

Conceptual Modeling of Web Application Transactions: Towards a Revised and Extended Version of the UWA Transaction Design Model

Damiano Distante

*Dept. of Innovation Engineering
University of Lecce, Italy
damiano.distante@unile.it*

Scott Tilley

*Dept. of Computer Sciences
Florida Institute of Technology, USA
stilley@cs.fit.edu*

Abstract

Business Web applications are complex data- and operation-intensive Web applications intended to support business processes. In this kind of application, the user executes sets of activities organized into transactions (also generally called workflows) in order to accomplish one of his/her tasks or fulfill one of his/her goals. The way these transactions are structured, the properties associated with their component activities and the way each activity is implemented by the Web application are a consequence of the transaction design as well as the business rules implied by the business process. The Ubiquitous Web Applications (UWA) design framework explicitly includes a design activity and a related meta-model, called the Transaction Design Model, specifically focused on the conceptual design of Web application transactions (Web Transactions for short). This paper describes preliminary steps towards a revised and extended version of the UWA Transaction Design Model that emphasizes a user-centered approach, is easier to adopt and manages the information, navigation and operation aspects involved in a Web Transaction in a comprehensive and integrated way.

1. Introduction

Web applications have rapidly evolved over the past few years from brochure-like read-only Web sites to complex data- and operation-intensive applications intended to support businesses.

From e-commerce Web sites to complex Web-based information systems, a large range of business applications, designed to be accessed via a Web browser and to exploit the Internet as a means of communication, are heavily reliant on the execution of transactions.

We use the term *Web Transaction* to indicate a

sequence of activities performed by a user in order to carry out a specific task or fulfill a specific goal related to a business process by means of a Web application. In this definition, a Web Transaction can be considered a specialization of the “workflow” concept in the context of Web applications and represents a way of implementing business processes (or portions of them) via a Web application.

In a Web Transaction the execution of operations and the navigation of contents are combined. The resulting complexity of Web Transactions requires them to be designed prior to being implemented. Neglecting Web Transactions during the design process of the Web applications often results in applications with poor usability and undesired behaviors [1][2][10].

In response to the above mentioned needs, a number of models and methods, formerly proposed for the design of informational hypermedia applications, have been extended in order to specifically address Web Transaction and business process design, e.g. the Object-Oriented Hypermedia Design Method (OOHDM) [1][2], the Web Site Design Method (WSDM) [3], the Object Oriented Hypermedia (OOH)[4], and the Araneus Data Model (ADM) [5].

The Ubiquitous Web Applications (UWA) design framework [8][14] which can be considered an evolution of the Hypermedia Design Model (HDM) [6] and the W2000 methodology [7] includes a design activity (Transaction Design) and a related meta-model (the Transaction Design Model) specifically focused on the conceptual design of Web Transactions [17].

Based on Unified Transaction Modeling Language (UTML) [9] and concepts derived from database theory, the UWA Transaction Design Model is one of the most rigorous models currently available in the literature for Web Transaction design. Despite its rigorousness, we found this model to be not truly user-centered like the remainder of the UWA framework and to fail in addressing issues of content navigation during the execution of a Web Transaction.

This paper describes the first steps towards a revised and extended version of the UWA Transaction Design Model that emphasizes a user-centered design approach and better addresses the navigational issues of a Web Transaction.

The paper is structured as follows. In section 2 a minimal list of requirements for a conceptual user centered Web Transactions design meta-model is proposed. Section 3 gives an overview of the UWA Transaction Design model and highlights the strengths and shortcomings we found in it. Section 4 presents the changes and the extensions we introduced in the UWA Transaction Design Model for overcoming the identified shortcomings. Finally section 5 summarizes the paper and highlights avenues for future work.

2. Requirements of a Web Transaction Conceptual Design Meta-Model

It is common practice during development of Web applications to treat Web Transactions as just another kind of navigation. In some cases Web Transactions are completely neglected, the functionalities of the application being designed and developed singly and the Web Transactions being “by-products” of the development process. Both these approaches result in poor quality and error-prone Web applications as documented in the literature [1][2][10].

