Analysing Dependencies between Documentary Fragments for the Management of Mediated Communities of Action

Nicolas Prat*, Manuel Zacklad**

*Département Systèmes d'Information et de Décision
ESSEC Business School
Avenue Bernard Hirsch
BP 50105 – 95021 Cergy cédex – France
Prat@essec.fr

**Equipe Tech-CICO – ISTIT FRE CNRS 2732
Université de Technologie de Troyes
12, rue Marie Curie
BP 2060 – 10010 Troyes cédex – France
Manuel.Zacklad@utt.fr

Abstract: This paper focuses on situations where documents serve to coordinate a distributed Community of Action engaged in a common goal-directed activity. In such contexts, a document appears as a set of fragments contributed by various authors. Furthermore, it is possible to analyse the traces of collective activity left in documents. This analysis may serve several purposes. Among other things, it may be used to understand the evolution of the cognitive and social dynamics of the community. After justifying the choice of the DofA (Documents for Action) conceptual framework, we elaborate a UML-based model of DofA. We use this model as a structure for conceptually representing traces of mediated collective activities. We pursue with the issue of dependency analysis and measurement, which leads us to propose specific dependencies in the context of DofA. We illustrate the early steps of an application to a scenario of collective writing. Finally, we discuss further research directions suggested by our approach, which analyses and measures dependencies from traces of mediated collective activity.

1 Introduction

The development of the Web (Internet, Intranet) has resulted in a multiplicity of document-related collective practices: projects mediated by plans and contracts, health-care networks based on patients' records, collaborative design of open source software, only to name a few examples. A whole set of new document-related practices are therefore emerging. Among other examples, let us mention (1) the collective writing of a document exchanged by e-mail and supported by annotations, (2) the contribution to newsgroups and (3) the use of Wikis or blogs.

In this paper, we focus particularly on situations where documents are used to mediate the coordination of a widely distributed group committed to working towards a common goal, corresponding to the concept of Communities of Action [Za03][1]. In contexts of this kind, the document in question can be viewed as a set of fragments contributed by various authors, the final content of which remains largely indeterminate, while its fast dissemination makes it a useful tool for conveying information, assisting decision-making and probing situations. The animation of these communities, whose members are often geographically distant, requires an analysis of their dynamics. This analysis concerns both the cognitive aspects (the problems solved by the communities in the context of frequently ill-structured business processes) and the social aspects (at times informal structuring in terms of affinities, competencies, and common interests).

Since collective activity is mediated by documents, it is possible to analyse the traces of this activity left in documents. This analysis may serve several purposes. Among other things, it may be used to understand the evolution of the cognitive and social dynamics of the community. Traces of collective activity may stem from extremely diverse sources. They are more easily understood and analysed if the system keeps track of the successive changes, as in Wikis or word processors offering a « revision » mode. For example, in the scenario presented in this paper, we will consider the collective writing of a file whose successive versions are exchanged by e-mail. We will therefore assume that the successive versions have been archived.

Our purpose is twofold :
1. Define a formal model of documentary traces. This model is inspired by research from the domain of information systems traceability [ZP01], and is based on the conceptual framework of Documents for Action [Za04].
2. Based on the formal model of documentary traces, highlight dependencies between documentary fragments, as well as between authors and readers of these fragments. We will then be able to analyse and measure these dependencies. This analysis may lead to several concrete applications. For example, dependency analysis may be applied to understand the collective dynamics within the community, thereby facilitating the analysis of this community's activity, as well as its regulation (by a manager, a facilitator, or participants themselves). In this paper, we do not elaborate on these practical applications of dependency analysis. However, we plan to investigate them as further research.

---

[1] Different types of communities are mentioned in the CSCW (Computer Supported Cooperative Work), sociology and management literature : community of practise, community of interest, epistemic community, etc.

The paper is organised as follows. After justifying the choice of the DofA (Documents for Action) conceptual framework, we will elaborate a model of DofA based on UML [Ob06]. We will use this model as a structure for conceptually representing traces of mediated collective activities. We will pursue with the issue of dependency analysis and measurement, leading us to propose specific dependencies in the context of DofA. We will illustrate the early steps of an application to a scenario of collective writing. Finally, we will discuss further research directions suggested by our approach, which analyses and measures dependencies from traces of mediated collective activity.

