zur Muehlen, Michael: Organizational Management in Workflow Applications. Information Technology and Management Journal. Kluwer Academic Publishers, 5 (2004) 3, pp. 271-291.
http://www.workflow-research.de/Publications/PDF/MIZU-ITM(2004).pdf

Organizational Management in Workflow Applications – Issues and Perspectives

MICHAEL ZUR MUEHLEN*

Wesley J. Howe School of Technology Management, Stevens Institute of Technology, USA

Abstract. Business processes automation requires the specification of process structures as well as the definition of resources involved in the execution of these processes. While the modeling of business processes and workflows is well researched, the link between the organizational elements and process activities is less well understood, and current developments in the web services choreography area completely neglect the organizational aspect of workflow applications. The purpose of this paper is to give an overview of the organizational aspects of workflow technology in the context of the workflow life cycle, to provide a review of existing work, and to develop guidelines for the design of a workflow-enabled organization, which can be used by both workflow vendors and users.

Keywords. Workflow Management, Resource Management, Organizational Modeling, Process Management, Performer Profile, Role Resolution

* Address for Correspondence: Wesley J. Howe School of Technology Management, Stevens Institute of Technology, Castle Point on the Hudson, Hoboken, NJ 07030, USA. Phone: 1.201.216.8293, Fax: 1.201.216.5385, e-mail: mzurmuehlen@stevens-tech.edu

Workflow management systems coordinate activities, resources, and data according to the formal representation of the process logic, the workflow model [48]. They can help realize efficiency potentials through the elimination of transport and wait times between process activities, and provide a detailed level of control over the assignment of work to process participants. The importance of human involvement in workflow applications has recently been pointed out by MOORE, who identified excessive activity automation and poor design of work assignment strategies as critical issues in workflow projects [33].

The roots of contemporary workflow management solutions can be found in the office automation prototypes of the 1970s (compare, e.g., [19, 51]), which were oriented towards the automation of human-centric processes. Recent developments in the area of web services choreography have discounted the organizational aspect of workflow solution and focus exclusively on the coordination of control flow structures. For instance, the proposed standards WSCI (a subset of BPML) [4], WSCL [5] and BPEL4WS [14] do not contain any notion of human activity performers. Instead, they are focused on the technical coordination of inter-enterprise processes, with little or no human intervention. This indicates a split in the traditionally diverse field of workflow management (for a discussion of different technical areas related to workflow management refer to [42]). While some vendors focus on the automation of mainly technical processes, such as the automated data exchange between disparate applications, other vendors stress the organizational aspect of their solutions and focus on human-centric processes [35].

The purpose of this paper is to give an overview of the organizational aspects of workflow applications, to provide a review of existing work, and to position this work in the context of the workflow life cycle. Based on this review we point out design guidelines that could improve the success of workflow applications.

In the following section we introduce a workflow life cycle model, and outline which resource-related tasks are performed during the different phases of the workflow life cycle. The rest of this paper is structured along the phases of this life cycle model. In section 2 we present a generic meta model for the organizational entities involved in a workflow application. In section 3 we discuss the specification of assignment and synchronization policies, and point out the relationship between organizational policies during build time and run time. In section 4 we present technical integration aspects that have to be addressed in the transition from build time to run time. Section 5 focuses on the maintenance of resource information at run time and the evaluation of workflow audit trail information for resource management purposes. Section 6 reviews related work and section 7 concludes the paper with a brief summary and an outlook on future work.

1 Resource Management in the Workflow Life Cycle

A *resource* (also known as actor, performer, or process participant [47]) is an entity that is assigned to a workflow activity and is requested at runtime to perform work in order to complete the objective of the activity. In this paper we focus only on resources that can actively contribute to the goal(s) of an activity. Passive resources, such as physical materials or information are part of the informational perspective of a workflow application. In this sense, our understanding of resource differs from the view of e.g. KWAN et al., who treat information used within workflow activities as a resource as well [29]. A *resource model* contains the definition of human and technical resources that are involved in the execution of a workflow model as workflow participants. While the resource model is a structured representation of organizational entities, it should be noted that both this model as well as the elements contained therein follow a life cycle and change over time. Therefore, a workflow management system not only needs to provide a mechanism to represent the organizational elements involved in the execution of workflows, but it also needs to provide mechanisms for continuous change within these elements.

The workflow life cycle reflects the desire to continuously improve the performance of business processes by monitoring the present, analyzing the past, and planning for the future. Procedure models for the design of workflow applications have been discussed by KWAN and BALASUBRAMANIAN [29], as well as WESKE et al. [46]. These models represent an extension of system analysis and design techniques and emphasize the separation between workflow logic and task logic. Explicit workflow life cycles have been proposed by GALLER [21] and HEILMANN [23], among others. These models provides detailed guidelines for the build time aspect of a workflow application, but provide little information about the system behavior at run time, especially the maintenance of resource information. The WISDM methodology developed by KWAN and BALASUBRAMANIAN explicitly refers to an organizational perspective that contains resource information (comparable to the organizational aspect of the widely discussed MOBILE architecture of JABLONSKI and BUSSLER [27]), but this perspective is mainly referenced in the initial analysis phase of the development cycle..

The workflow life cycle shown in Figure 1 builds on these approaches and extends them to explicitly represent the resource perspective during all phases of the workflow life cycle. The black circles in figure 1 indicate the sections of this paper that refer to the individual life cycle phases. The cycle starts with an initial *analysis* of the project goals, the environment of the future workflow application, and the organizational structures and rules surrounding the new system. This phase is followed by a *process design* phase, during which the overall process structure is engineered, the resulting workflow model is designed, and the resources involved in the process execution are specified. This includes the modeling of organizational structures as well as the definition of assignment policies and conflict resolution mechanisms. The completed workflow models are input of the *process implementation* phase. During this phase, the workflow solution is integrated with surrounding information systems. In terms of resource management, access to existing resource databases and security mechanisms need to be established. The synchronization of activity responsibilities with application access rights is of paramount importance in this phase. If errors are made regarding this synchronization, activities might be assigned to process participants who have insufficient access rights to execute the applications necessary for the fulfillment of the pending activity.

