Exploring dance notations systems for the implementation of a domain specification language for the salsa dance

Ana Rita Dauane
dnxana001@myuct.ac.za
Department of Computer Science
University of Cape Town
Cape Town, South Africa

ABSTRACT

In this literature review, we explore the different dance notations and the implementation of domain-specific languages. We examine the dances notations, the notation editor, and the use and implementation of the DSLs.

Throughout the research, we concluded that in literature had done no work to create a DSL tool to annotate dance steps. Additionally, research is needed for the creation of modelling language for a graphical representation of dance.

CCS CONCEPTS

• **DSL** → Modelling, Language Specification, Graphical Notation, Abstract Syntax, Concrete Syntax

KEYWORDS

Dance, Salsa dance, Step Annotation, Domain-Specific Languages, Conceptual Models

1. INTRODUCTION

Teaching through oral communication is the most used and convenient way to learn how to dance. However, it is very likely to pick up those steps at first while learning how to dance physically with verbal communication. There is no formal writing language for dance even though several languages describe the notation of steps available [1]. The majority of the dancers, dance students, teachers and the choreographs do not have the capabilities to encode a dance and convey it because of the complexity of the notation systems. Currently, students and teachers create and edit the steps annotation through photoshoot by copying and pasting, colouring and adding arrows to the steps annotations. At present, it is difficult and time-consuming to edit and draw all the notations carefully. Moreover, recording dance on paper is a time-consuming and challenging process [4]. Over the years, the dance education field has evolved technologically. Many tools have been developed to support dance teaching and learning through web-based applications, mobile applications, motion capture, and other things. Although, there is a need for a modelling language for dance where the user can create and edit the step annotation. As a result, many dancers still battle with the creation and editing step annotation because it is time-consuming and is an enormous amount of work.

According to the literature available, there is still a lack of research based on the dance steps, much less on the salsa dance. There is a need for a dance tool that creates and edits steps for a Latin American style in these circumstances.

The project aims to create a learning dance tool that provides videos, music and step annotations of the salsa dance. More precisely, the applications will allow users to create and edit steps notations through graphic annotations and save them for learning purposes.

This literature review aims to understand the current notation systems used to convey dance movements symbolically relating with the applications of domain specifications languages. By reviewing these areas, we seek to understand what previous research says about the current dance notations systems, the existence of other approaches for dance language. And how this relates with the complexity of the notations systems to how novice students and dance teachers write dance salsa dance notation for learning purposes and what remains to be studied. This paper will discuss two main topics that deem to be of particular interest to our area: dance notations systems and DSLs.

2. DANCE NOTATIONS

Many notation systems that represent the movement of the human body can be used to represent the dance notations. Dance Notations are used to describe the movements of the dance using various symbols such as dots, feet (step feet), shoes or heels [22] (i.e., the structure of tango notation), among others. In this section, we will focus on different dance notation systems.

2.1. Labanotation

Labanotation is one of the most popular dance notations. It has many applications in different areas such as choreography preservation, dance education, movement analysis and dance documentation [1].

The symbols are mapped on a vertical stave, which reads from bottom to top and left to right. Each symbol contains four elements: shape, level, time and the body part that is involved. The shape indicates the direction of the movement, the level of the movement represents the shading, and the time indicates the length of the symbol [9]. The shapes of the symbols indicate 9 nine directions in space and each and any symbol can be used to represent one of the other steps or gestures [9]. However, creating a labanotation score is a difficult task. It requires time, patience to observe the dance movements and draw the notations precisely and entails a good understanding of how to identify the body movements and translate them into Labanotation scores. As time has gone by, some digital tools have been created to facilitate manual work.

Laban Writer [10], LED & LINTEL [11] and LabanEditor [12] are computer tools that can be used to draw Labanotation scores and save them into digital form. Nonetheless, it is uncommon to find people that are experts in using these tools. More tools are human-readable because it represents the semantics of Labanotation using open text formats, such as LabanXML [16], MovementXML [21] and Dance-OWL [5].