2 Towards a model of Documents for Action

2.1 The concept of DofA: a semiotic product associated with a communicational transaction

The DofA (Documents for Action) conceptual framework has several advantages when it comes to analysing documentary information systems supporting the coordination of mediated communities. DofA account for the extreme diversity of documents and the various phase of their lifetime (e.g. writing or reading), especially in the context of Internet (accelerated file transfer via e-mail, Web content management systems, newsgroups, on-line annotation, weblogs…). While hypertext focused attention on the conditions of hyper-reading, digitised Documents for Action focus attention on « hyper-writing »: how are simultaneously evolving fragments of documents drawn together and made coherent through the collective activity which they were intended to mediate?

The concept of DofA thus redefines the notion of document. It also accounts for the extreme variety of document contents, which are more and more composed of multimedia data (e.g. image or sound), as opposed to plain text. By focusing on the collective dimension of the writing activity, the concept of DofA enables to analyse documents as resulting from partly asynchronous communication processes between producers and recipients who are partly bound by common interests. This communicational approach of documents is inspired by the notion of symbolic communicational transaction. The latter considers the document as resulting from a transaction between actors implied in an exchange process. In this exchange process, the « self » and the knowledge of actors come into play, resulting in the joint[2] production of a « work » [Za04].

---

[2] at least partly…

When describing an exchange in terms of communicational transactions, we consider a self playing the role of « producer » and a self playing the role of « recipient ». The relationship between the producer and the recipient is mediated through their jointly produced semiotic products. Further analysis of the semiotic product leads to distinguish between (1) the form and substrate i.e. the medium of this semiotic product and (2) the semiotic content conveyed by this medium. Thus, a semiotic product is composed of a semiotic content (e.g. the story of King Kong) and a medium, which is in turn decomposed into a mode of expression (oral narrative, written text, film) and a substrate type (presence of the narrator or voice recording for oral narrative, paper or electronic document for text, DVD or VHS for film, etc.). Naturally, the choice of medium will influence the « power of evocation » associated with the semiotic content.
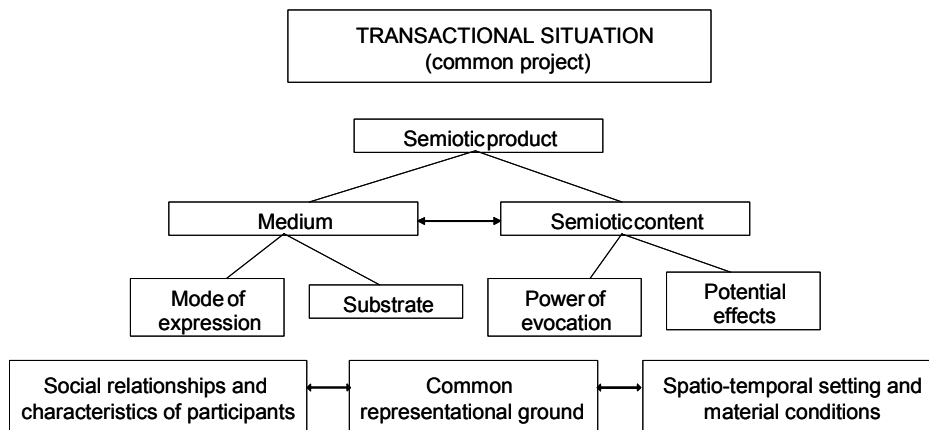

Figure 1: Components of the semiotic product (adapted from [Za04], [Za06b])

In widely distributed collective activities, producers and recipients of communicational transactions are not all in the same spatio-temporal framework. This means that the documents produced must last over time if they are to be used as a support for transactions. Transactions are initiated, interrupted, updated and repeated in configurations involving the presence or absence of producers and recipients. One strategy to compensate for spatio-temporal distribution is the strategy of documentarisation, which lies on the transcription and recording of the semiotic product on a perennial material substrate. However, transcription and recording procedures do not suffice for the purpose of documentarisation. Personal note-taking or an occasional recording intended to facilitate a semiotic activity in a given transactional situation can occasionally be useful. However, if the documentary investment required by the process of documentarisation has not been made, it will not be possible to go back to using these expedients in subsequent distributed transactional situations.

