Design and Evaluation of Database and API Supporting Shogi Learners on the Internet

Ryo Miura   Toshiki Kinuhata   Shinji Ohara
Hiroyuki Tarumi   Toshihiro Hayashi   Junko Ichino
Department of Reliability-Based Information Systems Engineering
Faculty of Engineering, Kagawa University
Takamatsu, Kagawa, JAPAN
e-mail: s15g484@stu.kagawa-u.ac.jp, tarumi@eng.kagawa-u.ac.jp

Abstract— **We have developed a Shogi (Japanese Chess) support system on the Internet, called SAKURA. SAKURA provides an environment for Shogi with game and discussion support functions. It also has databases and interfaces with artificial intelligence to play shogi. The databases of SAKURA consist of a game record database and a position database. These databases have links to each other.   As a result, information associated with game positions, such as comments, evaluations, and candidate moves can be integrated on a single system.   We have conducted experiments to evaluate the performance of API for the databases. This paper describes SAKURA's database design and results of the experiment for performance evaluation. The performance is efficient enough to be extensively used by external artificial intelligence.**

Keywords-shogi; database; web services

I. INTRODUCTION

A. Background

Shogi is a variant of chess-like games, which is very popular in Japan. Among all chess variants, shogi is the most complex one, mainly because it allows reuse of captured pieces.

Along with the progress of information technology, many systems for shogi have been developed. For example, game servers on the Internet and software to play shogi, etc., are popular services or products. Especially, technologies for artificial intelligence players have been developed by many researchers.   The top-level artificial intelligence shogi playing programs won against professional players by 4 to 1 in the series of "Dennousen" [1] (spectacle shogi games between professional players and artificial intelligence) in March, 2014. Examples of popular game servers on the Internet are, for example, "Shogi Club 24" [2] and "81 Dojo" [3]. If players use "81 Dojo", they can play games with world-wide players, since it is a multi-language service.

Many systems for shogi have databases to store game records. A game record is called "Kifu" in Japanese. "Shogi no Kifu Deeta-Beesu" [4] is a typical and popular service, which provide game records played by professional and/or top-level amateur players. However, databases of positions are usually lacked on such services.   A position, or "Kyokumen" in Japanese, means a status of shogi game. A position is defined by the arrangement of pieces on the game board, captured pieces in hands of both players, and the game turn. For example, the tenth position in a game is the status of the game board after ten moves from the initial status of the game. In case of chess, there are databases which includes position databases (for example, chessliB [5] and chessBase [6]).   However, in case of shogi, the complexity of position is not reduced toward the end of a game, because the number of pieces does not decrease by capturing. Therefore, it is almost impossible to provide an end-game database like that of chess. Moreover, the possible number of positions is estimated as 10^{220} in case of shogi, whereas it is 10^{120} in case of western chess. These facts make it difficult to provide position databases.

Recently, "Kyokumenpedia" [7], a database of positions on the web, has been developed and its beta version is now opened to access.   It stores data of positons including comments of study and candidate moves at each position. However, it does not support API (Application Program Interface) for external software.

A position database will be useful for players to make discussions on their games. After a game, players often look back the game and make discussions to determine the best move at particular positions in the game. This kind of post-game discussion is called "Kansousen" in Japanese, and very popular among shogi players. If all information about a position, including comments by excellent players and evaluation by artificial intelligence, is integrated in a single position data record and easy to access, it will help the discussion.

Here we review how existing services give functions for post-game discussions.   For example, "Shogi Club 24" provides only chatting interface for discussions. In case of "81 Dojo", it has graphical interfaces for discussions to draw arrows on the board, but does not support users to deal with variations of a game. In additional, these systems cannot save comments given in discussions and share them with other players on the net.

