An Ontology Design Pattern for Chess Games

Adila Krisnadhi[1,2], Víctor Rodríguez-Doncel[3], Pascal Hitzler[1], Michelle Cheatham[1], Nazifa Karima[1], Reihaneh Amini[1], and Ashley Coleman[1]

[1] Data Semantics Lab, Wright State University
[2] Faculty of Computer Science, Universitas Indonesia
[3] Ontology Engineering Group, Universidad Politécnica de Madrid

Abstract. We present an ontology pattern describing records of chess games. Besides being an interesting modeling problem by itself, the fact that chess is one of the most popular game in the world with hundreds of millions of active players, including several millions online players led to a huge amount of chess game data available from various online chess databases. Furthermore, these data are becoming available as linked data with links to prominent datasets, such as DBPedia or GeoNames. However, there is still a lack of a well-designed schema that can ease linked data publishing of chess games. Our pattern is hoped to fill that gap.

1 Introduction

Chess is one of the most popular game played by people worldwide. A 2012 study found that "605 million adults play chess regularly - a number comparable to regular users of Facebook." [1]. Among them, millions of players actively play chess online, e.g., Chess.com boasts more than 12 million members with tens of thousands players play concurrently online [2]. More interestingly, the large number of chess games played online led to huge repositories of chess games that people can access in the form of Portable Game Notation (PGN) files. Chess game data does not just contain details about chess moves, but also about player identities, chess tournament names, and spatiotemporal information relevant to the games. Making these data available also as *linked data* is an interesting proposal as we can generate even richer information by combining chess game data with data from prominent linked datasets, such as DBPedia, GeoNames, or Library of Congress. We have described an effort to realize this proposal elsewhere [4]. What is lacking, however, is a well-designed schema for these data that is still sufficiently flexible and extendible in the long run. Thus, to serve this purpose, we present an ontology design pattern for chess games, engineered by following a modular approach.

To guide the design of the pattern, competency questions were identified, e.g.: (i) "Who played against Kasparov in the round 1994 Linares tournament? Did (s)he play as a white or black player?"; (ii) "What is the first move taken by black player in the Sicilian Defence opening?"; (iii) "List all moves in a Fool's Mate game where black wins after 2 moves by both players?"; and (iv) "What did Kasparov say about his opponent first two moves in his commentary about his game against Topalov in 1999 Tournament in Wijk aan Zee?"

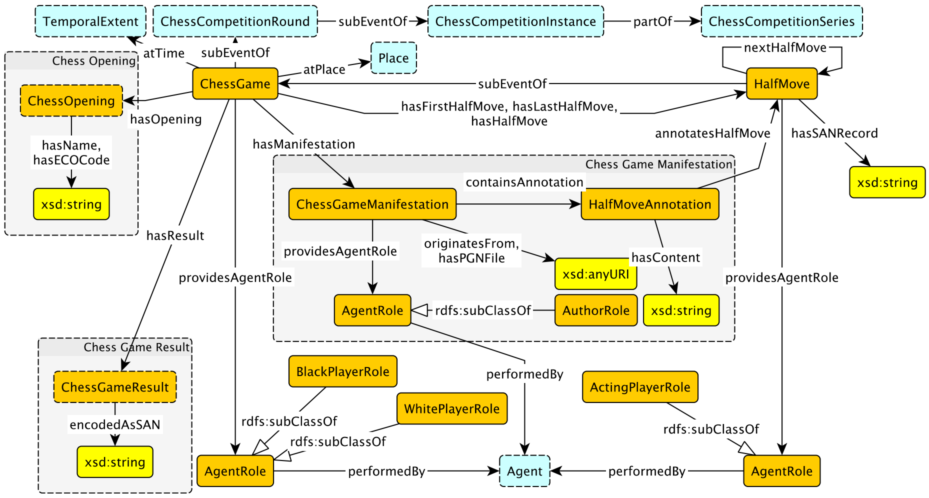

Fig. 1. The core of Chess pattern modeling chess games. Though Event is omitted, half moves, chess games, and chess competition rounds/instances/series are events. Orange boxes are atomic classes. Blue boxes are classes with details hidden in a separate pattern/ontology. Grouping boxes are sub-patterns that can be modeled separately.