Transactional objectives
focusing on the work
and/or the self

The creative
transaction

Individuation of
the productive
self

The jointly produced
semiotic/informational
& technical/material
work

Individuation of
the beneficiary
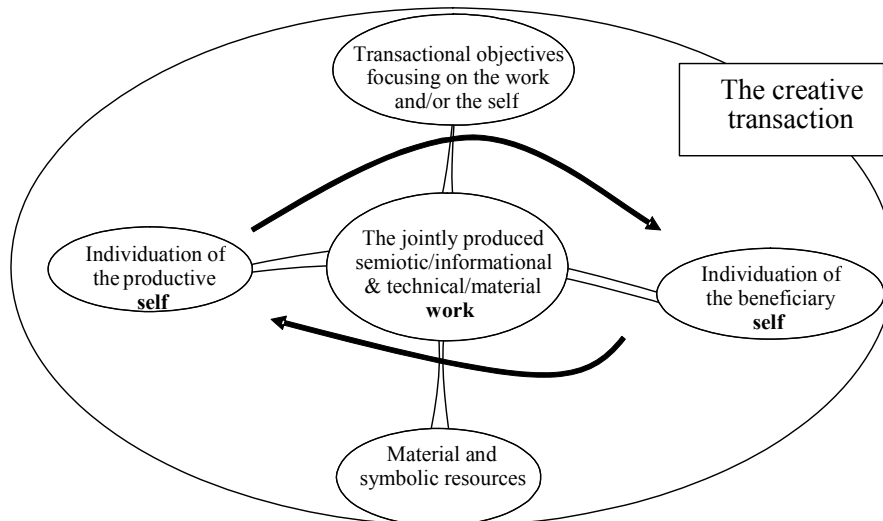**self**

Material and
symbolic resources

Figure 2: Diagram of the components of a creative transaction [Za06b]

Documentary investment, aka documentarisation of the semiotic product, consists in endowing the perennial substrate with specific attributes intended to facilitate its subsequent utilisation in the framework of distributed communicational transactions. These attributes (author's name, date, title, subtitles, version…) make it possible for the document to move through time and space among the communities of interpretation, with a view to prolonging and extending the communicational transactions initiated by its producers. When the document is written by a collective entity distributed in time and space, its is composed of several fragments. These fragments are themselves endowed with specific attributes through which they can be explicitly linked up with the other components of the document. Documentary fragments constitute DofA.

Various examples of DofA may be found in numerous professional contexts [Za04]. Some examples of DofA taken from different professional contexts are engineering design documents (mechanical descriptions, software programs); patients' case-records in the field of health care; business proposals which are gradually transformed into definite, formal contracts; digitised quality assurance documents; management consulting reports; and even the open source software forums described by [RS06].

The main characteristics of DofA can be defined as follows :
1. Extended state of incompletion: they go through a long process of completion during the active collective semiotic production phase, during which they are called evolving DofA (as opposed to stabilised DofA);
2. Perenniality: this characteristic is due both to the participants' commitment to the semiotic content of these documents, and to the widely distributed nature of the transactions, which gives rise to specific documentarisation problems, and hence to storage and indexing problems;

3. Fragmentation: at least during the evolving phase, they contain several fragments which are often only loosely semantically linked (especially in the case of annotations), and which cannot be mechanically and implicitly integrated into the document (see above);

4. Non trivial relationships between DofA fragments and their producers: the various parts of DofA are often produced by different authors (they can therefore be said to be plurivocal or pluritextual), who have different statuses in the transactional situation, and therefore have different rights to the semiotic product;

5. Non trivial argumentative relationships between the document fragments: each fragment stands in a potentially complex relationship with the others, depending on the modes of expression used, the level of certainty or uncertainty expressed, and the logical links with the other fragments (such as the presence of contradictory statements), for example.

2.2 Relationships between a fragment and the rest of the DofA

In the context of distributed communicational transactions which are mediated by DofA, the contributions take the form of fragments. These fragments are articulated with the main semiotic product (the main text, in most cases) with varying degrees of success. The fragment goes through different states (*status*) corresponding to its degree of integration with the body of the document. When the fragment is not clearly articulated – or not articulated at all – with the rest of the document, it is called a *free fragment*. When the fragment is endowed with attributes clarifying its relationships with other fragments or with the main semiotic product, it is called a *documentarised fragment* (aka contributing annotation). Finally, when the fragment is fully integrated into the main semiotic product of the DofA – by linking it up either implicitly (by simply adding it to the end of the text, for example) or explicitly (by numbering or referencing it, for example) –, it is called a *part of document*.