On the one hand Web Transactions are intended to provide the user with far more than just information. They involve the execution of activities under the rules of a business process in order that the user may accomplish his/her tasks. For this reason they cannot be designed as navigation paths through the contents of the application.

On the other hand, Web Transactions also involve content navigation and, thus, cannot be simply designed by adopting the concepts of database theory or the modeling tools provided by UML [11].

In designing Web Transactions, navigational and transactional semantics need to be considered at the same time and need to be integrated in a comprehensive approach. The OOHDM [1][2], OO-H and UWE [4] design methodologies propose approaches and models for Web Transaction design which go in this direction.

The experience acquired in a number of case studies adopting UWA [12][13][20] and the analysis of the above-mentioned methodologies allowed us to build a representative list of requirements for a suitable meta-model for user-centered design of Web Transactions. Such a meta-model should enable the designer to:

1. Represent the set of activities (partially derived

from the business process model) involved in a Web Transaction, their composition, semantic relations and properties.

2. Describe the overall execution flow of the transaction and the single execution rules associated with each component activity.
3. Specify, in accordance with the business rules available for the application, which activities can be suspended and resumed afterwards during long-lived transactions.
4. Describe the way two or more types of user involved in a Web Transaction collaborate in its execution.
5. Specify the way content navigation and activity execution affect each other during a Web Transaction.
6. Define which contents will be provided to the user in order to support the execution of a particular activity and which information objects are affected by the execution of the activity.
7. Describe the way an activity will be customized depending on the context of execution and the state of the ongoing transaction.

In addition to the previous list of requirements, a Web Transaction design meta-model that aims to be user centered and work at a conceptual level must also:

1. work at a high level of abstraction, focusing on aspects important to the user and enabling the designer to describe these as the user perceives them. Such a meta-model would facilitate the design of Web Transactions that are closer to user expectations and have improved usability [12][13].
2. be easy to create and easy to be read both by the client and the programmers who will develop the application.

Though not exhaustive, the above list of requirements represents a good starting point for defining a meta-model for the conceptual design of Web Transactions as well as for evaluating existing ones. We use this list to describe the strengths and shortcomings of the UWA Transaction Design Model [17] as found in a number of case studies in which it was adopted.

3. The UWA Transaction Design Model

3.1 The Original Meta-Model

The Ubiquitous Web Applications (UWA) design framework provides a complete design methodology and a set of meta-models and design tools for designing Web applications that are multi-channel, multi-user, and generally context-aware, in a word

ubiquitous. UWA organizes the design process of a Web application into four main activities [14]:

1. *Requirements Elicitation* [15][7]: intended to define the set of stakeholders of the application and their goals, and to derive the application requirements.
2. *Hypermedia and Operation Design* [16]: including the following four design sub-activities:
 - a. *Information Design*: to identify the contents of the application and their structure.
 - b. *Navigation Design*: to define the elementary “granules of information” (*Navigation Nodes*) the user will navigate between and their “accessibility rules” (*Navigation Paths*).
 - c. *Operation Design*: to design the operations the application will provide to the user.
 - d. *Publishing Design*: to define how the application will be organized into pages.
3. *Transaction Design* [17]: the specific activity for designing Web application transactions whose meta-model we propose to revise and extend here.
4. *Customization Design* [18]: to define the way the application is customized depending on the context of usage (device, user, location, etc.).

The UWA Transaction Design meta-model is based on the concepts of *Operation*, *Activity* and *Execution Contracts* [17]. An Operation is defined as a non-suspendable atomic unit of work performed by the user or the system. An Activity is defined as a set of operations and possibly other Sub-Activities with an optional execution flow. Execution Contracts are the properties that an Activity satisfies and that determine its type. For each of the previous primitives, a notation and a set of properties is provided.

Though not formally stated, in the UWA Transaction Model a Web Transaction corresponds to a complex Activity composed of Sub-Activities with an associated execution flow.

The design of a Web Transaction by means of the UWA Transaction Design model produces basically two models, the *Organization Model* and the *Execution Model*, in which the previous concepts are adopted.