2 The Pattern

The Chess pattern[1] (Figure 1) consists of a core part that models chess games as a series of half-moves performed by chess players, and a few other additional parts, modeling the game annotation, opening, result, and the structure of chess tournament as abstracted from FIDE regulations[2] and standard practice in chess community. We use the concise DL notation [3] to express the axioms below.

Chess Game. The ChessGame class represents chess games, modeled as a kind of event. That is, a chess game happens at some place and time – (1) & (2), and involves actors performing some role, e.g., white and black players. The TemporalExtent and Place classes are hooks for attaching spatial and temporal information to a chess game, whose model is out of scope of this paper. Actors in a chess game are agents who perform some role provided by the game. Instead of employing binary relationships using properties such as "hasWhitePlayer", we employ reification via the AgentRole class. From this complication, we gain flexibility and the ease of adding contextual information when necessary. Every chess game provides at least two distinct role instances: for the white and the black players – by (2) and disjointness between WhitePlayerRole and BlackPlayerRole explained at the end of this section. Each agent role is performed by exactly an agent – by (3). Agents can be a person, a team, a computer software, etc. Note that the axioms still allow one agent to perform two roles simultaneously. Also, we allow a a chess game played by multiple white players and/or multiple black

[1] OWL file available at `https://db.tt/oBAg1wlz`
[2] `http://www.fide.com/fide/handbook.html`

players. On the other hand, instances of WhitePlayerRole and BlackPlayerRole are roles specific for a particular ChessGame (axiom (4)). So, we assert the following where pAR is a shorthand for providesAgentRole:

$$\mathsf{Event} \sqsubseteq \exists \mathsf{atPlace.Place} \sqcap \exists \mathsf{atTime.TemporalExtent} \tag{1}$$

$$\mathsf{ChessGame} \sqsubseteq \mathsf{Event} \sqcap \exists \mathsf{pAR.BlackPlayerRole} \sqcap \exists \mathsf{pAR.WhitePlayerRole} \tag{2}$$

$$\mathsf{AgentRole} \sqsubseteq (=1\ \mathsf{performedByAgent}) \tag{3}$$

$$\mathsf{WhitePlayerRole} \sqcup \mathsf{BlackPlayerRole} \sqsubseteq \mathsf{AgentRole} \sqcap (=1\ \mathsf{pAR}^{-}.\mathsf{ChessGame}) \tag{4}$$

We model the standard chess rule where each player takes turn performing a *half move* by moving one of his pieces. A finite sequence of half-moves constitutes a chess game, which has exactly one first half-move and one last half-move within that sequence – (5). The axioms, however, do not enforce that a chess game has to start from the initial chessboard configuration, nor they assert that the white player is the one who moves first. This allows the pattern also to model description of chess problems and their solutions. Half-moves are represented by the HalfMove class, itself a kind of event that is a subevent of exactly one chess game, and provides a role for the corresponding acting player – by (6) and (7). A half-move can only be followed by exactly one other half-move and cannot follow itself – (8). Also, a half-move may have an associated record in the Standard Algebraic Notation (SAN), which is only assumed to be a string value here.

$$\mathsf{ChessGame} \sqsubseteq (=1\ \mathsf{hasFirstHalfMove.HalfMove}) \sqcap (=1\ \mathsf{hasLastHalfMove.HalfMove}) \tag{5}$$

$$\mathsf{HalfMove} \sqsubseteq \mathsf{Event} \sqcap \exists \mathsf{pAR.ActingPlayerRole} \sqcap (=1\ \mathsf{hasHalfMove}^{-}.\mathsf{ChessGame}) \tag{6}$$

$$\mathsf{hasHalfMove} \sqsubseteq \mathsf{subEventOf}^{-}, \mathsf{hasFirstHalfMove} \sqcup \mathsf{hasLastHalfMove} \sqsubseteq \mathsf{hasHalfMove} \tag{7}$$

$$\mathsf{HalfMove} \sqsubseteq (\leqslant 1\ \mathsf{nextHalfMove.HalfMove}) \sqcap \neg \exists \mathsf{nextHalfMove}.\mathit{Self} \tag{8}$$

Next, a chess game may have information about the game result – captured by the ChessGameResult class – which may be encoded in SAN. A chess game may also be associated with a standard chess opening from the Encyclopedia of Chess Opening (ECO). In addition, a chess game may be a subevent of a chess competition round and may have a chess game manifestation as explained below.