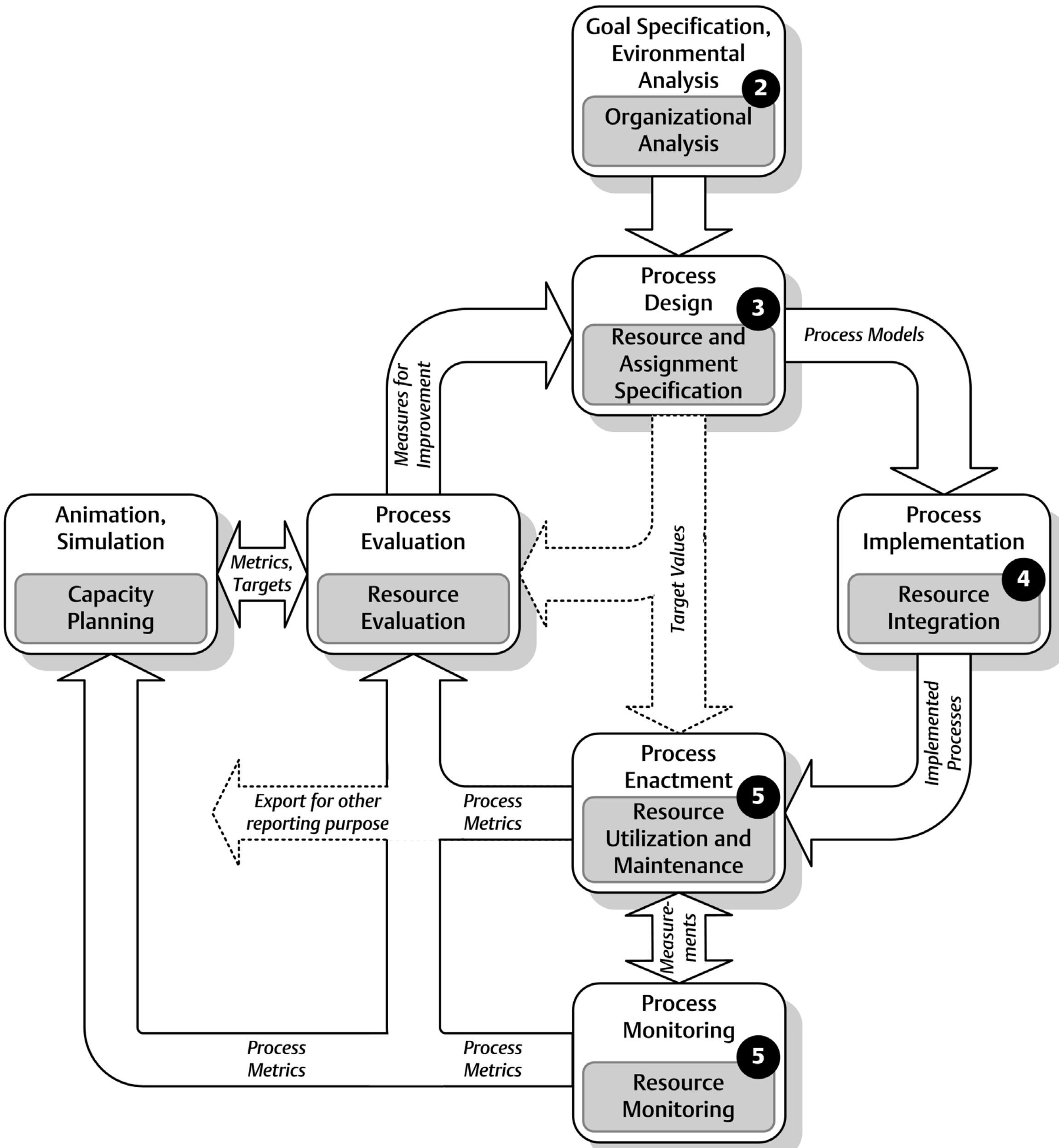


Figure 1: Workflow Life Cycle

During the *process enactment* phase, individual instances of the workflow model are created and coordinated by the workflow enactment service. *Internal* process participants (which are part of the workflow-enabled organization) are notified of pending activities through their work list and can select and activate these activities. Upon completion of an activity, control is handed back to the workflow enactment service. Depending on the nature of activities, process participants may be human resources, technical resources, or a combination of both. For internal workflow participants information about their capacity (in terms of permissible workload) and availability (in terms of schedule) can be obtained by the workflow engine. Activities that are assigned to internal process participants can be revoked and reassigned to other participants if their maximum processing time is exceeded, or if other exceptions occur [44]. *External* process participants are not under the control of the workflow-enabled organization. Information about the capacity and availability of these resources is rarely available to the workflow enactment service. These resources are typically informed about pending activities through an asynchronous medium, such as e-mail. In this case the revocation of activities is usually more difficult than for internal workflow participants.

Parallel to the process enactment phase *process monitoring* takes place. On the technical side the performance of the workflow management system itself is measured, while on the organizational side metrics such as the length of work queues, the idle time of resources, or the wait time of pending activities are supervised.

The *process evaluation* phase completes the workflow cycle. During this phase the execution of workflow instances is analyzed from an ex-post perspective, based on the execution protocols (the so-called audit trail). Results from this analysis can serve as the basis for the planning of resource capacities (i.e., how many performers are required to achieve certain quality of service levels). The capacity adjustments might be tested during an *animation and simulation* phase, where performance information from past workflow instances can be used as simulation parameters.

2 An Organizational Meta Model

In order to implement organizational control, workflow management systems maintain a model of the surrounding organizational structure. This model is typically kept separate from the workflow model, which focuses on the sequence of activities within a process. The division between a process model on the one side and a resource model on the other side fosters the separate evolution of both models, since the life-cycle of the resources within an enterprise typically varies from the life cycles of the enterprise's processes. In addition, the separation enables workflow designers to create workflow models that are independent of changes in the organizational structure of the enterprise, adding to their robustness.

Within the reference model of the Workflow Management Coalition [48], the management of resource information lies within the responsibility of the workflow enactment service. This reflects the fact that many workflow vendors have implemented proprietary resource management facilities for their workflow management systems, which are accessed through the workflow modeling environment, or through an auxiliary application.

In larger organizations managing resources inside the workflow enactment service may lead to problems if several workflow management systems are used for the implementation of a complex process [18]. These systems cannot share common information about enterprise resources, which leads to redundant resource specifications. In addition to this, information like the workload of single resources can only be determined if the distributed information of the individual workflow management systems is consolidated. If this information is not easily accessible, efficient use of the enterprise's resources can only be realized on a local level. Within the Object Management Group this fact has led to the discussion of a resource management facility [13], but to date no standard has emerged in this area.

During the build time of a workflow application, the workflow application designer has to design both the structure of the business process to be automated, and the structure of the resources that carry out the process. These resources may be managed by the workflow management system, or an external resource management facility. Resources and workflow activities are linked through the construct *role*. From a process perspective, a role represents the capabilities and privileges required for the proper execution of an activity. CURTIS et al. go one step further and define a role as "a coherent set of process elements to be assigned to an agent as a unit of functional responsibility" (see [15], p. 76). For a resource perspective, a role represents the combined capabilities and privileges of a process participant. Based on these two perspectives, the design of the resource model can follow two different directions. Using *workflow-driven resource modeling* the resource model is created according to the resources required by the workflow definition. In other words, the structure of the activities in the workflow model determines the roles of the workflow participants. Using *enterprise-driven resource modeling* the existing organizational structure of the enterprise (often captured in organizational charts) is created in the resource management portion of the workflow application, and a mapping between existing resources and workflow activities is performed.

In a previous study of commercial workflow management systems we have found the organizational modeling capabilities of workflow management systems to be very limited, a fact that has a negative impact on the adaptability of these systems to different enterprise structures found in actual organizations [34, 38]. Figure 2 shows a meta model for resource modeling, which serves both the workflow-oriented and the enterprise-oriented approach. It represents a hybrid meta model, which is based on our previous analysis of organizational meta models within workflow management systems [34].