B. Goal of our Research

Many systems to support shogi games and many AI programs have been released. However, these systems are basically stand-alone and cannot communicate with each other. In order to better utilize such systems, we need a
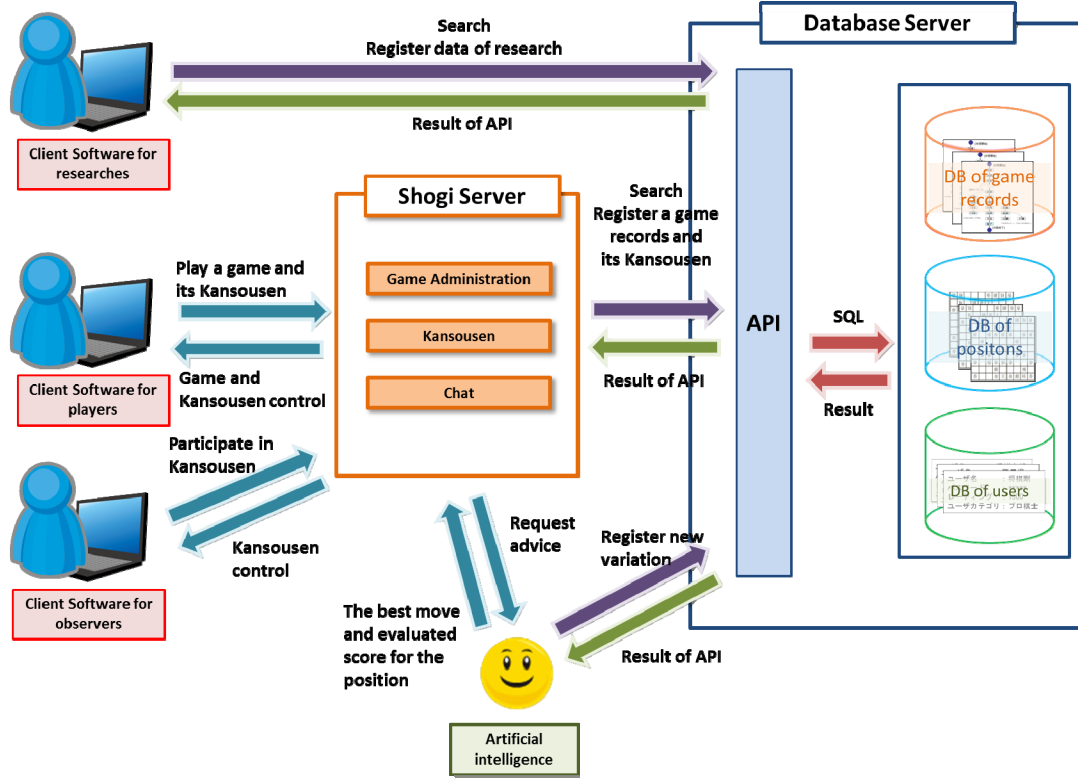
Figure 1. SAKURA's system architecture

platform to incorporate them. Especially, AI programs would be used for other purposes than playing shogi games, if they are connected to the platform. For example, they will help post-game discussions, or they will give recommended moves by considering positions in the database.

We are developing a platform for shogi players and software on the Internet, which supports playing games, post-game discussions (Kansousen), and studies of shogi with databases and API. We have named it SAKURA (Shogi Archives and Kansousen Utilities for Research and Advice) [8].

In this paper, we describe the design and implementation of SAKURA, especially its database and API. Results of the experiment for performance evaluation of API and databases are also presented, which proves that they are effective enough for AI programs to extensively access the databases and give new skillful data increasingly.

II. DESIGN OF SAKURA

Figure 1 shows the total design of SAKURA architecture. SAKURA has three kinds of client software. Shogi players can play games and enjoy post-game discussions using the client software for players. SAKURA also provides the second type of client software for observers. Observers enjoy watching games of other players remotely. They can chat with other observers during the game and may

participate in post-game discussions. The last client software is for researchers who study shogi.

Client software is used by players and observers; they are connected to the shogi server. The shogi server controls these clients when players are playing games or discussing with observers.

We are also developing a system which has links to artificial intelligence programs developed by other researchers. Artificial intelligence programs have two roles in our architecture: supporting with advices during post-game discussions and giving new variations and comments on positions stored in the database. Here, a variation means a series of alternative moves from one of the positions in the game, which were not played in the actual game. Currently, we have interfaces to Bonanza Feliz[9] artificial intelligence program. However, Bonanza Feliz cannot give players advices on positions in variations. Bonanza Feliz follows the only actually played moves, from the opening, step-by-step, and gives new variations based on their consideration. Therefore, we will use artificial intelligence programs supporting USI (Universal Shogi Interface) [10], which can consider any given positions.