Although Labanotation is a great tool for movement representations, it is not appropriate for implementing a dance step notation tool due to its complexity and the majority of the dancers do not hold the knowledge.

2.2. Benesh Movement Notation

Benesh Movement Notation is similar to musical notation. The Benesh Notation represents the movements human body divided in top of the head, shoulder, waist, knees, and feet placed in a 5-line stave [13]. The movements of the BNF are represented from the back of the performer. The movement and orientation are marked down below the stave and the timing is marked down above the stave. Currently, there are a variety of BMN tools for dance notation. MacBenesh and Benesh Notation Editor are examples of graphic editors that help produce BNF scores [14]. Another tool, a web-based application, the Web-based Movement Library (WML), gives an insight to the novice dancers on how the dance experts symbolize various parts of the recording and dance movements [18]. WML allows the user to search the dance documentation by the dance genre, by annotating the recordings, and by using keywords.

2.3. The Salsa Language

As a form of documenting dance, Boschetti and Lyons [8] introduced the concept of ‘salsa lines’ and ‘salsa elements’ for the salsa language. The five salsa lines is a system composed by a hand hold line, direction line, man (or leader line), common action line, lady (or follower) line and the salsa elements has 4 elements in plain English hand holds, directions, positions and actions which are part of the ‘Salsa Dictionary’. The time progression is illustrating in the salsa lines by diving four count bars by a vertical line.

Salsa lines are used to write down figures for salsa on 1 and salsa on 2 through [8]. The hand line indicates the type of hand hold during each bar, the direction indicates where the dancers are facing are, the man line indicates actions performed by the man independently from common actions performed by both dancers together and indicates the position for the man, common action line indicates actions performed together by body partners and the lady line indicates the action by the follower (or lady) following the leader (or man) [8].

The hands hold is distinguished between normal hand hold and crossed hand hold. The direction is defined by two variables: a partner and a line of dance. Each dancer can face the partner, be in the same direction as the partner or be in the opposite direction from the partner [8]. Additionally, every dancer can stand facing the line of dance or facing perpendicular to the line of dance. To indicate where the dancer is facing using arrows in junction with the letter ‘M’ for man and ‘L’ for lady. We will use arrows in combination with the letters 'M' (for Man) and 'L' (for Lady) to indicate where each dancer is facing. [8] defined actions as the vital elements of salsa and that they can be initiated by the leader or the follower. Generally, the leader leads the follower or move independently, and the follower follows the leader but can also initiate independent actions for such as free turn. The actions can be described as change of hands hold, change of directions, cross body leads, turns, ducking without turning, checks, hand drops and throws, natural top and walk. Positions happen after performing some actions. They can be described as hammealocks, half hammerlock, embrace, arm lock, arm loop and sombrero.

The salsa language is textual and can be useful when implementing the application for the definition of the technical terms and icons specific for the salsa dance.

2.4. Mathematics of Salsa

An as an attempt to define a mathematical language for the salsa dance position, Renesse and Ecke [3] formalized a mathematical Space of dance moves. Their notation is characterized by a quintuple as follows (leader orientation, follower orientation, leader's arm, follower's arm, the times the arms are crossed). [3] in this particular study only considered the fixed positions of the body and arms. The steps of the salsa dancing were not considered. The leader orientation and the follower can be either 0 or 1. When the partner is facing the other partner is 1 otherwise is 0. The follower and leader's arms represent the arms positions that can be either RB for 'back' or 'belly' or RH for 'head’. An asterisk is used to denote when the arm is behind the body. Ultimately, the times the arms are symbolised by a 'C'. When they are no crossing 0 is written. For the leader's position, to be represented if the arm is on top or if the arm is at the bottom. For example, we would write the C with a superscript R and subscript L, symbolising the right arm on top and the left arm underneath. The moves represented in this study were when the dancers are facing each other and join the same hands, when facing each other and join opposite hands or arms behind the leader's back. Some issues arose during the research, such as not representing the arms over the heads. The symmetry notation introduced in the language is complex and difficult to draw using this technique [3]. Also, it can be seen that the diagrams could have a different characteristic arm, such as thicker arms for the follower's arm or even a different colour because of the complexity of solving the graph. This research paper showed that the dance notation could be represented by the arms notation too through the salsa dance, which gives more room to study new techniques of the dance notations. The project application tool is an example of a new possibility of representing the steps notation of the salsa dance.