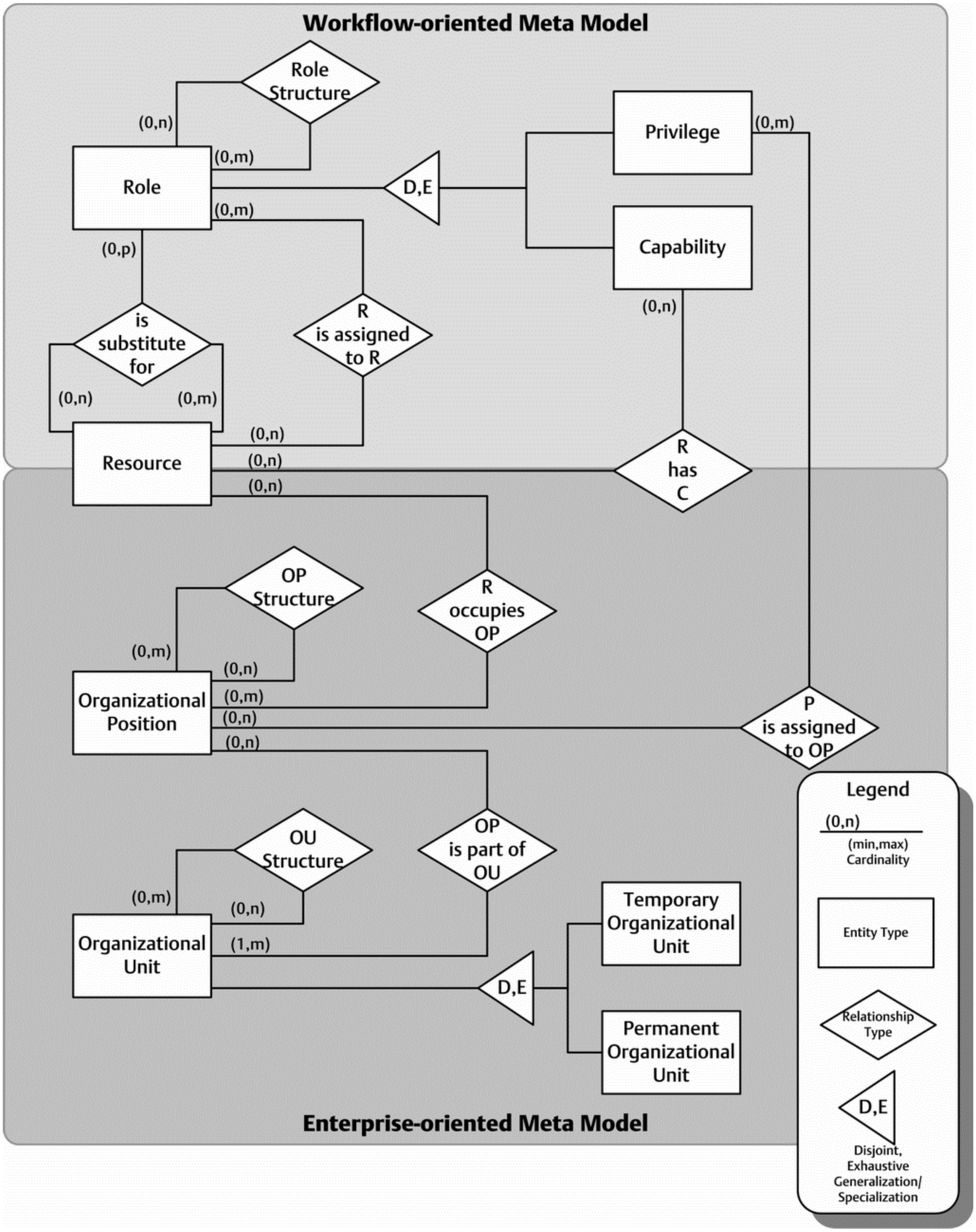


Figure 2: Reference Meta Model for Organizational Structures

At the heart of this resource meta model, a *resource* represents a technical or human workflow participant. The *role* entity is a named privilege granted to a resource, or a capability exhibited by a resource. The recursive *role structure* allows the combination of capabilities and privileges into more complex roles. Privileges and capabilities describe the possible actions a resource is permitted and/or qualified to perform. A *capability* is a direct property of a resource (e.g., "speaks Spanish"), and remains associated with the resource, even if the position of the resource in the enterprise changes. A *privilege* is a property of an organizational position, which can be occupied by one or more resources, which then inherit the privileges associated with the position (e.g., "authority to sign expense reimbursements over $ 50.000"). *Organizational positions* are the building blocks of the formal organizational structure of an enterprise, and their holders are granted the necessary authorities to perform the activities associated with these positions. Groups of organizational positions form larger *organizational units*, such as departments (permanent units) or project teams (temporary units). As long as there exist tasks within the enterprise that are not supported by a workflow management infrastructure, formal organizational positions and units are required to ensure accountability and responsibility for the execution of these tasks.

It is important to note that there is no single ideal model for a workflow-enabled organization. GALBRAITH'S contingency theory [20] states that there is no single optimal way to organize an enterprise. Instead, external factors such as customer expectations, market conditions, and regulatory aspects have to be considered. In workflow-based environments, the structure of the organizational entities as well as the structure of the business processes can be adjusted to find an optimal mix of resource and process efficiency. It should be noted that workflow technology does not resolve the goal conflict between resource and process efficiency, but it may help realize untapped efficiency potentials in an organization.

3 Definition of Assignment and Synchronization Policies

Assignment policies determine the strategy for work allocation among process participant candidates. Upon the instantiation of a workflow activity, the workflow enactment service places work items on the work lists of qualified performers who are determined using a process of *role resolution* (sometimes called staff resolution). Figure 3 shows the typical steps within the work assignment process. In step 1, the workflow enactment server determines the list of candidate resources based on the assignment policy and the available resources. A pending activity is placed on a shared worklist as a public work item. All qualified resources have access to this shared worklist. Once a candidate chooses to perform the pending activity (the grey arrow in Figure 3), the workflow enactment service removes the public work item from the shared worklist and places it on the private worklist of the designated performer. In a centralized workflow solution, this process can be implemented by manipulating the visibility properties of work items, similar to the concepts used in role-based access control (see e.g. [12]). This solution allows the workflow enactment service to hold a master table of all pending work items, with access restrictions based on the roles required by the individual activities and held by the individual performers. In a distributed environment the same effect can be achieved by sending pointers to work items on a central worklist as notifications instead of the actual work items. In this scenario, the work items are accessible through a web page, which is generated by the workflow enactment service for every work item.