Chess Competition. Important notions here are *round*, *tournament instance*, and *tournament series*, respectively represented by the Chess Competition Round, Chess Competition Instance, and Chess Competition Series sub-patterns. They are all modeled as events, so we simply assert:

$$\mathsf{ChessCompetitionRound} \sqcup \mathsf{ChessCompetitionInstance} \sqcup \mathsf{ChessCompetitionSeries} \sqsubseteq \mathsf{Event}$$

Chess Game Manifestation. From Figure 1, a chess game may have a chess game manifestation, represented by the ChessGameManifestation class that may provide roles, e.g., author roles for some agents – the existence of an agent for a given agent role was asserted in (3). Note that chess game manifestations are not events, but rather creative works by humans. A chess game manifestation may originate from some external source or PGN file indicated by a URI. A chess game manifestation may have annotations for half-moves containing comments given as a string. We leave a complete schema for all possible structures of the content of a PGN file for future work. So, we assert:

$$\mathsf{AuthorRole} \sqsubseteq \mathsf{AgentRole} \sqcap \exists \mathsf{pAR}^{-}.\mathsf{ChessGameManifestation} \tag{9}$$

$$\mathsf{HalfMoveAnnotation} \sqsubseteq \exists \mathsf{annotatesHalfMove}.\mathsf{HalfMove} \sqcap \exists \mathsf{hasContent}.\mathsf{xsd{:}string} \tag{10}$$

We can also assert that half-moves annotated in a chess game manifestation have to be the half-moves of the corresponding chess games as in the following axiom. This is, however, would lead us beyond OWL 2 if combined with (6).

$$\mathsf{hasManifestation} \circ \mathsf{containsAnnotation} \circ \mathsf{annotatesHalfMove} \sqsubseteq \mathsf{hasHalfMove} \tag{11}$$

Class Disjoitness, Domain and Range Restrictions. We assert class disjointness for every pair of classes in this pattern, except when the pair contains two classes for which a subclass axiom is explicitly asserted. We also assert guarded domain and range restrictions for every property in the pattern, e.g., for hasHalfMove, this would be in the form of:

$$\exists \mathsf{hasHalfMove}.\mathsf{HalfMove} \sqsubseteq \mathsf{ChessGame}, \quad \mathsf{ChessGame} \sqsubseteq \forall \mathsf{hasHalfMove}.\mathsf{HalfMove} \tag{12}$$

3 Conclusion

We have presented an ontology pattern for chess games. This pattern, on the one hand, coves key notions understood in chess playing community. Some parts of the pattern were left underspecified as modeling them in full would require separate patterns and more in-depth analysis. As the next step, we will investigate how this pattern would fare in the context of linked data publishing of chess games data that are widely available online.

Acknowledgements. This work was partially supported by the National Science Foundation under award 1440202 *EarthCube Building Blocks: Collaborative Proposal: GeoLink – Leveraging Semantics and Linked Data for Data Sharing and Discovery in the Geosciences*, by the European Commission under the International Reseach Staff Exchange Scheme (IRSES) of the EU Marie Curie Actions, project *SemData – Semantic Data Management*, and by the Spanish Ministry of Economy and Competitiveness (project TIN2013-46238-C4-2-R).

References

1. Chess Redux: how many people play chess? `http://en.chessbase.com/post/che-redux-how-many-people-play-che-` (August 2012), last accessed: July 7, 2015
2. Chess.com. `http://www.chess.com` (Last accessed: July 7, 2015)
3. Krötzsch, M., Simančík, F., Horrocks, I.: A description logic primer. CoRR abs/1201.4089 (2012)
4. Rodríguez-Doncel, V., Krisnadhi, A., Hitzler, P., Cheatham, M., Karima, N., Amini, R.: Pattern-based linked data publication: The linked chess dataset case. Tech. rep., Data Semantics Lab, Wright State University (2015), available from `http://dase.cs.wright.edu/publications/pattern-based-linked-data-publication-linked-chess-dataset-case`