Volume versus Variance: Implications of Data-intensive Workflows

Michael zur Muehlen
Stevens Institute of Technology
Hoboken, NJ 07030
mzurmuehlen@stevens.edu

Abstract

A data-intensive workflow is a process that is faced with a large volume or highly variable forms of information. The increasing digitization of office processes, the use of workflows to integrate publicly available data sources, and the application of workflow technology to scientific problem solving have led to an increased interest in the design and deployment of data-intensive workflows. This paper discusses the notion of data-intensive workflows and outlines the implications of increasing data volumes and variances for the design of process-aware applications.

1 Introduction

Process-oriented Information Systems have been developed for more than 30 years [6]. Their development is based on a behavioral view of the enterprise as a system. This view defines an organization as an information processing entity that transforms inputs into outputs according to a set of procedural rules. These procedural rules can be observed, (re-)defined, and managed. This process perspective on the organization is not a new concept. In management science its roots can be traced back to the early 1930s in Europe [8] and the late 1950s in the United States [7]. The restructuring of organizations along their core processes has demonstrated benefits in particular among functionally fragmented organizations that were striving to offset the side-effects of worker specialization and functionally-oriented departments. The efficiency benefits of process-driven application design have made workflow systems a readily available application in many organizations, to the extent that many middleware systems and packaged applications contain workflow technology.

In contrast to this focus on organizational behavior, the development of functional Information Systems has traditionally been dominated by data management concerns. Beginning with accounting and record-keeping systems, the need to make large data sets accessible and manageable has led to significant innovation in areas such as database technologies, query languages, and lately, the semantic markup of information using technologies such as *RDF* and *OWL*. The increasing maturity of data access standards such as *RSS* and *SOAP*, combined with authentication technologies for distributed environments is making significant data sets easily accessible. In the United States new data sharing initiatives such as *data.gov*, *usaspending.gov* and *recovery.gov* make government information publicly available using standardized access mechanisms and data formats.

Bulletin of the IEEE Computer Society Technical Committee on Data Engineering

The physical distribution of audio and video materials on optical or magnetic media is continually diminishing in favor of digital downloads. The increasing availability of digital information poses questions for the design of workflow applications, which are traditionally based on the notion of a well-defined and limited set of information that has to be routed between process participants, be they people, applications, or services. What does it mean for a workflow to be *data-intensive*?

2 Classes of Workflow Data

To answer this question we need to consider the types of data that surround a typical workflow application. An established classification for data handled in the context of workflow applications has been defined by the Workflow Management Coalition Glossary [1]:

Content Data (sometimes referred to as *application data*) relates to the (user-defined) payload of a workflow instance. This data is either supplied to the workflow management system by the initiator of the workflow when the workflow instance is created (i.e., the initial /payload), or it is created by individual activities throughout the life of the workflow instance. Content data in its most general form has no bearing on the execution path of a workflow instance. A typical example would be the line item description of an order or the content of a photograph submitted as evidence in an insurance claim.

Workflow Data refers to those data objects that are produced by the workflow execution environment itself during the enactment of the workflow instance. This class of data relates to technical information, such as audit trail information that documents the instantiation, invocation and completion of activity instances [2], user log-on and log-off information, or recovery data that a workflow server might generate in order to be able to recover after a failure situation. While this information is generally not considered for decision-making at the instance level, it can be used for applications such as server health checks, load balancing, and - combined with content data - process analytics.

Workflow-relevant Data relates to those data objects that can affect the routing logic of a workflow application, both in terms of control flow decisions (such as which outgoing sequence flow of a data-based XOR gateway to activate), as well as in terms of task assignment (i.e., which performer a particular work item should be offered to). If the decision logic of a workflow application is based on few stable attributes it is often encoded in the process model itself. If the decision logic requires the evaluation of multiple attributes, rules, or changes frequently it is increasingly located in a separate rules management system. Workflow-relevant data may be part of the externally generated payload (such as the status of a customer) or it can be generated by the workflow application during the execution of the workflow instance itself. A typical example is information about the starting user of the workflow instance. This data is not known until the workflow instance has been created, but in many cases it is being used to assign activities to the initiator of the workflow instance.

In many cases a workflow application plays the role of a mediation system that enables disparate systems or services to interact. If in the process of mediation the data generated by the source system or service is transformed so that it can be read by the destination system or service the management of provenance information plays an important role, and the traceability of transformations may become a requirement. In this sense, the workflow application may become an author of data that would otherwise be classified as application data. The boundary between data that is exposed to the workflow application for routing decisions, and pure application data is increasingly blurry, so that the main distinction in this taxonomy is between data that is generated by the workflow application and data that is consumed by the workflow application.

3 Data Volume versus Content Variance

The data surrounding different workflow applications varies both in terms of *volume* and *variance*. A data-intensive workflow application can be defined as a process-oriented information system that is designed to process data in large volumes and/or data with highly variable characteristics.