The Organization Model describes the Web Transaction from a static point of view, modeling its hierarchical organization in terms of Activities and Sub-Activities, the relations among them and their properties. The Organization Model is a particular type of UML Class Diagram [11] in which Activities are arranged to form a tree. The main Activity at the root of the tree represents the whole Web Transaction and the intermediate nodes and the leaves of the tree are its component Activities and Sub-Activities. The *Property-Set* of a given Activity is the subset of the properties of Isolation, Atomicity, Durability and

Consistency (ACID) which the Activity verifies. The semantics associated with the Property-Set are well known in the literature of database theory [21][23]. The labels attached to the tree branches of the Organization Model define the hierarchical relations (also referred to as associations) between a Sub-Activity and its ancestor Activity. For each Activity an *Operation-Set* specifies the set of operations that are functional to the management of the Activity (*Management-Set*) and the set of operations that are intended to implement the logic of the Activity itself (*Functional-Set*). Their semantics are also derived from database theory.

The Execution Model defines the possible execution flows among the component Activities of a Web Transaction. The Execution Model is a customized version of the UML Activity Diagram [11] in which Activities and Sub-Activities are represented by states (ovals), and the execution flow among them by state transitions (arcs). The model describes both the evolution of the state of the Web Transaction from a system point of view and the rules (chronological and logical) regulating the execution of each Activity.

3.2 Strengths and Shortcomings

Strengths

Several aspects of the UWA Transaction Model make it one of the most advanced and rigorous models in the literature. Its strengths include the following:

1. *Systematic approach*: the model provides the designer with a systematic approach for designing Web Transactions. A Web Transaction is associated with each “procedural” goal resulting from the Requirements Elicitation and is designed first from a static point of view by means of the Organization Model and then from a dynamic point of view using the Execution Model.
2. *Rigorous formalism and solid theoretical basis*: the set of concepts used by the model derive from the theory of database transactions [21][23] and the UTML meta-model [921].
3. *Standard notation*: the Organization and Execution models adopt a standard notation since they are basically UML diagrams [11].
4. *Extensibility*: the model is extensible to support Web Transaction models for specific application domains and requirements.
5. *Tools supported*: the model is supported by tools to facilitate its adoption. One of these is a Rational Rose Add-in [19] to draw this kind of diagram.

Shortcomings

Along with these strengths, in our experience of applying the UWA Transaction Model in a number of

reverse and forward engineering case studies, we have found a number of major shortcomings. With reference to the list of requirements formulated in section 2, the shortcomings of the model are:

1. It doesn't provide any way of specifying the interplay between Activity execution and content navigation within a Web Transaction.
2. It doesn't provide any way of specifying which contents and which informative objects are involved/necessary when executing an Activity.
3. It doesn't provide any way of representing the suspendability of Activities or the collaboration of two or more types of user in the execution of a Web Transaction.
4. It is directly related to the results of the Requirements Elicitation design phase, but doesn't take into account the results of the Information, Navigation and Publishing UWA design activities. Thus it appears poorly integrated in the UWA design methodology as a whole.
5. It takes account of both user and system/data related aspects and is thus not truly user-centered. For example, the semantics associated with the ACID Property-Set and the Functional and Management Operation Sets of an Activity are directly derived from database theory and are not meaningful for the user nor useful for representing conceptual issues.

4. A Revised and Extended Version of the UWA Transaction Model

To overcome the shortcomings identified above and make the UWA Transaction Model a desirable meta-model for the conceptual design of Web Transactions, we have proposed and tested the following revisions and extensions to it:

1. Removal of its data and system centered aspects to emphasize a user-centric approach.
2. Extension of the UWA Navigation Model to enable it to model the interplay between Activity execution and content navigation.

4.1 Emphasizing the User-centric Approach

Changes to the definition of Activity and Web Transaction

Activities modeled by the Transaction Model have to be meaningful for the user of the Web application. System-related activities and data-centered operations are de-emphasized. The OperationSet of an Activity is thus no longer considered in the model.