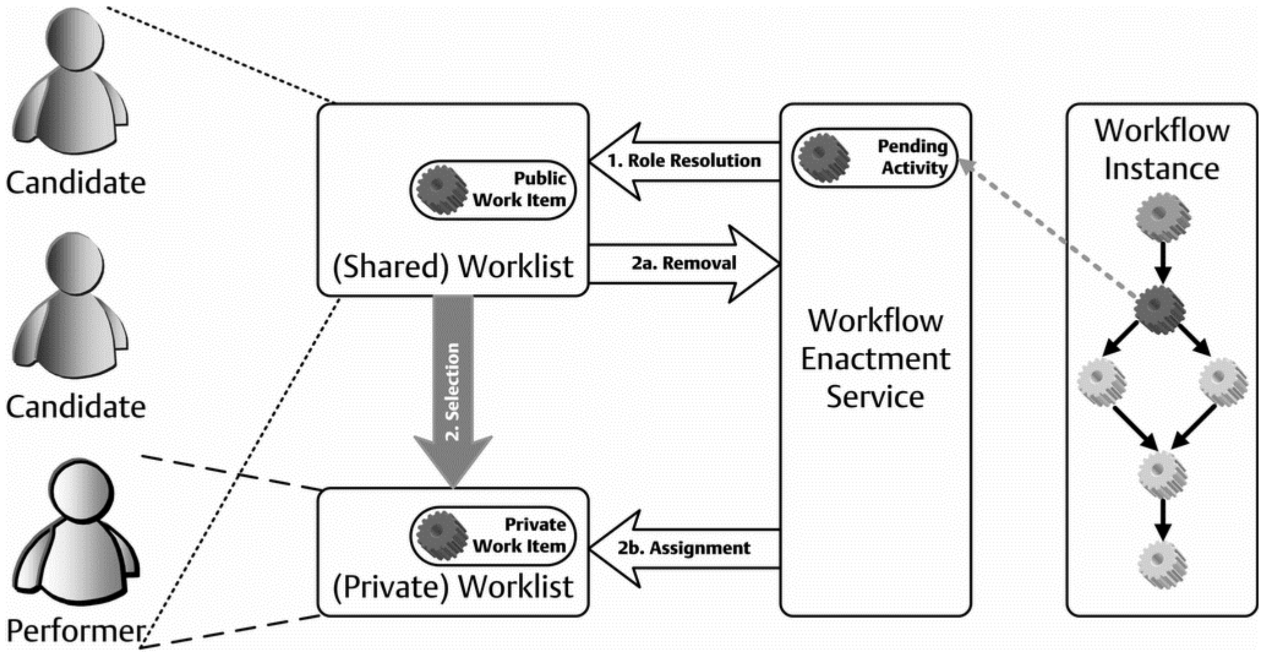


Figure 3: Role Resolution Procedure at Run Time

3.1 Assignment Policies

For the assignment of pending activities, different strategies can be implemented. These strategies have an impact on how the workflow enactment service prioritizes activities and notifies candidate performers. Figure 4 shows different work allocation strategies, and is based on the research of HOFFMANN et al. ([24], pp. 137-138).

The planning of new work items describes the behavior of the workflow enactment service, when new activities become executable. A *net change* strategy would only determine the assignment for the new work item, while a *re-planning* strategy would re-allocate all work items that have not yet been started, possibly removing work items from some performers' worklists and placing them on other worklists.

The workflow enactment service can notify performers about pending activities either upon the availability of these activities, at the latest start time of the activity, or at an arbitrary time between these two points. While human performers are typically informed upon availability of an activity, it may be desirable to delay the assignment of activities to technical resources, if the workflow management system is capable of optimizing the activity schedule based on the economic value or the priority of the activities. These parameters can either be assigned to a workflow instance at the time of instantiation, or the workflow enactment service can compute the relative importance of activities based on application data made available to the workflow management system (i.e., workflow-relevant data [49]). Example for this type of information could be the (internal) rating of a process customer or the revenue associated with a process instance.

Property	Possible Values			
Planning of new Work Items	Net Change	Re-Planning		Scheduling
Time of Notification	Upon Availability	Between Availability and Latest Start Time	At Latest Start Time	Scheduling
Queuing of new Work Items	Queue	Pool	Combination	Scheduling
Activity Execution	Individual	Collaborative		Task Properties
Decision Hierarchy	Final Assignment	Delegation Possible		Task Properties

Source: Modified from [24], pp. 137-138.

Figure 4: Assignment Policies

The *queuing of work items* can be performed either using a queue, ensuring that work items are selected in the order in which they become available; a pool, where resources can choose freely between available work items; or a combination of the two, where resources select a collection of work items at a time. The choice of queuing mechanism determines the degree to which the workflow enactment service can optimize the allocation of work items. If the priority of pending activities cannot be determined automatically, a pool-type distribution enables human performers to determine the priority of pending activities, while a queue-type distribution is recommendable if the optimal schedule of work items can be determined by the workflow enactment service.

The form of *activity execution* (individual or collaborative) prescribes, how many resources may select a work item for execution. While many workflow systems support only activities that are assigned to individual users, some research has been done on assigning work to teams, see e.g. [16, 17]. As a compromise, LEYMANN and ALTENHUBER propose the concept of *bundle* [30]. A bundle is modeled as a single activity in the workflow model, which may be instantiated multiple times at runtime, depending on conditions that are evaluated at runtime (the original IBM FlowMark workflow management system supported this concept [26], but this feature was dropped in the transition of this product to MQSeries Workflow and later Websphere MQ Workflow). While technically each of the bundle instances is assigned to only one resource, the parallel instantiation of bundle activities allows the modeling of collaborative activities to a limited extent.

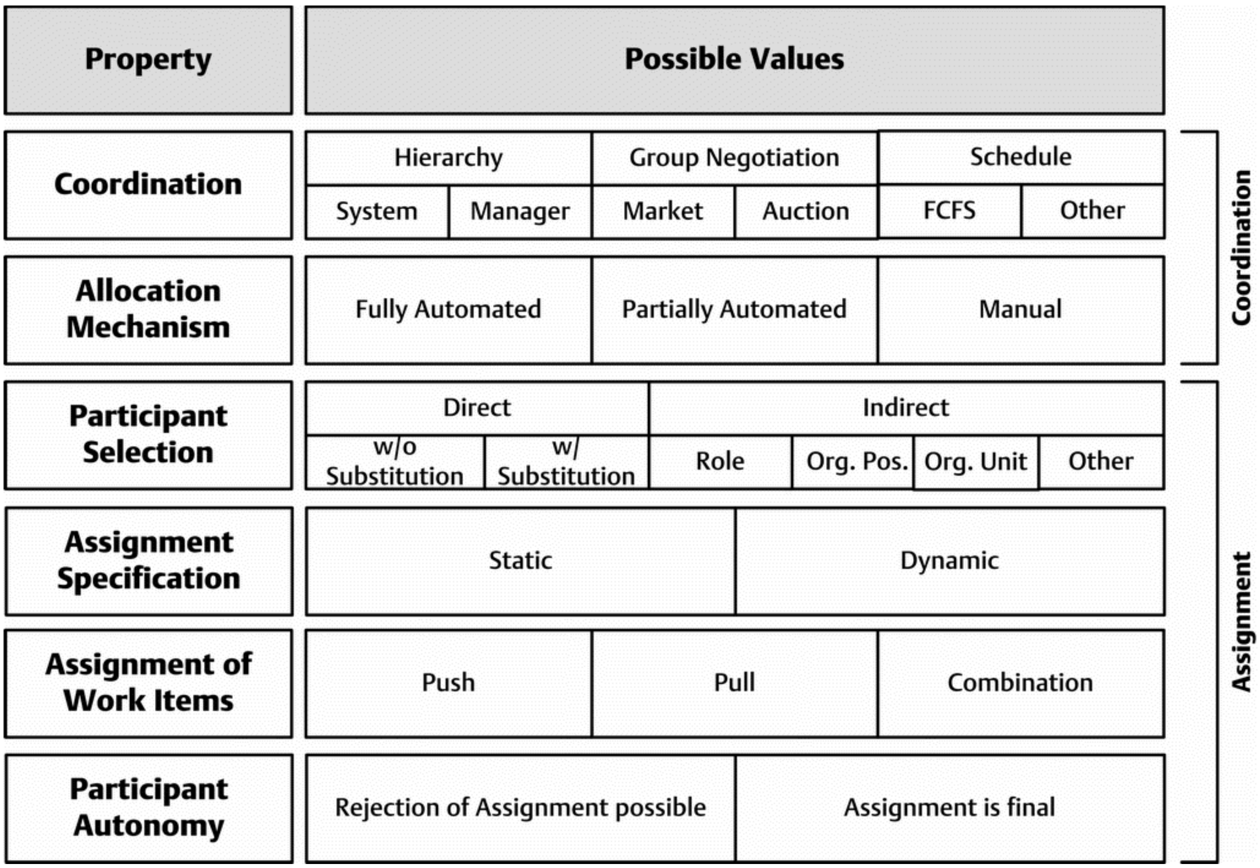
Finally, the *decision hierarchy* describes, whether an activity can be passed on from a workflow performer to another *delegate* performer. This act of substitution is common during the absence of a resource (when a deputy takes over some or all of the functions of the assignee). From a security perspective, delegation is a potentially harmful function. If the assignee is free to choose the ultimate recipient of an activity, this might endanger workflow constraints such as the separation of duty. For example, if an accounting process requires two separate members of the accounting department to authorize a purchase, the workflow enactment service would assign the second authorization activity to a member of the department who has not performed the first activity. However, if this member is allowed to delegate the activity, and chooses the performer of the first activity (since he or she is a qualified member of the accounting department), the process constraint "separation of duty" is violated. For this reason some workflow systems provide flags within their activity specification, which allow excluding activities from being delegated to performers who are not the original assignees. WAINER et al. have discussed the security problems of delegation and revocation in workflow systems in a recent technical report [44]. In a related paper, AHN et al. have proposed the use of existing role-based access control mechanisms to secure a web-based workflow management system [2].

