, one or more branches are chosen.

Staffware implementation:





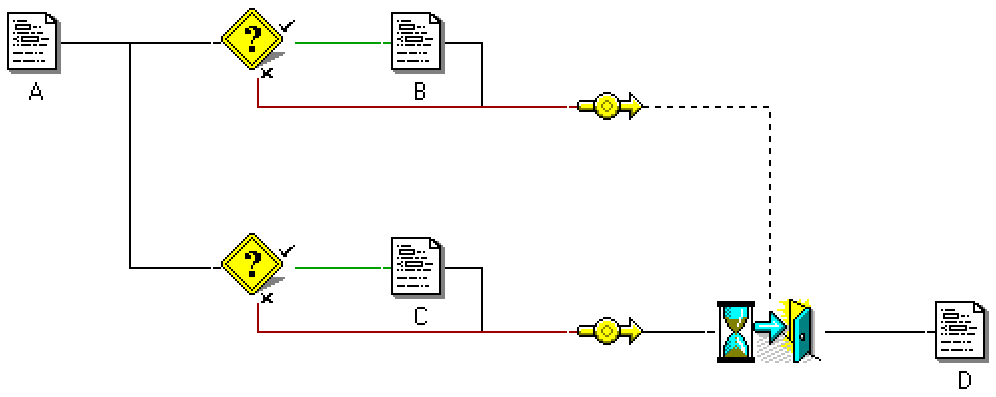
```
GOTOSTEP( "BONLY", "", 0 )
```

Pattern 7: Synchronizing merge

The Multiple Choice pattern can be handled quite easily by today's workflow products. Unfortunately, the implementation of the corresponding merge construct (OR-join) is much more difficult to realize. The OR-join should have the capability to synchronize parallel flows and to merge alternative flows. The difficulty is to decide when to synchronize and when to merge. Synchronizing alternative flows leads to potential deadlocks and merging parallel flows may lead to the undesirable multiple execution of activities.

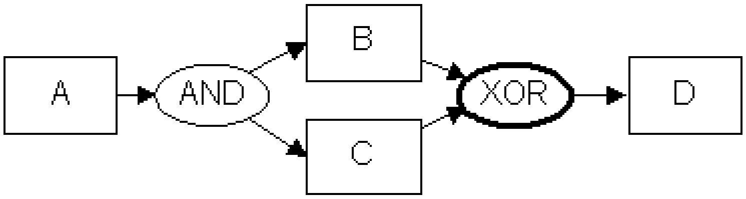
Description:
A point in the workflow process where multiple paths converge into one single thread. If more than one path is taken, synchronization of the active threads needs to take place. If only one path is taken, the alternative branches should converge without synchronization.

Staffware implementation:



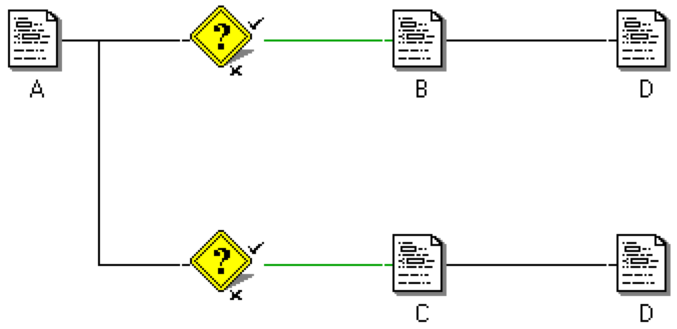
Pattern 8: Multi-merge

This pattern aims to address the problem mentioned in Simple Merge, which is the situation when more than one incoming transition of a merge is being activated.



Description:
Multi-merge is a point in a workflow process where two or more branches converge without synchronization. If more than one branch gets activated, possibly concurrently, the activity following the merge is started **once for every incoming branch that gets activated** (i.e. in the diagram above, *D* will be instantiated twice).

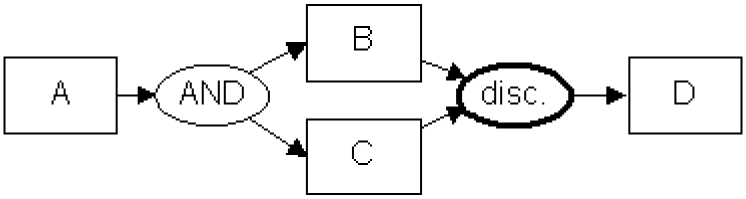
Staffware implementation:



This is the only way for Staffware to provide the possibilities: ABCDD, ACBDD, ABDCD, and ACDBD.