The attributes or links through which a fragment is documentarised (i.e. anchored to the DofA) may also be considered as auxiliary fragments and, consequently, as annotations. They are called *indexing annotations* [Za06a], to distinguish them from contributing annotations. An indexing annotation does not contribute directly to the main semiotic content. It aims at completing a document or a fragment in order to ease its articulation with other documents (externally), or with other parts or fragments of the document (internally). Indexing annotations, which correspond to the attributes or links through which documentarisation is achieved, may take various forms : name of the author (producer) of the fragment, date, graphical relationship representing a semantic relationship, place of production, keywords from a thesaurus, concepts from an ontology, etc. The UML model of Figure 3 formally represents traces of mediated collective activities. We assume that these collective activities are mediated by DofA. Consequently, our model is based on the conceptual framework of DofA, which we have just described.

Figure 3: Model of DofA (Documents for Action)

3 Dependency analysis and measurement

The formal model of DofA is at the root of our semi-automated approach for dependency analysis. We are interested in dependencies between fragments, or between the selves producing or receiving these fragments. This analysis may serve several purposes. In particular, it may be applied to adapt the coordination mechanisms (in the sense of T.Malone and K.Crowston [MC94] [Cr03]) used by the mediated community to reach its goals.

The analysis of dependencies and coordination mechanisms is a crucial issue to many disciplines, among which management [Mi79] and computer science. Based on a multidisciplinary analysis, T.Malone and K.Crowston [MC94] [Cr03] propose a taxonomy of dependencies and associated coordination mechanisms. Coordination is defined as the management of dependencies. Among the dependency types identified by the authors, let us mention the following:

1. flow dependency (one task uses a resource created by another task)
2. sharing dependency (multiple tasks use the same resource)
3. adjustment dependency (multiple tasks create the same output)
4. composition dependency between tasks (one task is broken down into subtasks).

The characterisation of dependencies may serve several purposes. In particular, it may guide the choice of the best suited coordination mechanisms and tools (e.g. groupware), depending on the type of dependency.

In information systems, [YM93] proposes an actor dependency model. This model is part of a more global framework which aims at supporting the requirements engineering process ; it also applies to business process reengineering. A dependency is an oriented link between two actors. It centres around a particular object. The type of this object (goal, task or resource) permits to distinguish different dependency types. The model provides for different degrees of strength of dependencies. However, no metric is defined to quantify these degrees of strength. Dependency between agents is also central to [Su00], which proposes a requirements analysis method for socio-technical system design. Dependencies between agents are studied in order to determine the degree of coupling between agents. Coupling metrics are applied to assess the degree of dependency between the system and the agents (users). These metrics are used to assist decision making when choosing between different system scenarios.

In software engineering, many metrics have been proposed to assess the cohesion of modules. Cohesion (modularity) eases reuse as well as maintenance. Concerning the object-oriented approach, numerous metrics have been proposed to measure class cohesion [Et04]. [Wa05] proposes such a metric, which is based on a typology of dependencies between elements (methods and attributes of a class). In essence, the approach is decomposed into three steps : (1) build the dependency graph between elements of a class, (2) compute the square matrix of (direct and indirect) dependencies for each couple of elements in the class, and (3) from this matrix, compute the global cohesion of the class. Finally, in the domain of databases, data may be placed in memory based on an analysis of dependencies between them. This way, data access performance may be improved, particularly in navigational multimedia applications [Va03].

In our approach, dependency analysis and measurement is based on DofA, represented with the model described in the previous section. A dependency is an oriented link from one object A to one object B, such that A depends on B. For the management of mediated communities, the most relevant dependencies are those related to fragments and selves. Therefore, three major types of dependencies may be defined : (1) dependencies between two fragments, (2) dependencies between one fragment and one self, and lastly (3) dependencies between two selves.