3. NOTATION EDITORS

3.1. Semantic Representations using Labanotation

3.1.1. LabanXML

Due to a more significant number of graphical editors for Labanotation and the need for a text representation to interchange Labanotation via the internet, Nakamura and Hachimura [16] found a need to have a dance editor that allows the user to input, edit, and search dance moves. The authors created a semantic representation of Labanotation in XML (eXtensible Markup).

The stave of the Labanotation is vertical and differs from the music notation stave, which is horizontal. [16] followed the same logic LabanXML has the same approach to MusicXML [17], a semantic representation of the music notation. The logical tree structure of elements by grouping. The core element is <laban> which includes the <attributes> and <notation> elements. The <attribute> element includes <time> element. And the <time> element includes <beat> and <beat-type> elements. Each <notation> element includes the <repeat> element and <measure> element. The columns are classified into <left>, <support> which is separation of support column and <right> elements [16]. The occurrence operator is represented by an asterisk, plus and the element name with any special character. The plus means that the element will occur one or more, zero means zero or more and no special character means occur exactly once [16]. the occurrences resemble the concept of Kleene closure and Kleene plus, for the zero and plus and some rule notations for grammars. The implementation can be found in LabanEditor2 [16].

3.1.2. MovementXML

MovementXML is an extension of the LabanXML. MovementXML is a straightforward translation of the Labanotation scores to XML and provides simple movements notation and interpretation of the dance [20].

3.2. Graphical Dance Notation Editors

As mentioned above, the available graphical editors for the Benesh notation are the MacBenesh and Benesh Notation Editor. With their system, Singh et al. [14] were able to offer a way to revise and produce dances based on the BMN. The Benesh Movement Notation Editor [14] was created to improve the user experience while creating, editing and modifying movements. The editor had two significant issues with the user interface. The need for an iterative language to allow the user to communicate with and issue commands to the system is one of the major corners. Also, the need for an interface to display the state of the system and the numerous options that a user could access was an issue [25]. In broad terms, a dance score can be referred to as a piece of dance. More precisely defined, it is the notation used for dance movements [14]. MacBenesh is a tool created by the Macintosh that allows the user to create a single dance and save its BMN scores. The Benesh Editor is a Windows tool for writing Benesh Movement Notation.

LabanEditor is a graphical editor for dance notation that is used to write and edit Labanotation Scores. The application allows the user to compute and edit dance movements and display the human body in 3D animation [15].

4. DOMAIN-SPECIFICATION LANGUAGES

Domain-Specific are languages that are designed for a specific application domain [6]. The DSLs are more expressive than general-purpose languages (GPL) by only focusing on creating programs of a specific area. They are used for many purposes, different contexts, different users, for an expert user or a novice user [7]. A good example would be companies with their language, a specifical design for them, that anyone who works there can use the language. There are many different applications of the DSL, some of them we even use on our daily basis without even knowing, such as the Unified Modelling Language (UML), the Entity Relationship Diagram (ERD), SQL database, Excel Spreadsheet and the HTML web layouts.

They are three types of DSL, domain-specific markup languages, domain-specific modelling languages and domain-specific programming languages. In this particular paper, we will focus on the domain-specific modelling languages.