The substitution of performers can be implemented at different levels. If a workflow application is deployed in a work environment where many resources share the same tasks and responsibilities, and workflow-controlled activities cover the majority of the overall tasks to be performed, the use of roles with multiple role-bearers creates an implicit substitute relationship, since multiple resources are qualified for the execution of an activity. If possible, organizational measures, such as the requirement to have an empty personal worklist at the end of the day, should limit the need for substitution mechanisms.

In a classic substitution scenario, the deputy can be granted access to the private worklist of the original performer, if this performer is absent. In some cases it may be desirable to limit the visibility of work items or create substitution relationships based on different roles that a process participant plays. For this reason, the meta model in Figure 2 contains a ternary substitution relationship between resource (recursive) and role. In this way, a resource may serve as the proxy for another resource, but this relationship can be restricted using one or more roles. For example, a manager might designate one of his subordinates as a substitute for his regular activities, while another manager is responsible for certain decision-making activities that govern the subordinates.

3.2 Synchronization Policies

Once a work item has been placed on a shared worklist, synchronization policies determine how it can be accessed by individual workflow participants. Figure 5 shows an overview of synchronization mechanism properties.



Source: Modified from [24], pp. 137-138.

Figure 5: Synchronization Mechanisms

The *coordination* of pending work items can either be hierarchical or market-based. In a hierarchical scenario, either the workflow enactment service assigns work items to individual performers (eliminating the need for a shared worklist), or in an intermediate step, a manager receives all pending work items and manually assigns them to the individual workflow participants. While the first strategy is useful if the workflow management system is capable of performing schedule optimization, the latter strategy reduces the "mechanistic" assignment of activities by a technical system, which may increase user acceptance (Action Technology's Coordinator, one of the first workflow systems, was criticized for its lack of flexibility, compare [40] for a discussion). If a market-based allocation policy is implemented, workflow participants compete for the right to perform a particular activity, either through auctions or other market mechanisms. While market-based work allocation is not a common feature in commercial workflow systems, it has been the subject of research studies by HARKER and UNGER [22], TAN and HARKER [43], as well as ALT et al. [3]. A very thorough discussion of auction protocols for the scheduling of decentralized resources was presented by WELLMANN et al. [45]. Finally, work items can be assigned according to a fixed schedule. In the most common case, they are assigned to a shared worklist, from where they can be selected by individual performers on a first-come-first-serve basis. However, other scheduling principles are theoretically possible.

The *allocation of work items* can be fully automated by the workflow management system, i.e. the system automatically determines the qualified performer(s) for an activity and assigns pending work items to these resources. In a partially automated scenario, the workflow management system may suggest qualified performers, but the ultimate assignment is performed by another decision instance, e.g. a team leader. In a manual scenario, the workflow management system would only show pending activities, but not perform any task allocation. It is up to the workflow-enabled organization to find organizational measures to ensure the actual execution of pending activities, or an external resource management application could perform the assignment.

The *participant selection* can be either direct (at the level of the individual performer), or indirect (using a proxy construct like role or organizational unit as the ultimate recipient of a work item). During direct assignment, the workflow system may check the availability of the selected resource, e.g., if the performer is currently logged into the system. If the resource is absent, the workflow system can determine a substitute and assign the work item to this resource, deliver the work item only to the initially chosen resource (e.g. if the activity is marked for direct assignment only) or raise an exception and escalate the process. Indirect selection is based on those elements of the resource meta model which allow the grouping of individual resources. In the meta model described in section 3.1, these entities are role, organizational position, and organizational unit, although other elements such as qualification or competency might be used as well.

The *specification of the resource requirements* can be either static or dynamic. Using the static assignment, an activity is associated with one or more instances of one or more resource meta model entities. For example, the activity "review customer order" could be assigned to the instance "sales manager" of the entity type "organizational position". This assignment is static in the sense that it does not vary with different workflow instances. Dynamic assignments take data from the current workflow instance into account for the selection of qualified performers. This data can either relate to the current workflow instance (e.g. the ID of the process initiator) or to the business objects processed in the current workflow instance (e.g. the customer number contained in an order document). Dynamic assignment allows a very detailed specification of assignment policies (e.g., all orders from a particular customer are handled by the same customer service agent), but require the tight integration of the workflow management system with external data sources. In practice, this type of assignment is found in embedded workflow solutions, where the workflow component has immediate access to all relevant system data.

The *assignment of pending activities* can either be performed in a push or pull manner. Using the push mechanism, resources signal their availability to the workflow system, which determines the next activity this performer should work on. This may be desirable in environments, when the workflow enactment service can optimize the scheduling of activities, the activities themselves are uniform, and the resources are mainly technical (without user intervention). In the example of a brokerage application, workflow participants were not given a choice between different work items, but the workflow management system prioritized pending activities according to different economic criteria and allowed brokers only to request the highest ranked work item [34]. Using a pull mechanism, a resource requests the next work item at a convenient time, but not necessarily upon availability. A combination of push and pull is possible, if a resource as assigned a number of activities, but the sequence of execution of the individual activities is left to the performer.

Finally, *participant autonomy* determines, whether the workflow enactment service determines the final recipient of a work item, or whether a workflow participant may refuse the assignment of a particular work item and send it back to the shared worklist or the workflow enactment service for reallocation.

4 Resource Integration

Integration is regarded as one of the primary goals during information system design. Literally, integration means to form, coordinate, or blend something into a functioning or unified whole by ending existing segregation [31]. Two distinct types of integration can be distinguished [37]:

Integration through connection occurs if a new system is created through the creation of links between disparate, but logically connected entities or sub-systems. Typically this is an ex-post integration of existing systems, like the integration of enterprise applications through a workflow management system. *Integration through combination* occurs if similar system elements are combined, thus leading to a decreased number of elements and relationships within the system (in the sense of abstraction). Typically this form of integration happens during the conceptual design phase of an information system, for example the development of a complex application with an integrated workflow layer for the transport of application data.

ROSEMANN names reduction of redundancy, increased system consistency and integrity, and better decision support through timely information supply as the main goals of integration efforts [36]. The design of a workflow application creates integration requirements, which can be differentiated into internal and external integration requirements. Internal integration requirements concern those systems a workflow application needs to connect to in order to ensure the functionality of the core workflow system. External integration requirements exist with regard to systems that either invoke the workflow system from outside (embedded usage) or systems that are invoked by the workflow application.

4.1 Internal Integration Requirements

A workflow application coordinates activities, resources, data, and applications. Consequently, all these elements need to be integrated to ensure the functionality of the workflow systems.

Resource integration is required by the workflow system to keep track of the participants available for activity assignment. Since many companies maintain resource information in organizational directories or similar applications, a fully integrated workflow application would use this information rather than replicate resource data internally. For this reason, the organizational elements available during the specification of the workflow need to be synchronized with the organizational entities found in the actual enterprise.