Item	Data Volume		Content Variance	
	Low Volume	**High Volume**	**Low Variance**	**High Variance**
Workflow Data	The workflow application generates a small amount of audit data (low fidelity)	The workflow application generates a large amount of audit data (high fidelity)	The structure of the workflow audit trail is similar from one workflow instance to the next	The structure of the workflow audit trail can vary widely between instances
Workflow-relevant Data	The control flow of a workflow instance is determined based on a limited set of data	The control flow of a workflow instance is determined based on a large set of data	The control flow of a workflow instance is determined based on predictable datatypes	The control flow of a workflow instance is determined based on varying data types
Content Data	Each workflow instance processes a small amount of data	Each workflow instance processes a large amount of data	The data types are stable from one workflow instance to the next	The data types can vary widely between workflow instances
Performer Data	Few performers participate in the execution of a workflow instance	Many performers participate in the execution of a workflow instance	The set of performers is stable between workflow instances	The set of performers can vary widely between workflow instances
Context Data	The execution of the workflow is relatively independent of context information	The execution of the workflow is highly dependent on context information	Workflow instances are executed under similar circumstances	Workflow instances are executed under highly varied circumstances

Figure 1: Data Volume versus Content Variance

Large data volumes can relate to the type of data that as well as to the number of data objects that are routed by the workflow engine to different processing stations. Examples for large data types are applications that process large images or movie files, such as digital scans from medical devices, satellite imagery, or applications that post-process video streams. Even though each workflow instance may only transport a limited number of these objects, the size of each object can be in the *MB* to *GB* range. If the workflow application moves these objects across a network the requirements for network throughput increase with the number of concurrent workflow instances. If no mediation is required, the workflow application may refer to these objects using URIs without moving them physically. However, if the workflow application has to mediate data formats (e.g. encoding of materials for different end user devices) it may be necessary to physically transport large data volumes. Examples for a large number of data objects are high-volume workflow applications such as trading systems, traffic monitoring applications, or telephony applications. Even though the size of each data object is very limited, the number and frequency of these objects, combined with requirements for low latency information flow puts an emphasis on the data processing capacities of a workflow applications.

Content variance relates to the rate with which the structure of content data changes. A workflow application can be regarded as data-intensive if it has to operate in an environment where the payload varies highly between workflow instances. A typical example are intelligence applications where a large variety of information sources are routed to analysts based on content correlation and the context in which they were gathered. An analyst may be presented with textual, visual, and auditory information, and the composition of data in each workflow instance may differ widely. Workflow applications with a high degree of content variance tend to favor the use of case-management techniques, where an individual actor is provided with the total set of information related to the workflow instance, but is given some leeway to decide the appropriate course of action.

4 Participant Volume versus Participant Variance

A different aspect of data-intensive workflow applications is the number and variation of workflow participants. Some workflow applications are enacted in stable environments with a defined number of humans, systems, or services that participate in the execution of each workflow instance. Other workflow applications may allow an unforeseen number of participants to interact with it (workflow for the crowd). And yet other workflow applications may interact with a defined number of participants, but the capabilities of these participants may vary with each workflow instance.

An example of high participation volume is a process where a complex problem is broken into smaller units which are assigned to individual agents to solve. Amazon.com's mechanical turk service is an example of such a *crowdsourcing* application. In this example a large task (such as the analysis of a large number of images) is broken into small, identical subtasks that are assigned to individual actors. The number of actors in the system is not know ahead of time and can be influenced through the use of bidding mechanisms and the creation of task-dependent incentives. If a workflow is performed by a large number of casual users the design of user interfaces has to consider in particular how an untrained user can learn the task at hand, whereas a workflow task that is regularly performed by a select group of specialists can be tailored to the specific abilities of the specialist, with less regard to common accessibility.

An example of high participation variability is the military process of Close Air Support. This process describes how ground troops may request the assistance of airborne assets in the fulfillment of their mission [3]. An instance of this process may involve fixed wing or rotary wing aircraft, which may have different operating capabilities and communication devices. In addition, these assets may be prepared to assist (pre-planned scenario) or may be diverted from another mission (ad-hoc scenario). Despite these differences, the overall structure of the Joint Close Air Support process remains the same, but its execution needs to be tailored to the specific communication capabilities and requirements of the participating actors. The military has solved this issue by standardizing the content of the messages exchanged between process participants, rather than standardizing the medium through which these messages are communicated.

5 Context Information

Workflows may be instantiated in different environments. These environments may affect the reliability of the services or actors that the workflow enactment service depends upon. Taking the aforementioned Joint Close Air Support example, this process can be invoked in a daytime environment with clear visibility, reliable communication links between participants, and a technical infrastructure that allows the exchange and confirmation of broadband information such as video streams recorded by aircrafts. In another setting the process can be invoked at night, in a mountainous terrain, where image processing equipment is unavailable, communication bandwidth is limited, and the accuracy of information is much less certain.

If a workflow is executed in the same or similar context its design and development can be performed in a closed environment, and it can be optimized to perform under these anticipated circumstances. This is typically the case when the enactment environment is entirely under the control of the organization that performs the workflow, as are back-office processes and certain transactional processes where the provider can dictate data formats and interaction patterns to the requester, e.g. insurance claim scenarios.