4.1. Why Domain-Specific Languages?

According to Van Deursen et al. [19], a well-designed DSL has its risks and benefits. This study aimed to discuss the terminology as well as the risks and benefits of DSLs relating to the general purposes languages. [19] listed the benefits of enhancing productivity, allowing the solutions to be expressed at the level of abstraction of the problem domain and the disadvantages as cost of education for DSL users, implementation, and difficulty finding an appropriate scope of the DSL. They also state that DSL is end-user programming. In other words, it allows an end-user to perform simple programming tasks of a specific domain. A good example is spreadsheet programming using the Excel program. DSLs have many application domains, namely Software Engineering, Systems Software, Multimedia, Telecommunication and Miscellaneous [19].

4.2. Domain-Specific Modelling Languages

Domain-Specific Modelling aims to increase the level of further abstraction programming, similar to the problem domain. As explained by Frank [21], domain-specific modelling languages because provide model integrity which consists of constraints included in the language that, if not, would have to be added manually. The user does not need to have rebuilt their technical terms on their own, which promotes the convenience and productivity of modelling. Generally, DSML is represented using a graphical notation, a concrete syntax that encourages comprehension and clarity.

A DSML is defined by its abstract syntax, concrete syntax and semantics. The abstract syntax defines the concepts of the area of expertise. The concrete syntax describes the representation of the different constructs, either visual (graphically using icons) or

textual. Semantics define models' meaning, which incorporates "what is happening" and how to give meaning. [21] proposed a process that is intended to guide the development of a DSML for the domain of enterprise modelling. He suggests a sequence of steps for the DSML development, namely Clarification of Scope and Purpose, Analysis of Generic Requirements, Analysis of Specific Requirements, Language Specification, Design of Graphical Notation and Evaluation and Refinement.

4.2.1 Clarification of Scope and Purpose Phase
The purpose and the scope need to be clarified in the development of the DSML. This phase should describe essential design objectives and a definition of the budget, with an expectation of higher productivity and quality. The product specifications may include explicit modelling tasks of the past or other tasks, such as programming, which could be replaced by modelling tasks in a beneficial way. In addition, it is important to consider the earlier projects designed at these modelling tasks looking at what impact a DSML would have had on performance and outcome [21].

4.2.2 Analysis of Generic Requirements Phase
During the design and conception of the DSMLs, we should consider the generic requirement. The generic requirements are applied to all the DSMLs but with different weights and also, they may need to be adapted to a specific DSML. Each requirement should be described and justified with respect to the purpose of the DSML, and also each requirement should be characterized with respect to its relevance [21].

4.2.3 Analysis of Specific Requirements Phase
All the stakeholders (users, language designers) need to be to have clear idea need to be supported with developing a clear idea of what they may expect from the DSML [21]. The specific requirements are based on using scenarios that are developed, taking into account previous and future tasks with the potential use of the DSML, which is illustrated during the design of the preliminary diagrams. Specific requirements act as an intermediary for further polishing the use scenarios. The use of special requirements is essential for describing the scenarios in a certain structure.

4.2.4 Language Specification Phase
The language specification phase is the most crucial phase when designing a DSML because of the specification of the abstract syntax and semantic of the language.[21] states that it will usually include various design decisions, some of which are common in conceptual modelling while others are specific for the design of metamodels. The specification of a DSML aims to remodel the concepts corresponding to a designated domain. Because it makes sense to first develop a glossary with the key terms. At first, the glossary is referred to as basic because it contains terms with corresponding descriptions [21]. The main function of the basic glossary is to collect terms to be used in the representation of the graphical notation, which is beneficial for the comprehension, usability and productivity of a DSML. The basic glossary serves as a collection of terms that are used in the targeted domain of discourse. The graphical notation is of considerable relevance for the comprehensibility, usability and productivity of a DSML.