The *Suspendability* property has been added to the Activity's PropertySet and the semantics associated

with the existing properties tuned to be user centered. An Activity is defined *Suspendable* if its execution can be stopped and resumed afterwards without causing the Web Transaction to be aborted.

The concept of Web Transaction is explicitly defined as the sequence of (user) Activities with the associated execution control flow which enables the user to reach a goal or accomplish a task according to a business process.

Changes to the Organization Model

The possible relations between an Activity a_1 and its Sub-Activities $a_{1.1} \dots a_{1.n}$ have been redefined and divided into two categories: *Hierarchical Relations* and *Semantic Relations*. Hierarchical Relations are the set of "part-of" relations and include: *Requires*, *Requires One*, and *Optional*. Semantic Relations are the set of relationships that are not "part-of", and include *Visible*, *Compensates*, and *Can Use*.

Changes to the Execution Model

The original UWA Execution Model includes both user and system-design directions for developers. Mixing both kinds of design aspect complicates the designer's task and results in a model that is harder for the client to follow. Since our focus is on the user, several changes have been introduced. We purged the model of the system-related aspects, which in our opinion should be specified by a subsequent logical and implementation design phase. The changes we introduced to the model to enable it to focus on conceptual aspects include the following:

1. The two pseudo-states *Commit* and *Rollback* have been removed; the success of an Activity's execution can be verified from the execution flow, while its failure or deliberate interruption is modeled by the "Aborted Activity" pseudo-state.
2. Each possible user-permissible transition between Activities must be explicitly represented in the model with a transition line between them. A simple and extensive labeling mechanism provides information about the user actions that cause the transition and/or the conditions (execution rules) required for it to happen.

These changes to the Execution Model provide a better view of the dynamic that the user will experience while completing a specific Web Transaction. UML Swimlane Diagrams [11] serve to describe how two or more types of user collaborate in the execution of a Web Transaction.

Figures 1 and Figure 2 show respectively the Organization and Execution Models recovered for the "Complete Flight Reservation" Web Transaction on the Alitalia.it [22] airlines company Web site by applying the revised model described in this section.

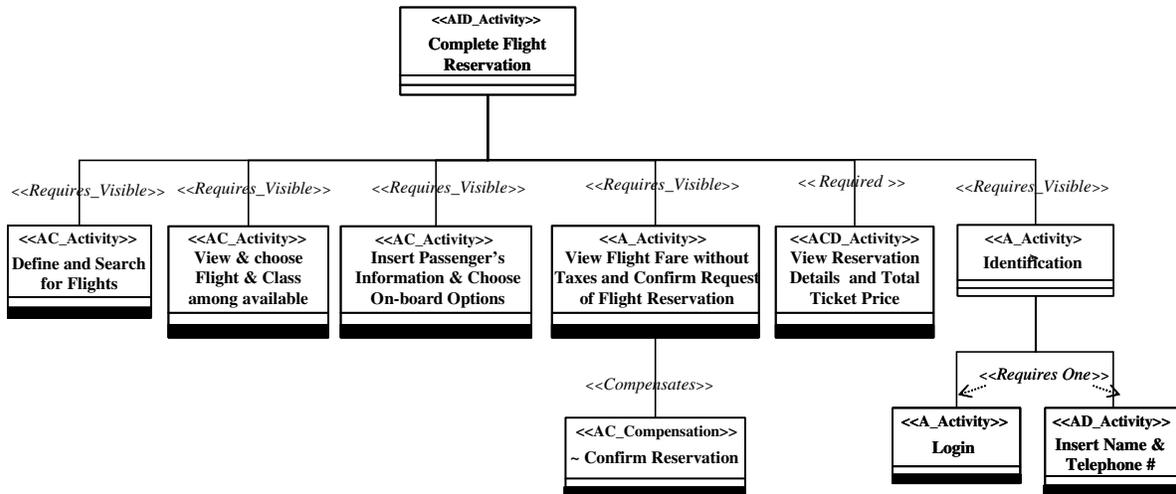


Figure 1. The extended version of the UWA Organization Model applied to the “Complete Flight Reservation” Transaction on the Alitalia.it [22] Web site.