Our approach is based on a typology of dependencies, defined from the model of DofA. Starting from one DofA, our approach for dependency analysis and measurement is decomposed into the following steps:

1. Clarify (i.e. make explicit) in the DofA all the links (indexing annotations) which were only implicit so far. This step mostly deals with traceability links (when it is not possible to determine these links automatically).
2. Generation of the dependency graph for the DofA. Once all the links have been clarified, this generation may be performed automatically. In the dependency graph, nodes represent fragments or individual selves; edges represent direct (aka canonical) dependencies between two fragments, or between one fragment and one individual self.
3. From this graph, compute two dependency matrices: (1) one matrix of dependencies between individual selves and (2) one matrix of dependencies between fragments (in case there exist several successive versions, interrelated by a traceability link, only the latest version of fragments is considered). The two dependency matrices are built taking into account both the direct and indirect dependencies in the dependency graph.

In the rest of this section, we focus more particularly on the typology of dependencies, which is central to our approach. We present an application of the approach to a simple scenario (collaborative writing of a paper for a workshop).

3.1 Typology of dependencies

We make a distinction between (1) canonical (i.e. direct) dependencies, defined from the model of DofA presented in section 2 and (2) indirect dependencies, obtained by combining canonical dependencies.

Canonical dependencies

These dependencies are defined from the links that exist in the DofA between two fragments, or between one fragment and one self (Table 1). In this table, the representation of each dependency is supplemented by its formal definition with OCL (Object Constraint Language), the constraint language associated with UML [Ob06]. The OCL expressions show how each dependency is formally defined from the model of DofA presented above.

DEPENDENCIES BETWEEN TWO FRAGMENTS	
Dependency	**Expression**
F1 —references→ F2	dofA.annotations→exists(RL:Indexing annotation | RL.oclIsTypeOf(Reference link) **and** RL.source=F1 **and** RL.target=F2)
F1 —derives from→ F2	dofA.annotations→exists(TL:Indexing annotation | TL.oclIsTypeOf(Traceability link) **and** TL.source=F1 **and** TL.target=F2)
F1 —composed of→ F2	dofA.annotations→exists(CL:Indexing annotation | CL.oclIsTypeOf(Composition link) **and** CL.composite=F1 **and** CL.component=F2)
F1 —composes→ F2	dofA.annotations→exists(CL:Indexing annotation | CL.oclIsTypeOf(Composition link) **and** CL.component=F1 **and** CL.composite=F2)
DEPENDENCIES BETWEEN ONE FRAGMENT AND ONE SELF	
Dependency	**Expression**
F1 —produced by→ S1	dofA.productions→exists(Pro:Production | Pro.product=F1 **and** ((Pro.producer.oclIsTypeOf(Individual self) **and** Pro.producer=S1) **or** (Pro.producer.oclIsTypeOf(Collective self) **and** Pro.producer.made up of→includes(S1)))
S1 —recipient of→ F1	dofA.receipts→exists(Rec:Receipt | Rec.product=F1 **and** ((Rec.recipient.oclIsTypeOf(Individual self) **and** Rec.recipient=S1) **or** (Rec.recipient.oclIsTypeOf(Collective self) **and** Rec.recipient.made up of→includes(S1)))

Table 1 : Canonical dependencies

Among the links between two fragments, only the composition link results in two dependency links. A reference link (including the particular case of traceability link) results in a dependency from source to target.

Concerning dependencies between one fragment and one self, two types of dependencies are defined, similarly to the dependencies defined in DMC [Wa05] between methods and attributes that are either read or written by these methods.

Indirect dependencies

These dependencies are defined by combining canonical dependencies using combination rules. These rules are far from evident. In particular, dependency is not always transitive. Among other things, combination rules depend on the semantics of the considered dependencies. Table 2 illustrates some examples of rules to infer indirect dependencies from canonical dependencies.