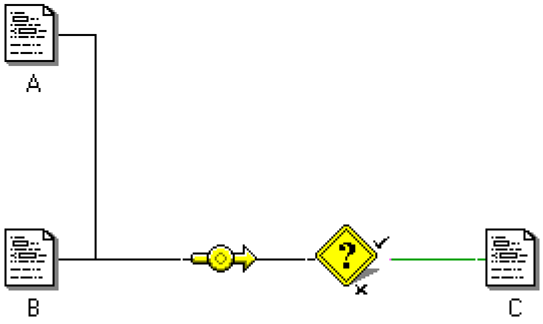
Pattern 9: Discriminator

This pattern can be seen as the converse of the multi-merge. It should be implied when our semantics is that only one activity should be instantiated after merge.



Description:
The discriminator is a point in a workflow process that waits for a number of incoming branches to complete before activating the subsequent activity. From that moment on it waits for all remaining branches to complete and “ignores" them. Once all incoming branches have been triggered, it resets itself so that it can be triggered again.

Staffware implementation:

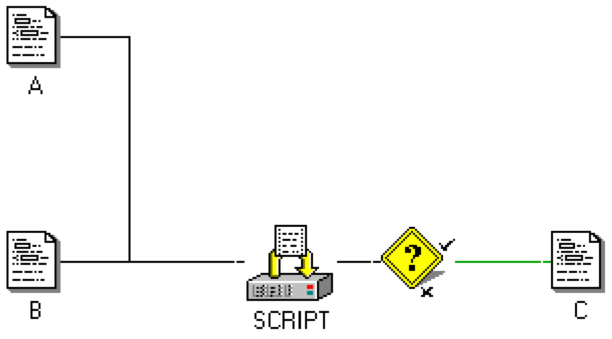


The release script of step “A” and “B” is:

```
IF SYNC = SW_NA
        SYNC := "YES"
ELSE
        SYNC := SW_NA
ENDIF
```

If more than one input is required, a counter can be added in order to reset after ‘n’ inputs. Even in another point in the workflow process the counter can be reset.

Another implementation is displayed below.



In this way all incoming paths are handled by the script step. The same script content can be used.

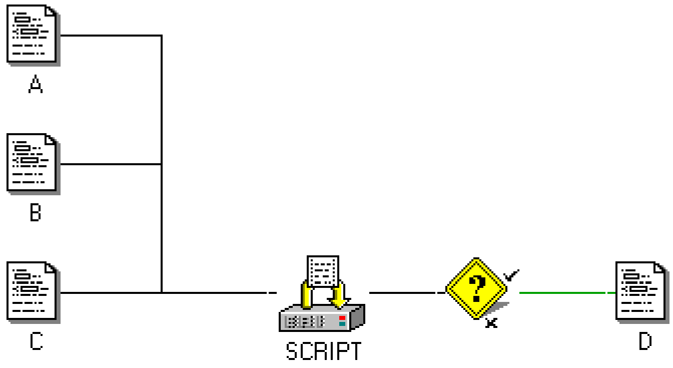
N-out-of-M Join

The following pattern can be seen as a generalization of the basic Discriminator. We would like to synchronise N threads from M incoming transitions.

Description:

N-out-of-M Join is a point in a workflow process where M parallel paths converge into one. The subsequent activity should be activated once N paths have completed. Completion of all remaining paths should be ignored. Similarly to the discriminator, once all incoming branches have "fired", the join resets itself so that it can fire again.

Staffware implementation:



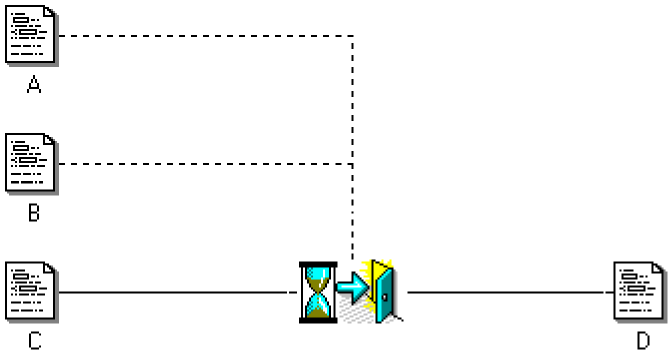
The content of the script used in this example is:

```
;Increase counter
COUNT := COUNT + 1

;See if this is triggered for
N times
IF( COUNT = N )
        SYNC := "YES"
ELSE
        SYNC := "NO"
ENDIF

;Check for reset
IF( COUNT = M )
        COUNT = 0
ENDIF
```