In cases where a workflow is executed under different circumstances, and where these circumstances have a direct impact on the routing, decision logic, or performance of individual tasks, the workflow designer has fewer options to optimize the performance of the process a priori. In these cases the workflow design needs to provide event handling capabilities to react to changes in the environment and mechanisms that allow for the flexible routing, performance, and assignment of tasks (see e.g., [9]).

6 The Role of Semantics

The formal representation of the data context of workflow applications in a semantic format such as *RDF-S* or *OWL* can be beneficial in case of high content and/or context variability. The semantic annotation of content and context information allows for the following:

If a process modeling grammar such as BPMN is encoded in a semantic markup format, then a process model can be automatically compared to the grammar in order to identify modeling mistakes. Since the presence of modeling mistakes has been documented even in commercial reference models [5], such an evaluation would assist workflow developers in minimizing the risk of failed workflow instantiations and executions.

If a process model is encoded in semantic markup format, then a process instance can be evaluated for compliance against the process model. This might be useful if the process instance is not derived directly from the model (as is the case in many production-type workflow systems), but is rather a dynamically evolving execution path that is constrained by a declarative process modeling formalism, such as GPSG [4].

If the payload of a process is described in a semantic markup format, then the process designer may be able to specify the process logic by referencing the semantic classes of information that the workflow is designed to process, rather than the actual data format that needs to be ingested and transformed. This would allow for a separation of the processing concerns (what the workflow is designed to achieve) from the execution concerns (how the transformation has to take place).

If the audit trail information of the process is described in a semantic markup format, the designers and users of process analytics may be able to evaluate workflow instances that did not process the same data formats, yet were enacted on information with similar semantics.

The use of semantic markups and ontologies for the design of process-aware information systems has seen an increased interest recently, as demonstrated e.g. by the EU-funded IP-SUPER project (*www.ip-super.org*) and remains a promising area of research to allow for workflow applications that can perform well in heterogeneous data environments.

Item	Data Volume		Content Variance	
	Low Volume	**High Volume**	**Low Variance**	**High Variance**
Workflow Data	Little processing and storage requirements for analytics information, however: limited insight	Increasing processing and storage requirements for analytics information, however: rich insight	Allows for the design of stable analytics views and reporting components	Requires adaptive transformation logic to feed analytics information, views must be configurable
Workflow-relevant Data	Control-flow rules may be specified as part of the process model	Control-flow rules should be handled by separate rules logic	Process debugging and automated decision making are possible	Manual decision making may be required if data types cannot be anticipated
Content Data	Lightweight, fast workflow applications	Increasing demands for storage and network bandwidth	Predictable data formats can be used to optimize data flow	Variable data formats may lead to case-management-based workflow solutions
Performer Data	Organization structures can be designed based on process logic	Workflow organization model may have to reflect real-world organization	User interface screens can be tailored to the specific abilities of performers	User interface screens have to be easy to learn by new performers
Context Data	Testing, simulation and deployment of the workflow application can be performed in a closed environment	Event-processing capabilities are required to react to context data changes	Workflow design can be optimized to a particular execution scenario	Workflow design and execution capabilities need to be flexible to accommodate context changes

Figure 2: Implications of Data Volume and Content Variance

7 Implications for Workflow Application Designers

Data volume and content variance can have a significant impact on the design of workflow applications. Workflow designers should be aware how big and how stable the different classes of data are that their application interacts with. While the volume of data typically affects network throughput and storage requirements, the variability of content information has a more pronounced impact on design decisions. We have provided a classification schema for the different classes of data typically encountered in the context of workflow applications, and discussed the implications of changes in data volume and content variance. Data-intensive workflows can be encountered in many different disciplines, but their management may be simplified by a common set of design principles based on the characteristics of data that makes the workflow data-intensive.

References

[1] D. Hollingsworth: The Workflow Reference Model. Document Number TC001003, Workflow Management Coalition, Winchester, UK, 1995.

[2] Workflow Management Coalition: Business Process Analytics Format - Draft Specification. Document Number TC-1015, Version 1.0, 2009, Cohasset, MA.

[3] U.S. Department of Dense: Joint Tactics, Techniques, and Procedures for Close Air Support (CAS). Joint Publication 3-09.3, 2 September 2005. Washington, DC, 2005.

[4] N. S. Glance, D. G. Pagani, and R. Pareschi, Generalized Process Structure Grammars (GPSG) for flexible representations of work, Boston (MA), 1996, ACM, pp. 180-189.

[5] J. Mendling, Metrics for Process Models: Empirical Foundations of Verification, Error Prediction, and Guidelines for Correctness, Springer, Berlin et al. 2008.

[6] M. Zisman, Representation, Specification, and Automation of Office Procedures, University of Pennsylvania, Philadelphia (PA), 1977.

[7] E. D. Chapple, and L. R. Sayles, The Measure of Management. Designing Organizations for Human Effectiveness, Macmillan, New York (NY), 1961.

[8] F. Nordsieck, Grundlagen der Organisationslehre, C. E. Poeschel Verlag, Stuttgart, 1934.

[9] P. Dadam, M. Reichert et al., Towards truly flexible and adaptive process-aware information systems, Proc. UNISCON, 2008, pp. 72-83.