Data integration is required to make workflow relevant data accessible to the workflow system. This can be achieved by connecting the system to databases used by external application systems. If the workflow system acts as an enterprise application integration hub, conversion of data types and field values may be necessary. From a resource management perspective, data integration is required if field values of business objects are to be used during the selection of qualified workflow performers.

Application integration describes the ability of the workflow system to invoke external application systems during the enactment of a process. For organizational processes, applications are often called in their entirety (e. g., a word processing application), while for software processes the granularity of application invocation is at the method or function level.

In addition to these three integration requirements, the use of existing security infrastructures is another important feature of workflow applications. *Security integration* relates to the use of existing authentication and authorization mechanisms through the workflow system, such as single-sign-on, role-based access control mechanisms and public key infrastructures.

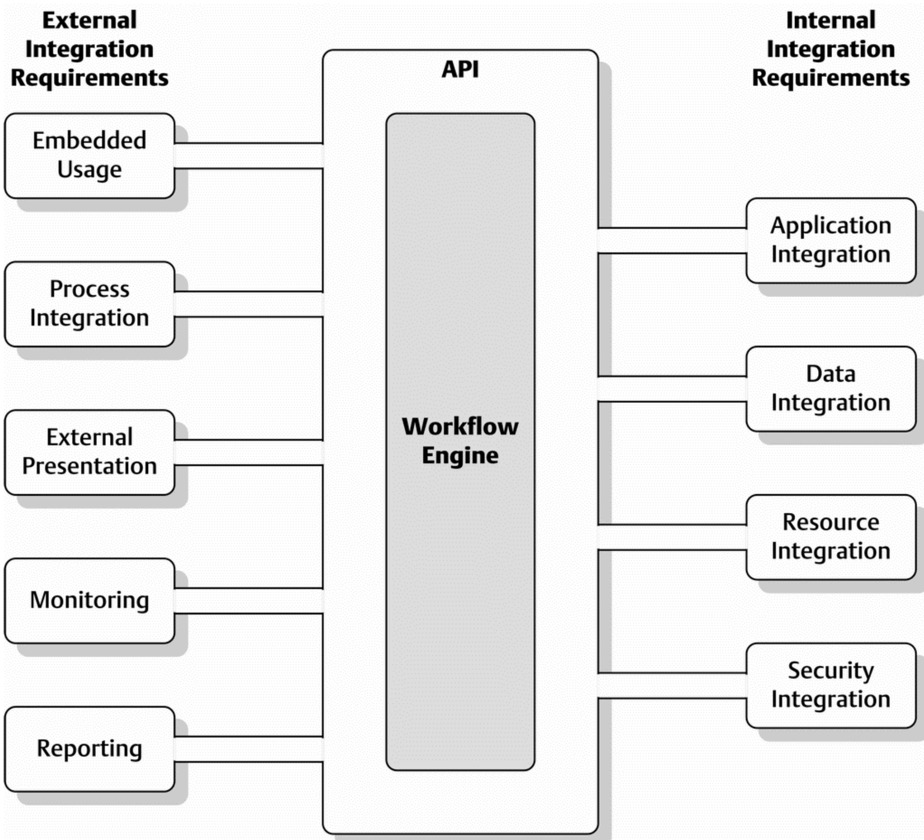
4.2 External Integration Requirements

The external integration of a workflow system relates to the fact, that a workflow system is an application system in itself. External applications may require calling the services of a workflow engine from outside, invoking workflow instances, querying the status of activity instances, or handling resource assignments through external scheduling mechanisms. Additionally, the workflow system may be required to present work to outside parties, which are not direct users of the workflow application.

External invocation of the workflow engine is used during B2B process integration, among other scenarios. The workflow enactment service can exhibit its capabilities as services to outside parties, allowing them to invoke a process and pass initial data to the process instance. Examples for an external invocation are e-mail (mail daemon triggers the workflow), the web (a web server triggers the workflow) or other applications, which embed the workflow system (a function within an application results in the start of a workflow).

Presentation of information to outside parties is necessary, if the workflow system has to notify external participants about the status of "their" workflow instances or if system load information is passed on to external system management tools. Also, the use of audit trail information by external applications (e.g. reporting or business activity monitoring tools) falls into this category.

Figure 6 summarizes the integration requirements from an internal and an external perspective.



Source: Modified from [7].

Figure 6: Integration Requirements

Besides the actual execution of role resolution and allocation, the workflow application has to handle the notification of performers and process managers, as well as the management of unexpected situations (exception handling, compare [11]).

5 Organizational Maintenance at Run Time

While the sequence of activities within a workflow definition is relatively stable, the resource model needs to be constantly monitored at runtime to reflect the changes in the organizational population of the enterprise. These changes can be either macro changes at the entity level, or micro changes at the attribute level.

Changes at the *macro level* include the arrival and departure of members of the organization, changes of the association between resources and organizational units, and the creation of new or the disbanding of existing groups (e.g. project teams). If the resource information is stored within the workflow application itself, user accounts and their associated privileges need to be updated for arriving or departing process participants. In fact, the process of hiring a new staff member can be represented as a workflow itself, as implemented in the workflow management system CARNOT.

Changes at the *micro level* relate to changes in the authorization profile of resources , e.g. the granting of additional authorizations, or the revocation of temporary privileges, the establishment of temporary substitute relationships, the acquisition of new capabilities by resources (e.g. through the completion of training classes), and the change within a resource's experience levels. If the role resolution process takes individual attribute values of resources into account, the integrity of the attributes provided by the resource model and used by the workflow model(s) has to be maintained.

The change of a user's experience level can be very useful to track, in order to allow an activity allocation at a fine level of granularity. As a simple measure, a workflow enactment service could keep track of the number of activity executions by a particular performer, and apply the simple formula more executions = more experience. By the time a new activity instance of the same type is to be assigned, the workflow enactment service could rank the candidate performers according to their experience and assign the work items in that order, taking additional constraints (such as workload) into account. Another possibility would be to measure the activity completion time of individual resources, ranking these resources by their efficiency. Note that if all activities are allocated according to this schema without taking further constraints into consideration, the resulting workload distribution will follow a power-law curve, with few performers sharing the majority of the workload, while the majority of the performers receive significantly fewer work items [6]. Therefore, it is useful to balance an experience-based work distribution by taking additional parameters into account, such as the average length of the worklists and/or the current workload of individual candidates.

In addition to activity assignment purposes, experience data is also useful during the handling of exceptions, such as overdue activities or processes. Instead of a hard-coded exception handling scheme, the workflow enactment service could assign an escalating activity to a more experienced person, if experience information is maintained in the system.

6 Related Work

Resource models have been the subject of scientific study for some years, but only a few concrete examples have been proposed. The OMM Organization and Role Model described by CHENG aims at a separation of organization and roles in the context of of electronic commerce applications [12]. It consists of the entity types enterprise, organization, member and virtual link. While an enterprise is a collection of a number of organizations, each organization consists of a number of members that share common attributes. Member objects are the elementary resources that map to the actual resources of the enterprise. Virtual links provide relationships between members of the same or different organizations.