The condition after the script checks whether SYNC equals “YES”.
If N equals M, the standard Staffware Wait-step can be used:

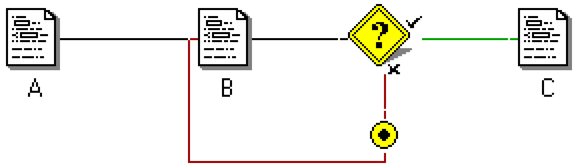


Pattern 10: Arbitrary cycles

During the workflow analysis/design time it is undesirable to be exposed to various syntactical constrains of the specific workflow enactment tool such as for example that there should be only one entry and one exit point to the loop. In fact, to achieve proper abstraction, the workflow engine should allow for execution of unconstrained models as typically they are much more suitable for the end-users to trace the execution of the process.

Description:
A point in a workflow process when one or more activities can be done repeatedly.

Staffware implementation:



Pattern 11: Implicit termination

Another example of the requirement imposed by some of the workflow engines on a modeler is that the workflow model is to contain only one ending node, or in case of many ending nodes, the workflow model will terminate when the first one is reached. Again, most business models do not follow this pattern - it is more natural to think of a business process as terminated once there is nothing else to be done.

Description:
A given sub-process should be terminated when there is nothing else to be done. In other words, there are no active activities in the workflow and no other activity can be made active (and at the same time the workflow is not in deadlock).

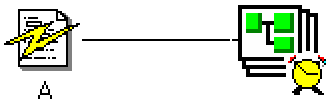
Staffware implementation:

As stated above.

Pattern 12: Multiple instances without synchronization

Description:
Within the context of a single case (i.e., workflow instance) multiple instances of an activity can be created, i.e., there is a facility to spawn off new threads of control. Each of these threads of control is independent of other threads. Moreover, there is no need to synchronize these threads.

Staffware implementation:



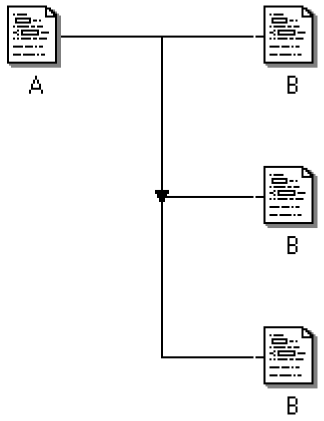
By using the trigger “A” a dynamic number of sub-cases can be started. By setting the property: ‘*no wait for finish*’, no synchronization is applied.
The event “A” can be triggered any time it is needed.

Pattern 13: Multiple instances with a priori design time knowledge

We would like to be able to create many instances of one activity. The number of instances is known at the design time.

Description:
For one case an activity is enabled multiple times. The number of instances of a given activity for a given case is known at design time.

Staffware implementation:

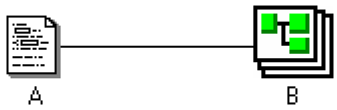


Pattern 14: Multiple instances with a priori runtime knowledge

We would like to be able to generate many instances of an activity. The number of instances is dynamic, i.e. not known at the design time. It is known though at some point before all instances need to be executed. You can think about this pattern as a FOR loop that instantiates an activity.

Description:
For one case an activity is enabled multiple times. The number of instances of a given activity for a given case is variable and may depend on characteristics of the case or availability of resources, but is known at some stage during runtime, before the instances of that activity have to be created.

Staffware implementation:



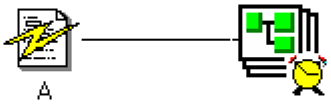
With the '*multiple sub-procedure call step*' a number of sub-cases can be started. The number and types of the sub-cases can dynamically be determined using arrays. Mapping of data is done via a template procedure.

Pattern 15: Multiple instances without runtime knowledge

We would like to be able to generate many instances of an activity. The number of instances is dynamic, i.e. not known at the design time nor it is known at any stage during the execution of the process before all these instances needs to be activated. You can think about this pattern as a WHILE loop that instantiates an activity.