DEPENDENCIES BETWEEN TWO FRAGMENTS	
Obtained by combining canonical dependencies between fragments.	
DEPENDENCIES BETWEEN TWO SELVES	
If	**Then**
S1 —recipient of→ F1 **and** F1 —produced by→ S2	S1 —depends on→ S2
S1 —recipient of→ F1 **and** F1 —derives from→ F2 **and** F2 —produced by→ S2	S1 —depends on→ S2
S1 —recipient of→ F1 **and** F1 —composed of→ F2 **and** F2 —produced by→ S2	S1 —depends on→ S2
F1 —produced by→ S1 **and** F1 —produced by→ S2	S1 —depends on→ S2 **and** S2 —depends on→ S1
S1 —depends on→ S2 **and** S2 —depends on→ S3	S1 —depends on→ S3

Table 2 : Indirect dependencies

Concerning dependencies between two fragments, indirect dependencies may be obtained by combining canonical dependencies, even if these canonical dependencies are not of the same type (for example, if a fragment F1 references a fragment F2 which is itself composed of a fragment F3, then F1 depends on F3). However, the algorithm should avoid circular references, e.g. by always following composition dependencies in the same direction (from component to composite or vice-versa).

The first rule of indirect dependency between two selves, results from the combination of the two canonical dependencies between fragments and selves. This rule – as well as the second rule – may be compared with the flow dependency defined in [MC94]. The third rule is similar to the composition dependency between tasks in [MC94], and the fourth is similar to the adjustment dependency (necessary coordination between two selves in a situation of co-production). Finally, the last rule of indirect dependency between two selves, generates new dependencies by combining the first four rules.

When combining dependencies, the algorithm should ensure that the same dependency is not counted twice. To achieve this, dependencies should be considered at the most detailed level. For example, an individual self producing a component of a fragment is – by definition – one of the producers of this fragment ; the corresponding dependency should only be counted once, by considering only the dependency « produced by » between the component of the fragment and the individual self.

3.2 Applying the approach for dependency analysis and measurement

In order to illustrate the application of our approach, we present a simplified example of collective writing of a workshop paper. The paper has three co-authors. The subject of the paper is the presentation of a Web site design method. The three authors use Word, and make use of the various possibilities proposed in this software for structuring documents properly (e.g., explicit definition of the hierarchical structure of the document – sections and sub-sections –, utilisation of cross-references when referencing figures or sections…). The communication is both face-to-face and virtual (e-mail transmission of the file at each stage of writing).

The article written by the three co-authors is organised as follows:
1. Introduction
2. The graphical notation (this section contains a figure illustrating the graphical notation used in the Web site design method, as well as an accompanying text).
3. The method
   3.1. Presentation of the method
   3.2. Example (i.e. an application of the method)
4. Conclusion/discussion.

The scenario of collective writing is the following:
1. Author3 makes the figure of section 2 and the accompanying text.
2. Based on this first version of the paper, the three authors meet and write section 3.1 together.
3. Author1 and Author2 then write the example (section 3.2) together ; they send the resulting complemented file to Author3.
4. Author3 completes the example, and writes the conclusion/discussion and the introduction. He sends the resulting completed file to Author1 and Author2.
5. Author1 and Author2 reread the paper and complete the introduction (by adding the presentation of the outline of the paper). Finally, they send the final version to the workshop chair.

We proceed in three steps, applying our dependency analysis and measurement approach to this example:

1. We clarify all the links which were only implicit so far. In our case, there are few implicit links, since we have assumed that the authors have used all the possibilities offered by Word in terms of document structuring. Moreover, the producers and recipients of the different fragments may be partly inferred from (1) the properties of the successive versions of the article (Word file) and (2) the « From » et « To » headers of the exchanged mails. Concerning traceability links, these links need to be represented explicitly.

2. We generate the dependency graph for the DofA of our example. This graph is shown in Figure 4. To improve readability, Figure 4 makes a distinction between canonical dependencies between two fragments (left part of the figure) and canonical dependencies between an individual self and a fragment (right part of the figure).