A generic organizational meta model has been proposed by BUSSLER, and was implemented in the research prototype MOBILE [8-10]. The meta model is on a very high level of abstraction and provides agent types that can be addressed directly by workflow activities, non-agent types that have to be resolved before a workflow activity can be assigned and attributes, operations and consistency rules on a type and an instance level. In order to depict a real-world organization the meta model has to be instantiated, i.e. a domain-specific model has to be derived from the abstract entity types. For example, in order to depict an organization that consists of users, groups, departments and project teams the abstract entity type agent would be instantiated as the entity type user, the entity types group, department and project team would be instantiated non-agent types. The maximum number of members in a group would be an instance-level consistency rule and the process of assigning a member to an organization would be implemented as a type-level operation.

The Organization and Resource Model (ORM) presented by RUPIETTA is an independent repository for organizational structures that has been used by the (discontinued) workflow management system SNI WorkParty [39]. It provides a semantically rich meta model that can be modified through the inheritance of the existing entity types. The entity types of the ORM, such as organizational positions, position types, tasks, and units reflect the German organizational theory. However, some of the entity types provided cannot be used for the assignment process, such as technical resources or the ownership relation between users and technical resources.

The resource meta model proposed by VAN DER AALST et al. consists of a UML class diagram and a corresponding XML rendition for the specification of workflow resources [1]. This model was developed in the context of the XRL/Woflan project, which aims at the development of an exchangeable workflow modeling language. The goal of XRL is the exchange of workflow specifications between different workflow systems. For this purpose, the resources required by a particular workflow specification can be expressed in XML.

An independent resource management facility was presented by HUANG et al. [25], and has been implemented in the workflow management system HP Chang*e*ngine. Their approach consists of an independent resource manager that integrates existing local and site resource management systems under a common schema. The proposed system implements an SQL-like language for the querying and assignment of resources. The underlying schema for representation of resources differentiates between resources that can belong to units and the managing entities of these units.

MOMOTKO and SUBIETA have stressed the importance of dynamic change within the workflow participant assignment and have presented a modification to the Workflow Management Coalition's WPDL [47] that allows the specification of formal expressions used during the taks allocation phase of workflow execution [32].

SHEN et al. went one step further by discussing and benchmarking different criteria upon which the activity assignment process should be based [41]. In addition to the static relationship between the capabilities required by an activity and the capabilities exposed by a resource, they propose the use of activity relationships between pending activities and existing worklist entries, as well as social relationships between process performers. The authors present a fuzzy-logic based approach to measuring the three criteria, which allows a more precise formulation of properties required by and activity and properties actually exposed by a resource. SHEN et al. propose the pair wise measurement of social relationships between individual process participants, but do not mention, how these measurements should be obtained [41].

ZHAO has classified the knowledge contained within a workflow solution [50]: Process knowledge relates to the structure of processes and activities, institutional knowledge relates to the roles, business procedures and regulations of the enterprise, and environmental knowledge such as governmental regulations. From the perspective of resource modeling, the maintenance and improvement of institutional knowledge is of significant importance. KLAMMA and SCHLAPHOFF have presented a prototype that maintains and explicates organizational memory information in a workflow context [28].

7 Summary and Outlook

In this article we have outlined the major aspects of resource management within workflow applications, following the workflow life cycle. After a discussion of different workflow stakeholders we presented a generic resource meta model for the initial specification of the resource structure and its population, which combines workflow-oriented with organization-oriented modeling concepts. Subsequently, we outlined variations in the assignment and synchronization policies for the run time allocation of tasks to performers, and presented strategies for the actual activity assignment at run time. In section 5 we discussed the maintenance of resource information during the operation of a workflow application at the macro and micro level. Finally, we presented strategies for the use of workflow audit trail data to improve resource utilization and develop new process strategies.

While the scientific community has shown some interest in the definition of resource structures, and issues related with the actual resource assignment, the later stages of the workflow life cycle have received significantly less attention. This includes the implementation of knowledge-based resource allocation algorithms, as well as the continuous improvement of resource structures and the respective allocation strategies. Only if these areas are successfully addressed by both researchers and practitioners, the true potential of workflow-based applications will be realized, and the alignment of process technology and organizational capabilities will be successful in the long run.

The use of workflow technology for human-centric processes presents both a technical and organizational challenge. Consequently, it has to be addressed from both the community of workflow vendors and the community of workflow users. Vendors can improve their products by providing comprehensive integration mechanism with existing resource repositories, maintenance interfaces for resource information, and support for different assignment and synchronization strategies. Workflow users can improve their applications by aligning the resource models, work allocation strategies, and maintenance procedures with the organizational structures and policies that best suit their company. An evaluation and validation of the different modeling and management approaches presented in this paper is the next step. Experimental research is needed to determine the effects of workflow applications on organizational structures and their management, and to establish the success factors for workflow-enabled organizations.

References.

1. van der Aalst, W.M.P., Kumar, A. and Verbeek, H.M.W., Organizational Modeling in UML and XML in the context of Workflow Systems. in *Symposium on Applied Computing (SAC) 2003*, (Melbourne, Florida, USA, 2003), ACM.
2. Ahn, G.-J., Sandhu, R.S., Kang, M. and Park, J., Injecting RBAC to secure a Web-based workflow system. in *Fifth ACM Workshop on Role-Based Access Control July 26-27, 2000*, (Berlin, Germany, 2000), ACM, 1-10.
3. Alt, R., Klein, S. and Kuhn, C., Service Task Allocation as an Internal Market. in *Second European Conference on Information Systems (ECIS 1994)*, (Breukelen, Netherlands, 1994), Nijenrode University Press, 424-432.
4. Arkin, A., Askary, S., Fordin, S., Jekeli, W., Kawaguchi, K., Orchard, D., Pogliani, S., Riemer, K., Struble, S., Takacsi-Nagy, P., Trickovic, I. and Zimek, S. Web Services Choreography Interface (WSCI) 1.0, W3C, 2002, 121.
5. Banerji, A., Bartolini, C., Beringer, D., Chopella, V., Govindarajan, K., Karp, A., Kuno, H., Lemon, M., Pogossiants, G., Sharma, S. and Williams, S. Web Services Conversation Language (WSCL) 1.0, World Wide Web Consortium (W3C), Palo Alto, CA, USA, 2002, 22.
6. Barabasi, A.-L. *Linked : the new science of networks*. Perseus Pub., Cambridge, MA, 2002.
7. Becker, J., zur Muehlen, M. and Gille, M. Workflow Application Architectures: Classification and Characteristics of workflow-based Information Systems. in Fischer, L. ed. *Workflow Handbook 2002*, Future Strategies, Lighthouse Point, FL, 2002, 39-50.
8. Bussler, C., Analysis of the Organization Modeling Capability of Workflow-Management-Systems. in *Conference of the Pacific Research Institute for Information Systems and Management (PRIISM '96)*, (Maui, HI, USA, 1996).
9. Bussler, C. *Organisationsverwaltung in Workflow-Management-Systemen*. Deutscher Universitäts-Verlag, Wiesbaden, Germany, 1998.
10. Bussler, C. Policy Resolution in Workflow Management Systems. *Digital Technical Journal*, *6* (4). 26-49.
11. Casati, F. Models, Semantics, and Formal Methods for the design of Workflows and their Exceptions, University of Milan, Milan, 1998, 180.
12. Cheng, E.C., An Object-Oriented Organizational Model to Support Dynamic Role-based Access Control in Electronic Commerce Applications. in *32nd Annual Hawai'i International Conference on System Sciences (HICSS '99)*, (Wailea, HI, USA, 1999), IEEE.
13. Cummins, F.A. Workflow Resource Management Facility Request for Proposal (Draft). Object Management Group, Framingham, MA, USA, 1999.
14. Curbera, F., Goland, Y., Klein, J., Leymann, F., Roller, D., Thatte, S. and Weerawarana, S. Business Process Execution Language for Web Services, Version 1.0, BEA, IBM, Microsoft, 2002.
15. Curtis, B., Kellner, M.I. and Over, J. Process Modeling. *Communications of the ACM*, *35* (9). 75-90.
16. Domingos, H.J., Martins, J.L., Preguiça, N.M. and Duarte, S.M., A Workflow Architecture to Manage Mobile Collaborative Work. in *Encontro Português de Computação Móvel 1999 (EPCM 99)*, (Tomar, Portugal, 1999).