Description:
For one case an activity is enabled multiple times. The number of instances of a given activity for a given case is not known during design time, nor it is known at any stage during runtime, before the instances of that activity have to be created.

Staffware implementation:



By using the trigger "A" a dynamic number of sub-cases can be started.
The event "A" can be triggered any time it is needed.

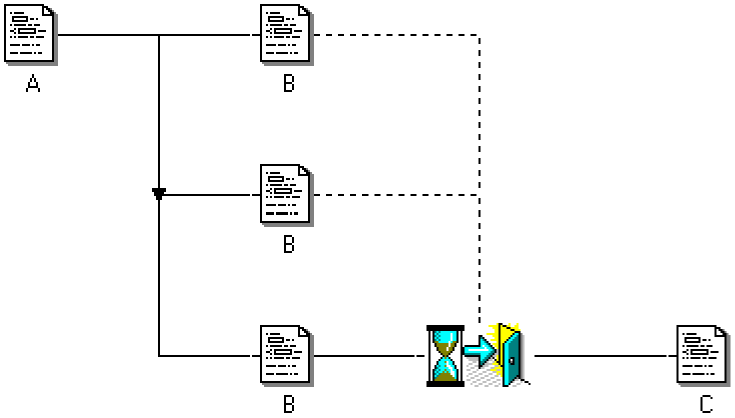
Pattern 16: Multiple instances requiring synchronization

The other multiple-instances related patterns do not consider the synchronization of created instances. For example, spawning off a variable number of subprocesses from the main process, as supported by Visual WorkFlo and I-Flow, does only launch multiple instances without considering synchronization issues. But sometimes it is required to continue the process only after all instances are completed, possibly without any a priori knowledge of how many instances were created.

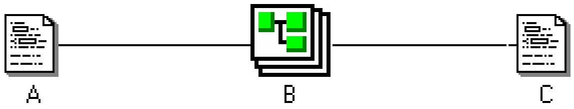
Description:
For one case an activity is enabled multiple times. The number of instances may not be known at design time. After completing all instances of that activity another activity has to be started.

Staffware implementation:

Depending on the method used for creating multiple instances the following constructs can be used:



The Wait-step object will synchronize all parallel branches connected to the Wait-step object.

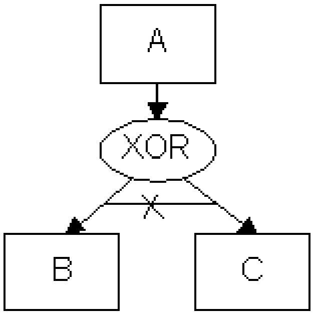


The '*multiple sub-procedure call step*' can synchronize the sub-cases.

Another way to synchronize a number of cases is the use the Graft-step.

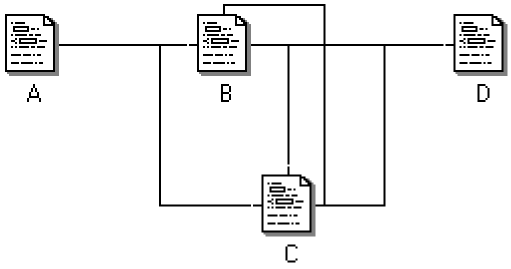
Pattern 17: Deferred choice

Moments of choice, such as supported by constructs as XOR-splits/OR-splits, in workflow management systems are typically of an explicit nature, i.e. they are based on data or they are captured through decision activities. This means that the choice is made a-priori, i.e. before the actual execution of the selected branch starts an internal choice is made. Sometimes this notion is not appropriate. We may want to have a situation where two threads are "enabled" for an execution (suppose one thread enables an activity A, the other enables activity B. We would like to see both activities on a worklist). Once one of the threads is started, the other thread should be disabled (i.e. once activity A gets started, B should disappear from the worklist).



Description:
A point in the workflow process where one of several branches is chosen. In contrast to the XOR-split, the choice is not made explicitly (e.g. based on data or a decision) but several alternatives are offered to the environment. However, in contrast to the AND-split, only one of the alternatives is executed. This means that once the environment activates one of the branches the other alternative branches are withdrawn. It is important to note that the choice is delayed until the processing in one of the alternative branches is actually started, i.e. the moment of choice is as late as possible.