3. Based on this graph, two dependency matrices may then be computed: (1) one matrix of dependencies between the 4 individual selves and (2) one matrix of dependencies between the fragments (concerning the introduction and section 3.2, only version 2 is considered).
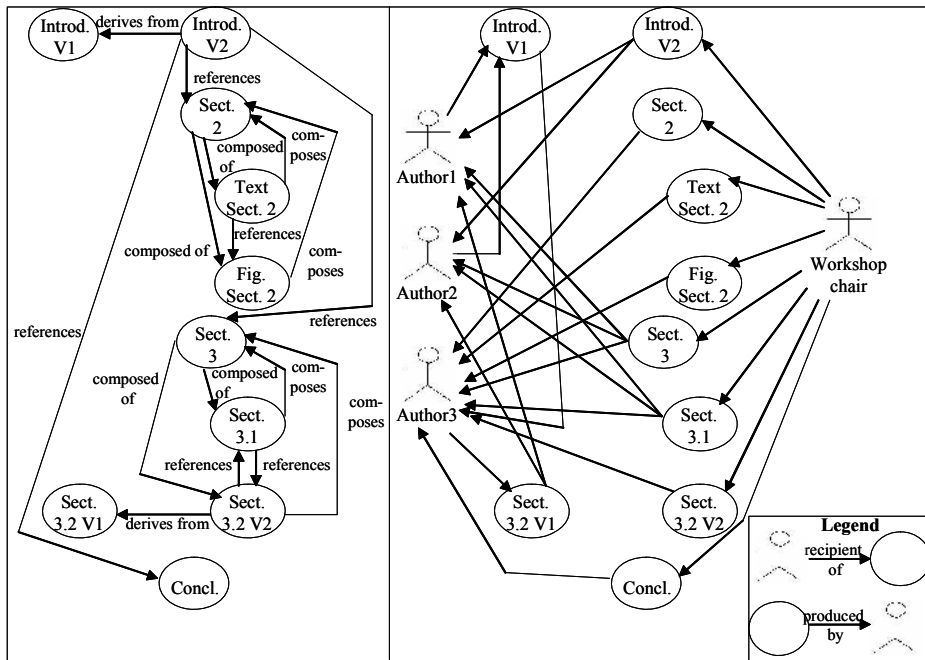

Figure 4: Dependency graph of the example

4 Discussion and conclusion

The conceptual framework of documents for action and communicational transactions, presented in section 2, may seem rather complex. However, we believe this complexity reflects the very complexity inherent to the production of documents by mediated communities; the organisation of these mediated communities is itself complex, emerging, and constantly changing. Our further research will be devoted to two complementary objectives: (1) refine our approach for dependency analysis and measurement and (2) empirically assess its relevance.

Concerning the first objective, we will pursue several directions :

1. First, we believe it is necessary to further characterise dependencies. This may be achieved by considering in greater detail the semantics of links in the model of DofA. For example, reference links should be detailed further in order to specify the semantics of the reference (for the moment, the only type of reference link explicitly represented in the model of DofA is the traceability link). Similarly, we believe it is relevant to specify more precisely the semantics of the links between the fragments and the selves producing or receiving these fragments. For example, we may distinguish different types/roles of producers (e.g. main author/second author, controller, writer, signing authority, etc.). These different type of producers all bring a specific contribution to the production of the fragment. Consequently, the dependency of a fragment vis-à-vis a producer depends on the role of the producer.

2. In order to refine our approach, we believe it would also be relevant to associate weights to the different dependency types. We could then combine these weights (if A depends on B with weight/coefficient p1 and B depends on C with coefficient p2, then A depends on C with coefficient p1*p2). The determination of weights for the different dependency types requires more detailed typing of dependencies; it may also be based on heuristics. For example, considering the composition link between two fragments, it seems (a priori) that the dependency of the composite vis-à-vis the component is stronger than the opposite dependency.

399

Concerning the objective of empirical evaluation, we plan to assess the main transactional situations mediated by DofA. This will enable us to chose a real application field. Based on this assessment, we will also be able to provide recommendations for the specification of environments supporting collective production of documents and the traceability of this production. Many situations of collective document production may be analysed through traces. Today, the most commonly studied example concerns communities of open source software developers cooperating through newsgroups ([Sa06], [RS06]). The early results of this research are promising, concerning (1) the identification of patterns for the mediated resolution of problems (in this case, software bugs) [RS06] and (2) improved understanding of the associated social processes [Sa06]. By presenting a formal model of DofA and by applying it to a scenario of mediated writing in a small workgroup, we believe we have presented strong arguments supporting further research on documentary traces. These traces may support the animation and management of mediated communities of actions, which concern an increasing number of professional situations. The formal model of DofA which we have presented is generic. Thus, it applies to collective writing as illustrated in this paper, to newsgroups in the context of open source software development… Our approach for dependency analysis and measurement is based on the model of DofA. Consequently, this approach is also generic. However, we will have to refine this approach in order to represent more precisely the dependency types, the associated weights, and the composition rules. Once the dependencies between fragments and between individual selves have been measured, the information provided by these measures may be used to improve the management of mediated communities of action: optimisation of the presentation and/or indexing of documents based on the dependencies between fragments, detection of « abnormal » situations of dependency between selves (« abnormal » meaning that the measures of dependencies reveal imbalance, or contradictions with the organization chart…), etc. This way, empirical evaluation will not only enable us to refine/calibrate our approach, but also to illustrate concretely the usefulness of dependency analysis and measurement.