4.2.5 Design of Graphical Notation Phase
As mentioned above, graphical notations are relevant for the usability, comprehensibility and productivity of DSML. However, Frank [21] states that usually, they are no experts in the design of a graphical notation because some might not be interested in language design. Additionally, the ones that trained for the design of icon symbols will lack knowledge on how to use the DSML. Although there is no theoretical background, [21] provides some guidelines with theoretical considerations. The author [21] based the following guidelines on his own experience and some literature.

4.2.6 Evaluation and Refinement Phase
The evaluation needs to account for specific challenges to ensure a certain quality level to create a comprehensible and systematic evaluation and, if required, a revision of the tool.

4.3. Comparison of DSMLs and Dance Notation Systems

Considering the dance notation systems and the DSLs analysed are the opposite from each other. The similarity DSMLs and the dance notation systems on this representation that both can be represented graphically.
In addition to that, the DSML is a language that is represented at the possible abstract level and sets constraints based on the applied domain. For that reason, they increase productivity and comprehensibility [21]. They can be represented graphically or textually. An expert user can use the DSMLs with their domain knowledge without requiring technical skills, such as programming and others. Additionally, a novice user can have more facility in learning and understanding the language.
Whereas the dance notation system, specifically labanotation and the BMN, is complex and only experts can write those dance notations. The dance notations are challenging to translate, and a piece of knowledge is necessary to use the notation because of its language. Besides that, the students and teachers currently document dance through videos and use some tools to edit the steps annotations. Writing paper-based notations and photoshoot notations is a waste of time and a source of annoyance.
Overall, the DSMLs is better to notate a language because of their simplicity, whereas the dance notations use complex symbols and are challenging to learn.

5. DISCUSSION

Although the notations system aids in defining a language for dances, the ones mentioned in Section 3 might be complex to document for learning purposes. Consequently, dance experts and students who do not have knowledge in the area recur to another different way of writing dance for learning purposes such as paper-annotation, video and editing existing steps or

creating new steps annotations pictures, which is time-consuming.
Despite that, the Labanotation and Benesh Notation have graphical editors, allowing the user to create the dance annotation. It still is too complex, considering that the dance experts on those notations are rare. The use of the LabanEditor or Benesh Notation Editor is not convenient for implementing the project application.
This literature also discusses the use of the DSL, focusing more on the DSML. The DSL is a specific-purpose language that allows its representation to be abstract. The use of the DSMLs is around abstract and concrete syntax, which can be represented visually in a meaningful manner. The meaning of the language is essential when building a DSLM because of the concepts and terms for representing the language.
Its implementation is appropriate for the step annotation because it is end-user-centered, focusing on the simplicity of the icons and elements relating to their domain knowledge, such as salsa dance.

6. CONCLUSION

This literature review is centralized around two topics: dance notations and domain-specific languages.
It focused on understanding the different types of dance notation and the available graphical dance editor tools. There are no existing domain-specific languages representing salsa dance, and only a few dance notations can describe the salsa dance mathematically and textually.
In conclusion, no similar tools are available for a graphical representation of the dance steps, more specifically for the salsa dance. For future work purposes, we will be investigating the different ways to implement a DSL and exploring the other graphical notations considering the requirements and guidelines of a DSL.

REFERENCES

[1] Katerina El Raheb and Yannis Ioannidis. 2014. From Dance Notation to Conceptual Models: A Multilayer Approach. In Proceedings of the 2014 International Workshop on Movement and Computing (MOCO '14). ACM, New York, NY, USA, Pages 25, 6 pages. DOI: https://doi.org/10.1145/2617995.2618000

[2] M. Li, Z. Miao and C. Ma, "Dance Movement Learning for Labanotation Generation Based on Motion-Captured Data," in IEEE Access, vol. 7, pp. 161561-161572d
DOI: 10.1109/ACCESS.2019.2951588.

[3] RENESSE, C.V. AND ECKE, V. 2011. Mathematics and Salsa dancing. Journal of Mathematics and the Arts 5, 1, 17–28.