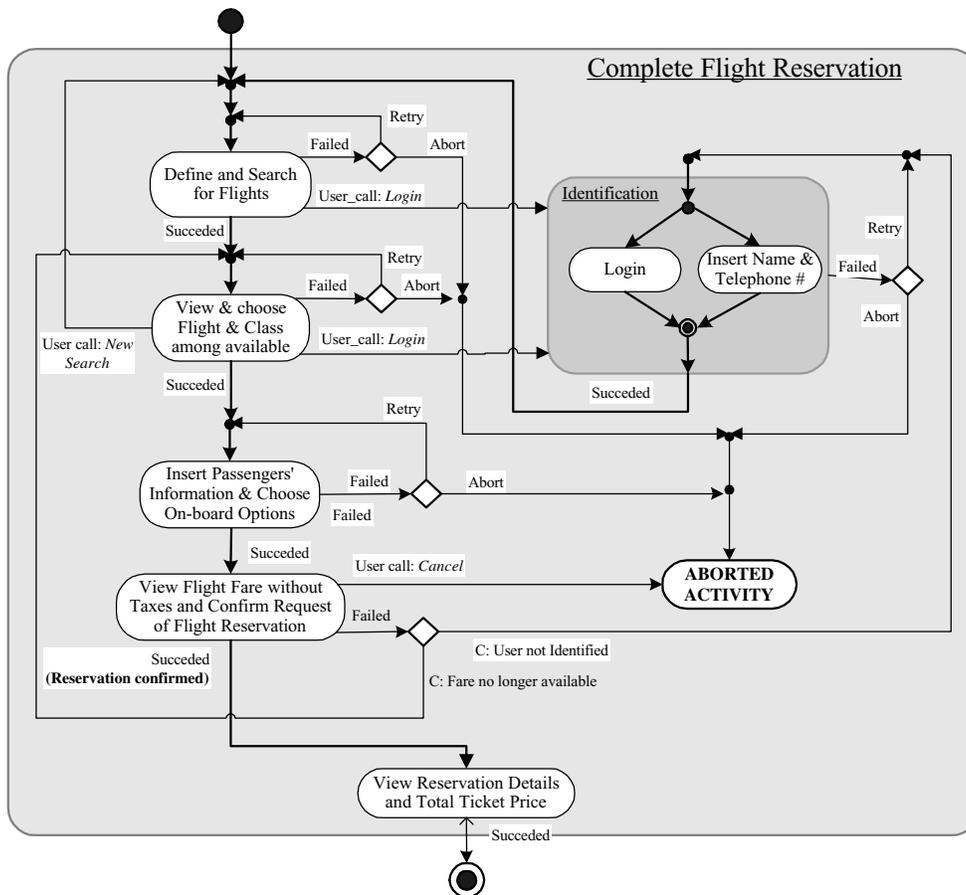


Figure 2. The extended version of the UWA Execution Model applied to the “Complete Flight Reservation” transaction on the Alitalia.it [22] Web site.

4.2 Designing the Interplay between Activity Execution and Content Navigation

As briefly mentioned in section 3, the UWA Navigation Design activity is intended to define the elementary granules of information provided to the user and the rules governing accessibility among them. It is based on the concepts of *Navigation Node* (Node for short) and *Navigation Cluster* (Cluster for short).

Nodes and Clusters are derived from the UWA Information Model and are associated with the concepts of Entity Types, Entity Components, Access Structures, etc. included therein.

For our purposes, we introduced a new type of Node, the *Activity Node*, and a new type of Cluster, the *Activity Cluster* [16].

Activity Node

We define the Activity Node as a new type of Navigation Node used by the designer to define which information is provided to/requested from the user when executing an Activity and the interaction elements, e.g. buttons and links, that will be used to invoke actions and execute the Activity. An Activity Node can be associated with each elementary Activity of a given Web Transaction and once defined it can be handled from a navigation point of view like any other Navigation Node.