Bibliography

[Cr03]   Crowston, K.: A Taxonomy of Organizational Dependencies and Coordination Mechanisms. In (Malone, T.; Crowston, K.; Herman, G., eds): Tools for Organizing Business Knowledge: the MIT Process Handbook. MIT Press, Cambridge, Massachusetts, 2003; pp. 85-108.

[Et04]   Etzkorn, L.H. et al.: A Comparison of Cohesion Metrics for Object-Oriented Systems. Information and Software Technology, volume 46, issue 10, August 2004; pp. 677-687.

[MC94]   Malone, T.; Crowston, K.: The Interdisciplinary Study of Coordination. ACM Computing Surveys, volume 26, issue 1, March 1994; pp. 87-119.

[Mi79]   Mintzberg, H.: The Structuring of Organizations : a Synthesis of the Research. Prentice-Hall, Englewood Cliffs, New Jersey, 1979.

[Ob06]   Object Management Group: Unified Modeling Language (UML) specification, version 2.0. http://www.omg.org/technology/documents/formal/uml.htm.

[RS06]   Ripoche, G.; Sansonnet, J.P.: Experiences in Automating the Analysis of Linguistic Interactions for the Study of Distributed Collectives. Computer Supported Cooperative Work, volume 15, issue 2-3, June 2006; pp. 149-183.

[Sa06]   Sack, W. et al.: A Methodological Framework for Socio-Cognitive Analyses of Collaborative Design of Open Source Software. Computer Supported Cooperative Work, volume 15, issue 2-3, June 2006; pp. 229-250.

[Su00]   Sutcliffe, A.: Requirements Analysis for Socio-Technical System Design. Information Systems, volume 25, issue 3, May 2000; pp. 213-233.

[Va03]   Vakali, A. et al.: Hierarchical Data Placement for Navigational Multimedia Applications. Data & Knowledge Engineering, volume 44, issue 1, January 2003; pp. 49-80.

[Wa05]   Wang, J. et al.: DMC: a more Precise Cohesion Measure for Classes. Information and Software Technology, volume 47, issue 3, March 2005; pp. 167-180.

[YM93]   Yu, E.; Mylopoulos, J.: an Actor Dependency Model of Organizational Work – with Applications to Business Process Reengineering. In Proc. Conf. on Organizational Computing Systems (COOCS 93), Milpitas, California, USA, November 1993. ACM Press, New-York; pp. 258-268.

[Za03]   Zacklad, M.: Communities of Action: a Cognitive and Social Approach to the Design of CSCW Systems. GROUP'2003, 09-12 November 2003, Sanibel Island. ACM, 2003; pp. 190-197.

[Za04]   Zacklad, M.: Processus de Documentarisation dans les Documents pour l'Action (DopA) : Statut des Annotations et Technologies de la Coopération Associées. Le Numérique : Impact sur le Cycle de Vie du Document pour une Analyse Interdisciplinaire http://archivesic.ccsd.cnrs.fr/sic_00001072.html, 13-15 October 2004, Montréal; 19 pages.

[Za06a] Zacklad, M. et al.: Processus d'Annotation dans les Documents pour l'Action : Textualité et Médiation de la Coopération. In (Charlet, J., ed): Documents et Contenu : Création, Indexation, Navigation. Hermès, Paris, 2006.

[Za06b] Zacklad, M.: Documentarisation Processes in Documents for Action (DofA): The Status of Annotations and Associated Cooperation Technologies. Computer Supported Cooperative Work, volume 15, issue 2-3, June 2006; pp. 205-228.

[ZP01]   Zamfiroiu, M.; Prat, N.: Traçabilité du Processus de Conception des Systèmes d'Information. In (Cauvet, C.; Rosenthal-Sabroux, C., eds): Ingénierie des Systèmes d'Information. Hermès, Paris, 2001; pp. 245-276.