Staffware implementation:



Audit Trail view:

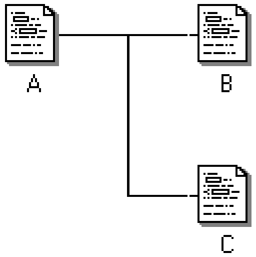
7-3-2003 13:36:42	Case started by swadmin@utrecht
7-3-2003 13:36:42	A processed to swadmin@utrecht
7-3-2003 13:36:42	A released by swadmin@utrecht
7-3-2003 13:36:42	B processed to swadmin@utrecht
7-3-2003 13:36:42	C processed to swadmin@utrecht
7-3-2003 13:36:48	C released by swadmin@utrecht
7-3-2003 13:36:48	B withdrawn from swadmin@utrecht
7-3-2003 13:36:48	D processed to swadmin@utrecht
7-3-2003 13:36:54	D released by swadmin@utrecht
7-3-2003 13:36:54	Case terminated normally

Pattern 18: Interleaved parallel routing

Patterns AND-split and AND-join are typically used to specify parallel routing. Most workflow management systems support true concurrency, i.e. it is possible that two activities are executed for the same case at the same time. If these activities share data or other resources, true concurrency may be impossible or lead to anomalies such as lost updates or deadlocks.

Description:
A set of activities is executed in an arbitrary order: Each activity in the set is executed, the order is decided at run-time, and no two activities are executed at the same moment (i.e. no two activities are active for the same workflow instance at the same time).

Staffware implementation:



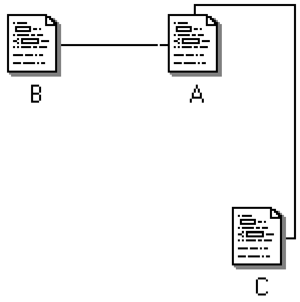
If “B” or “C” is opened, “C” or “B” needs to be locked. This can be done within Staffware, however there is no way of showing this graphically.

Pattern 19: Milestone

This pattern allows for testing whether a workflow process has reached a certain phase. Upon reaching some phase we would like to disable the activities that were previously enabled.

Description:
The enabling of an activity depends on the case being in a specified state, i.e.the activity is only enabled if a certain milestone has been reached which did not expire yet. Consider three activities *A*, *B*, and *C*. Activity *A* is only enabled if activity *B* has been executed and *C* has not been executed yet, i.e.*A* is not enabled before the execution *B* and *A* is not enabled after the execution *C*.

Staffware implementation:

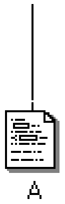


Pattern 20: Cancel activity

Typical processing after the activity is completed is to enable another activity. But some business process require a different action to be taken. Cancel Activity pattern recognizes that it may be valid to disable another activity as a result of an activity being completed.

Description:
An enabled activity is disabled, i.e. a thread waiting for the execution of an activity is removed.

Staffware implementation:



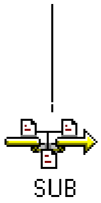
By drawing a line into the top of the Step-object, this activity is cancelled.

Pattern 21: Cancel case

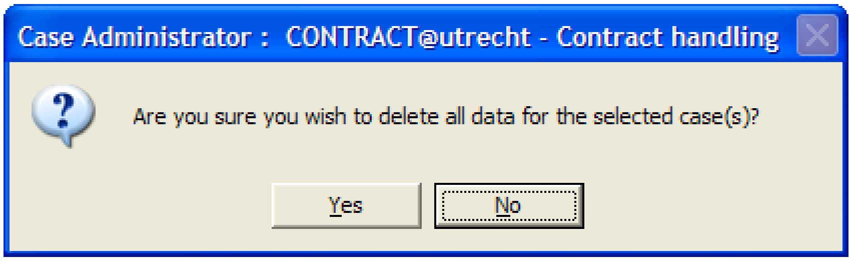
Description:
A case, i.e. workflow instance, is removed completely.

Staffware implementation:

Sub cases can be withdrawn completely by using the Withdraw action of the sub procedure.



Using the Case Administrator a case can be removed completely (purged).



Are you sure you wish to delete all data for the selected case(s)?

Purging cases can also be done using the API.