17. Dourish, P., Holmes, J., MacLean, A., Marqvardsen, P. and Zbyslaw, A., Freeflow: Mediating Between Representation and Action in Workflow Systems. in *Proceedings of the 1996 Conference on Computer Supported Cooperative Work (CSCW '96)*, (Boston (MA), 1996), ACM, 190-198.
18. Du, W., Davis, J., Huang, Y.-N. and Shan, M.-C. Enterprise Workflow Resource Management, Hewlett Packard Laboratories, Palo Alto, CA, USA, 1999.
19. Ellis, C.A. and Nutt, G.J. Office Information Systems and Computer Science. *ACM Computing Surveys*, *12* (1). 27-60.
20. Galbraith, J.R. *Designing complex organizations*. Addison-Wesley Pub. Co., Reading, Mass.,, 1973.
21. Galler, J. and Scheer, A.-W. Workflow-Projekte: Vom Geschäftsprozeßmodell zur unternehmensspezifischen Workflow-Anwendung. *Information Management*, *10* (1). 20-27.
22. Harker, P.T. and Ungar, L.H., A Market-Based Approach to Workflow Automation. in *NSF Workshop on Workflow and Process Automation in Information Systems*, (Athens, GA, USA, 1996).
23. Heilmann, H. Die Integration der Aufbauorganisation in Workflow-Management-Systeme. in Heilmann, H., Heinrich, L.J. and Roithmayr, F. eds. *Information Engineering*, Oldenbourg, München, 1996, 147-165.
24. Hoffmann, M., Löffeler, T. and Schmidt, Y. Flexible Arbeitsverteilung mit Workflow-Management-Systemen. in Herrmann, T., Scheer, A.-W. and Weber, H. eds. *Verbesserung von Geschäftsprozessen mit flexiblen Workflow-Management-Systemen*, Physica, Heidelberg, Germany, 1999, 135-159.
25. Huang, Y.-N. and Shan, M.-C., Policy-Based Resource Management. in *11th International Conference on Advanced Information Systems Engineering (CAiSE '99)*, (Heidelberg, Germany, 1999), Springer, 422-428.
26. IBM. Flowmark - Modeling Workflow. Release 2.1, IBM Deutschland Entwicklung GmbH, Vienna, Austria, 1995.
27. Jablonski, S. and Bussler, C. *Workflow Management. Modeling Concepts, Architecture and Implementation*. International Thomson Computer Press, London et al., 1996.
28. Klamma, R. and Schlaphoff, S., Rapid Knowledge Deployment in an Organizational-Memory-Based Workflow Environment. in *8th European Conference on Information Systems (ECIS 2000)*, (Vienna, Austria, 2000), 364-371.
29. Kwan, M.M. and Balasubramanian, P.R., Adding Workflow Analysis Techniques to the IS Development Toolkit. in *31st Annual Hawai'i International Conference on System Sciences*, (Kohala Coast, HI, 1998), IEEE, 312-321.
30. Leymann, F. and Altenhuber, W. Managing business processes as an information resource. *IBM Systems Journal*, *33* (2). 326-348.
31. Merriam-Webster. Integration, 2002.
32. Momotko, M. and Subieta, K., Dynamic Changes in Workflow Participant Assignment. in *Sixth East-European Conference on Advances in Databases and Information Systems, ADBIS'2002*, (Bratislava, Slovakia, 2002).
33. Moore, C. Common Mistakes in Workflow Implementations, Giga Information Group, Cambridge, MA, 2002, 2.
34. zur Muehlen, M., Resource Modeling in Workflow Applications. in *1999 Workflow Management Conference*, (Münster, 1999), University of Münster, 137-153.
35. Orlowska, M.E., Current Issues in Workflow Management. in *2003 SAP Innovation Congress*, (Miami, FL, USA, 2003), SAP AG.
36. Rosemann, M. Gegenstand und Aufgaben des Integrationsmanagements. in Scheer, A.-W., Rosemann, M. and Schuette, R. eds. *Integrationsmanagement*, University of Muenster, Muenster, Germany, 1999, 5-18.

37. Rosemann, M. *Komplexitätsmanagement in Prozessmodellen. Methodenspezifische Gestaltungsempfehlungen für die Informationsmodellierung*. Gabler, Wiesbaden, 1996.
38. Rosemann, M. and zur Muehlen, M. Evaluation of Workflow Management Systems - a Meta Model Approach. *Australian Journal of Information Systems*, *6* (1). 103-116.
39. Rupietta, W. Organization and Role Models for Workflow Processes. in Lawrence, P. ed. *Workflow Handbook 1997*, John Wiley & Sons, Chichester, UK, 1997, 165-172.
40. Sellen, A. and Harper, R. Can Workflow Tools support Knowledge Work? A Case Study of the International Monetary Fund, Rank Xerox Research Center, Cambridge, UK, 1996, 21.
41. Shen, M., Tzen, G.-H. and Lio, D.-R., Multi-Criteria Task Assignment in Workflow Management Systems. in *36th Hawai'i International Conference on System Sciences (HICSS 2003)*, (Waikoloa, HI, USA, 2003), IEEE.
42. Sheth, A. From Contemporary Workflow Process Automation to Adaptive and Dynamic Work Activity Coordination and Collaboration. *SIGGROUP Bulletin*, *18* (3). 17-20.
43. Tan, J.C. and Harker, P.T. Designing Workflow Coordination: Centralized Versus Market-Based Mechanisms, University of Pennsylvania, Department of Systems Engineering, Philadelphia, Pennsylvania, USA, 1997.
44. Wainer, J., Kumar, A. and Barthelmess, P. Security management in the presence of delegation and revocation in workflow systems, Universidade Estadual de Campinas, Campinas, Brazil, 2001.
45. Wellman, M.P., Walsh, W.E., Wurman, P.R. and MacKie-Mason, J.K. Auction protocols for decentralized scheduling. *Games and Economic Behavior*, *35* (2). 271-303.
46. Weske, M., Goesmann, T., Holten, R. and Striemer, R., A Reference Model for Workflow Application Development Processes. in *International Joint Conference on Work Activities Coordination and Collaboration (WACC '99)*, (San Francisco, CA, 1999), ACM, 1-10.
47. WfMC. Interface 1: Process Definition Interchange Process Model, Workflow Management Coalition, Winchester, UK, 1999.
48. WfMC. Terminology and Glossary, 3rd Edition, Workflow Management Coalition, Winchester (ID), 1999.
49. Workflow Management Coalition *Workflow Management Coalition Terminology & Glossary. Document No. WFMC-TC-1011*. WfMC, Winchester, UK, 1999.
50. Zhao, J.L., Knowledge Management and Organizational Learning in Workflow Systems. in *1998 Americas Conference on Information Systems (AIS '98)*, (Baltimore, MD, USA, 1998), AIS.
51. Zisman, M. Representation, Specification, and Automation of Office Procedures *Wharton Business School*, University of Pennsylvania, Philadelphia, 1977.