Activity Cluster

We define the Activity Cluster as a new type of Navigation Cluster used by the designer to define the way Activity execution and content navigation interact with each other and the Transaction execution state. An Activity Cluster is associated with a Web Transaction Activity and includes:

- at least one Activity Node, associated with same Activity as the Cluster.
- Navigation Links (between standard Navigation Nodes) and Transition Links (inward or outward Activity Nodes), whose associated labels describe the alterations produced on the state of the Transaction by both navigation and Activity execution and the navigation associated with the execution of an Activity or a user action.

The concepts of Activity Node and Activity Cluster enable the designer to define the interplay between transaction execution and content navigation, and the relationships between the application's informative and transactional aspects.

Figure 3 shows an example of an Activity Cluster associated with hypothetical Activity A, including Navigation and Activity Nodes and Transition and Navigation links among them.

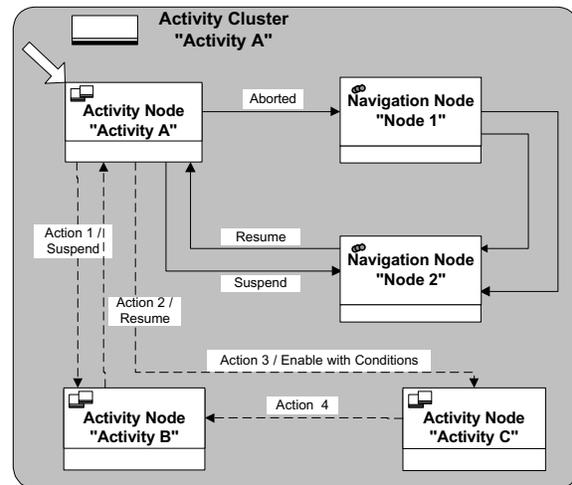


Figure 3. An example of an Activity Cluster associated with hypothetical Activity "A".

5. Summary and Future Work

The addition of business processes to Web applications gives rise to challenges and risks that require a suitable design methodology to be used in order to create Web-based hypermedia systems with improved usability, free of errors and undesired behaviors. The design of Web Transactions that implement business processes requires great care to be taken of the information and navigation issues, due to the intrinsic navigational nature of the Web. This doesn't mean that emulating Web Transactions as a sequence of navigational steps is a good solution. On the contrary, as also discussed in the literature, this approach, as well as treating Web Transactions as a "by-product" of the development process, results in error-prone and low quality applications.

Another aspect to be considered in designing Web Transactions is how far their structure and properties meet user expectations and so improve the user's experiences of the Web site.

In this paper we proposed an extension of the UWA Transaction Design meta-model, which specifically addresses the design of Web application Transactions. Starting with the analysis of methods for business process design in Web applications proposed in the literature, and considering the experience we gained from using the UWA design framework in several "real-world" case studies, we defined a representative list of requirements that a desirable Web Transaction design meta-model should verify. The list of requirements is the basis for evaluating and comparing different Web Transaction design approaches. We used this list to identify the strengths

and shortcomings of the UWA Transaction Model. The revisions and extensions we proposed for the UWA Transaction Model try to overcome the identified shortcomings and improve the model's effectiveness, usefulness and ease of adoption. The resulting new version of the UWA Transaction Model enables the designer to clearly represent the way Web Transaction execution and content navigation interact with each other, to define which information objects are involved and support the execution of a given user activity, and to capture and design a Web Transaction from a user's point of view. The resulting meta-model is better integrated with the remainder of the design process since the results of both Information and Transaction Design are merged in the extended Navigation Model.

We have already applied the revised version of the meta-model to a number of case studies that support its validity. Future work concerns the extension of the design tools already developed in the UWA framework to support the extended Navigation and Transaction Model. There also needs for a systematic assessment of the usability of the designed Web Transactions by means of techniques available in the literature.