10 good reasons for buying Staffware process management technology

1. **Proven Technology**
   Staffware has focused exclusively on Business Process Technologies since the first release of the product in 1987, so we have tackled the technology roadblocks that now prevent our competitors developing successful, fast and scalable technology.
2. **Customer Endorsement**
   Staffware has over 2 million users spread across some 5000 organizations in just about every industry and government sector. Staffware customers win awards, and are not afraid to talk about the success they have had. Even the analysts say that Staffware is a clear leader in the field.
3. **Rapid Application Deployment**
   Staffware's BPM toolkit enables end users and technicians to work together, rapidly building solutions that are adaptable as the business requirements change.
4. **Effortless Scalability**
   Staffware's iProcess Engine is designed to grow as your business grows and enables you to leverage your existing infrastructure, whilst accommodating unexpected peaks with ease.
5. **True 24 x 7 Operation**
   24 X 7 operation is not just about facilitating on-line back-ups. It is about true uninterrupted systems operation. Staffware's technology allows you to add processes, tune the system, add people, load balance, take security back ups - without taking the system down. We also provide unique self-healing capabilities, which means that if you do lose a process for whatever reason, Staffware will restart it without user intervention.
6. **Proven Integration**
   Staffware provides countless ways of integrating with your enterprise and legacy applications. We support Java, Com+, EAI, Web Services, Custom Adapters, MOM, Database, to name a few. And if we can't provide it out of the box, Staffware's Process Integration tools are extensive. Staffware covers the full range of people-to-people, application-to-application, and all points in between.
7. **Business Activity Monitoring**
   You need to know, at a glance, how your business processes are performing so that you can take corrective action and keep the business running smoothly and effectively. Staffware provides comprehensive near real time reporting capabilities and alerts that can be customized to meet your key performance indicators and service level agreements. Staffware's unique prediction technology also gives you the power to see what will happen next as well.
8. **Global Support**
   Staffware operates in every major business center around the world, which means that, wherever you are, there is a skilled professional close by to assist. Our teams are supported by the most comprehensive dedicated support infrastructure in the industry - 24 hours a day, seven days a week.
9. **Process Frameworks**
   Staffware has a growing library of process frameworks that contain industry specific best practices, which reduce time to market, risk and overall project costs. The Frameworks provide up to 80% of required functionality, leaving you free to customize to suit your requirements.

10. StEP+

The Staffware Enterprise Partnership Program (StEP+) offers significant benefits to organizations committing to Staffware technology. These benefits include:

- Membership of the Product Advisory Board
- VP account management (Board to Board partnership)
- Preferential training and access to Professional Services
- Extended support relationship

About Staffware

Staffware is the leading Business Process Management (BPM) specialist and has over 2 million licensed users in 5,000 customer organizations within the Banking, Insurance, Telecommunications, Utilities, General Commercial, Manufacturing and Government sectors.

Staffware is headquartered in the UK, has offices in 16 countries and employs approximately 350 people. The company is profitable, debt-free, listed on the London Stock Exchange and is the leading global vendor focused exclusively on providing BPM solutions.
Staffware's BPM expertise has evolved over 15 years of automating and managing processes, so the interaction of people, applications and processes has remained central to its philosophy. Unlike the majority of its competitors, Staffware has focused exclusively on BPM and has a deep understanding and unique insight into the complex people-to-people, people-to-application and application-to-application interactions that make up business processes.

The Staffware Process Suite provides a complete set of tools to create, transform and streamline the internal and external processes and tasks of an organization. Staffware enables the creation of an Independent Process Layer that separates process from the application logic, facilitating rapid development and change to an organization's processes.
The Staffware Process Suite is an open and standards-based BPM solution that, from the outset was designed to provide ease of integration with existing IT infrastructures and applications.
The Staffware Process Frameworks consolidates Staffware's process technology, with embedded domain expertise to deliver industry-specific 'best practice' process frameworks to facilitate rapid implementation and reduce development cycles.

The SPS has been designed with three major objectives in mind:

- Linear scalability
- Very fast throughput
- The non-stop platform for Business Process Management

The cube; version 2

The SPS is a unique Process Management framework, which enables customers to pick-‘n’-mix the components they need to automate their process requirements without having to purchase products from different vendors.

iProcess Engine

The only BPM Engine for mission critical processes. The iProcess Engine is proven and highly scalable with inbuilt fault tolerant technology that ensures critical processes do not fail, as independently verified by Doculabs and the Unisys Benchmark Center. The iProcess Engine is the only BPM Engine that supports complex people intensive and system intensive processes ranging from traditional workflow and imaging applications through to very high volumes of automated processes - Straight Through Processing (STP).