[4] Mila,Parrish. 2007. Technology in dance education. In L. Bresler (Ed.), International handbook of research in arts education. Springer, 1381-1397

[5] K. El Raheb and Y. Ioannidis. 2011. A labanotation-based ontology for representing dance movement. Springer ,106–117.

[6] M. Fowler and R. Parsons. 2010. Domain-Specific Languages. Addison-Wesley.

[7] Strumenta. What are Domain-Specific Languages?. Retrieved May 25, 2021 from https://tomassetti.me/domain-specific-languages/

[8] SalsaIsGood. 2001. A Dictionary for Salsa and Mambo moves. Retrieved May 27, 2021 from https://www.salsaisgood.com/dictionary/Salsa_language.htm

[9] DBN. READ A GOOD DANCE LATELY? . Retrieved May25,2021
http://dancenotation.org/lnbasics/frame0.html

[10] L. Wilke, T. Calvert., R. Ryman, and I. Fox. 2005. From dance notation to human animation: The LabanDancer project: Motion Capture and Retrieval. Comput. Animat. Virtual Worlds 16, 3-4 (July 2005),201-211.
DOI: http://dx.doi.org/10.1002/cav.v16:3/4

[11] F. Hunt, G. Politis, and D. Herbison-Evans. 2010. Led & lintel: A windows mini-editor and interpreter for labanotation. (2010). Retrieved 1 May, 2020. Retrievedfrom
http://satd.com.au/don/pubs/led.html

[12] K. Kojima, K. Hachimura, and M. Nakamura. 2002. LabanEditor: Graphical editor
for dance notation. Proceedings of the 11th IEEE International Workshop on Robot and Human Interactive Communication. 59–64.
DOI: https://doi.org/10.1109/ROMAN.2002.1045598

[13] Benesh International. 2001. The Benesh Movement Notation score. Retrieved May 29, 2021 from
https://www.royalacademyofdance.org/documents/benesh-docs/BMNscore.pdf

[14] Singh, B., Beatty, J.C., And Ryman, R. 1983. A graphics editor for benesh movement notation. Proceedings of the 10th annual conference on Computer graphics and interactive techniques - SIGGRAPH 83.

[15] K. Kojima, K. Hachimura,M., Nakamura. 2002. LabanEditor:Graphical Editor for Dance Notation. Proceedings of IEEE 2002Int, Workshop on Robot and Human Interactive Communication, 9–64

[16] Nakamura, M., Hachimura, K. 2006. An XML Representation Of Labanotation, LabanXMl, And its implementation On The Notation Editor Labaneditor2. Преглед НЦД

[17] Good, M., Actor, G. Using MusicXML for File Interchange. Proceedings of the third International Conference Web Delivering of Music 15-17 (September 2003), 153.

[18] K. E. Raheb, A. Kasomoulis, A. Katifori, M. Rezkalla, and Y. Ioannidis. 2018. A Web-based system for

annotation of dance multimodal recordings by dance practitioners and experts. In Proceedings of the 5th International Conference on Movement and Computing (MOCO '18). ACM, New York, NY, USA, Article 8, 8 pages.
DOI: https://doi.org/10.1145/3212721.3212722

[19] Van Deursen, A., Klint, P., Visser, J. 2000. Domain-specific languages: an annotated bibliography. In ACM Sigplan Notices.
DOI: https://doi.org/10.1145/352029.352035

[20] Hatol, J. 2006. MOVEMENTXML: A representation of semantics of human movement based on Labanotation. (Doctoral dissertation, School of Interactive Arts and Technology-Simon Fraser University).

[21] U, Frank. 2013 . Domain-specific modeling languages: Requirements analysis and design guidelines. In Domain Engineering. Springer, 133–157.

[22] S. Rahmatian. 2018. The structure of Argentine tango. Journal of
Mathematics and the Arts, 12:4, 225-243.
DOI: 10.1080/17513472.2018.1474331