References

- Rossi, G., Schmid, H., Lyardet, F.: Engineering Business Processes in Web Applications: Modeling and Navigation Issues. Proceedings of the 3rd International Workshop on Web Oriented Software Technology (IWWOST03) Oviedo, Spain, 2003.
- Schmid, H., Rossi, G.: Modeling and Designing Processes in E-Commerce Applications. IEEE Internet Computing, January/February 2004.
- De Troyer, O., Casteleyn, S.: Modeling Complex Process for Web Application with WSDM. 3rd International Workshop on Web Oriented Software Technology (IWWOST03), WISE Research group, Oviedo, Spain, 2003.
- Koch, N., Kraus, A., Cachero, C., Meliá, S.: Modeling Web Business Processes with OO-H and UWE. 3rd International Workshop on Web Oriented Software Technology (IWWOST03), Oviedo, Spain, 2003.
- Atzeni, P., Parente, A., Specification of Web Application with ADM-2. Araneus Project, Università Roma Tre, 1.er Workshop Internacionale de Tecnologia Software Orientada a Ambiente Web, Valencia, Spain, 2001.
- L. Baresi, F. Garzotto, P. Paolini, and S. Valenti: HDM2000: The HDM Hypertext Design Model Revisited. Technical Report, Politecnico di Milano, Italy, Jan. 2000.
- Baresi, L., Garzotto, F., Paolini, P., Extending UML for Modeling Web Applications. In Proceedings of 34th Annual Hawaii International Conference on System Sciences (HICSS-34). IEEE Computer Society, 2001.
- UWA Consortium, Ubiquitous Web Applications, Proceedings of e2002 eBusiness and eWork Conference, Prague (Czech Republic), October 2002.
- Gioldasis, N., Christodoulakis, S.: UTML: Unified Transaction Modeling Language, The Third International Conference on Web Information Systems Engineering (WISE'02), Singapore, December 12 - 14, 2002.
- Baresi, L., Denaro, G., Mainetti, L., Paolini, P.: Assertions to Better Specify the Amazon Bug. 14th ACM International Conference on Software Engineering and Knowledge Engineering (SEKE 2002), Ischia, Italy, 2002.
- Object Management Group (OMG). Unified Language Modeling Specification (Version 2.0). Online at www.omg.org, 2004.
- Distante, D., Parveen, T., and Tilley, S.: Towards a Technique for Reverse Engineering Web Transactions from a User's Perspective. In *Proceedings of the 12th IEEE International Workshop on Program Comprehension (IWPC 2004)*: June 24-26, 2004; Bari, Italy). Los Alamitos, CA: IEEE CS Press, 2004.
- Distante, D., Tilley, S., Huang, S.: Web Site Evolution via Process Restructuring from a User's Perspective. The sixth IEEE International Workshop on Web Site Evolution (WSE 2004), Chicago, IL, Sept. 11-17, 2004.
- UWA Consortium: Deliverable D2: General Definition of the UWA Framework. Online at www.uwaproject.org (2001).
- UWA Consortium: Deliverable D6: Requirements Elicitation: Model, Notation and Tool Architecture. Online at www.uwaproject.org, 2001.
- UWA Consortium: Deliverable D7: Hypermedia and Operation design: model and tool architecture. Online at www.uwaproject.org, 2001.
- UWA Consortium: Deliverable D8: Transaction design., Online at www.uwaproject.org 2001.
- UWA Consortium: Deliverable D9: Customization Design Model, Notation and Tool Architecture. Online at www.uwaproject.org, 2001.
- UWA Consortium: Tool supporting the UWA design process (Rational Rose Add in). Downloadable at www.uwaproject.org, 2001.
- Fischetto, G., Bochicchio, M., Distante, D., La Progettazione delle Transazioni nelle Applicazioni Web: Aspetti teorici ed applicativi", Master thesis (in Italian) in Computer Science Engineering, University of Lecce, Italy, February 2004.
- Chrysanthis, P., Ramamritham, K.: Synthesis of Extended Transaction Models Using ACTA". ACM Transactions on Database Systems Vol. 19, No. 3 (Sept. 1994), pp. 450-491.
- Alitalia.it, Online at www.alitalia.it (period of observation Oct. '03 - Dic '03).
- Elmasri, R., Navathe, S.: Fundamentals of Data Base Systems (4th Edition). Benjamin-Cummings, 1998.