Process Definer

A graphical process design tool that allows quick and easy development of the process without any need for coding and can be readily understood and used by non-IT personnel.

Process Monitor

Allows business managers to monitor process performance, helping to identify changing business conditions and supplying the information they need to make strategic business decisions.

Process Integrator

A range of options is available to deliver a wide variety of integration methods including full web services support and high-volume transactions based on Enterprise Application Integration.

Performance

Staffware iProcess is the only process engine that can demonstrate proven capability to handle 7.9 million process steps per hour and a user population of 30,000. This means that we can support your enterprise, not just individual departments.

Prediction

A unique capability in Staffware that automatically estimates the future processing path and timescale of work. This allows Business Users to see the future flow of cases, helping to determine resource requirements and work scheduling and enabling maximum leverage of existing infrastructure investments. Prediction also allows modelling of ‘what if’ scenarios to determine performance/financial exposure to the business.

Adhoc Process Management

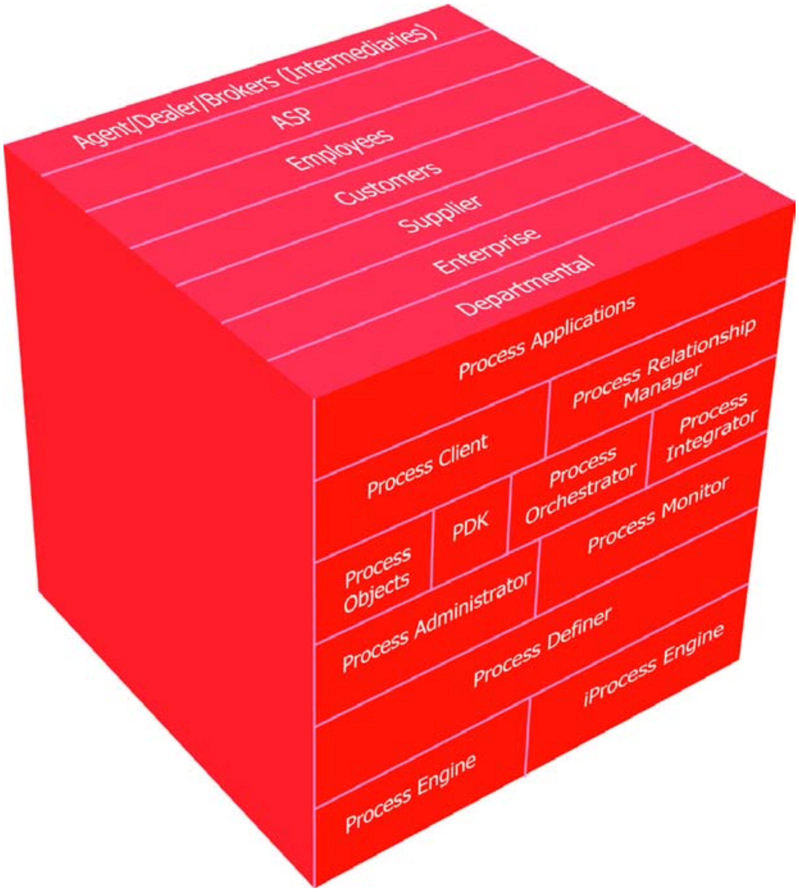
Controlled flexibility that allows the core process flow to instantly adapt to changes and demands from external inputs under full audit control. The system can thus react to exceptions and unplanned external events.

Process Orchestrator™

BPM and Web Services are the backbone of the Service Oriented Architecture and key for large enterprises in a dynamic real-time environment. It is increasingly recognized that the integration requirements of large enterprises cannot be satisfied by a messaging/adaptor approach. For the holistic view BPM and Web Services are now seen as key to binding complex organizations together. Process Orchestrator enables external applications to flexibly and dynamically link multiple tasks to the process case – allowing web services and BPM to work together closely linked yet with each taking responsibility for its own area of operation.

Performance

Staffware iProcess is the only process engine that can demonstrate proven capability to handle 7.9 million process steps per hour and a user population of 30,000. This means that we can support your enterprise, not just individual departments.



More information: www.staffware.com