The databases of SAKURA consist of a game record database and a position database. These databases have links to each other. SAKURA's database stores records of games, positions, and records of post-game discussions. SAKURA's databases also include a user database for their
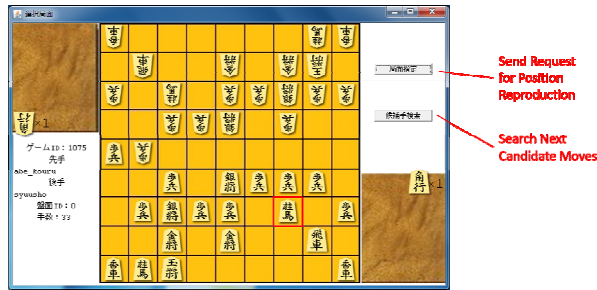
Figure 3. Shared board interface for discussion

Figure 2. Personal board interface for discussion

Figure 4. Chatting interface for discussion

Figure 5. Tree chart interface for discussion

management. SAKURA provides API. If SAKURA's client software, the shogi server, artificial intelligence programs, or other external software access databases, they must use the API to avoid inconsistent operations, and to make the client codes simpler. Currently more than 220,000 game records (the game records of floodgate [11]) and positions that appeared in them (more than 27 million) have already been stored.

III. SAKURA'S CLIENTS

SAKURA clients provide a shared board interface (Figure 2) and a personal board interface (Figure 3). The shared board interface supports WYSIWIS and shared by all participants of the discussion. The shared board interface has some important commands: to start chatting, to request for advice to an artificial intelligence, to save comments, and to show the tree chart. On the other hand, the personal board interface is provided for each participant and operations on this interface is not shared with other personal board interfaces. Participants are allowed to move pieces freely for themselves. If participants push the "Send Request for Position Reproduction" button, the position on the personal board interface is shared with other shared board interfaces.

SAKURA clients also have a chat function (Figure 4) and the tree chart interface (Figure 5). The chat function is not developed by us, but implemented with free software. The
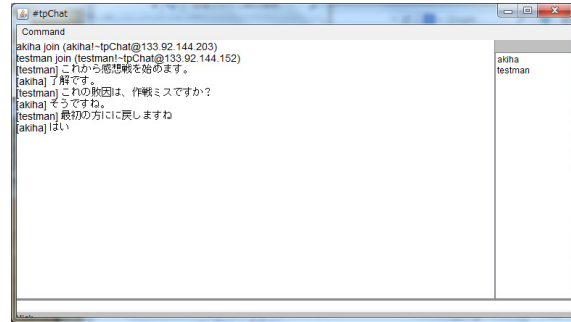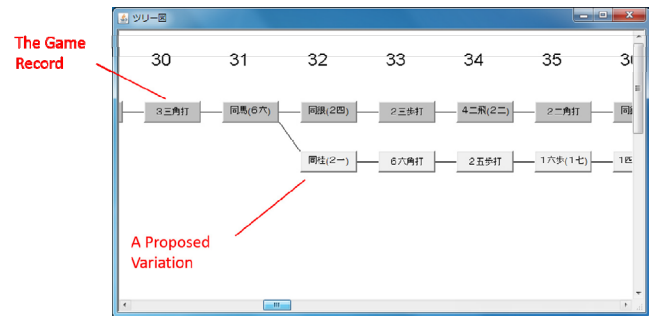
chat function is used not only by participants of the discussion, but also by artificial intelligence programs.

The tree chart interface shows the game record (the main horizontal sequence) and its proposed variations (branched sequences). This interface supports WISIWIS. When a variation is proposed, a new node is appended. When a user clicks a node in the tree, the user can reproduce the position at the node on the personal board.

IV. DATABASE AND API

A. Design of the Database

SAKURA's database consists of game record database and position database. The database of game records stores data of actual games (game records, player names, winners, venues of games, date and time, etc.). It also saves comments and variations given during post-game discussions. The position database collects data associated with positions: comments on the position, candidate next moves, etc. We have also defined the format of position data (535 bits: first 486 bits represent the arrangement of pieces on the game board, next 48 bits represent pieces in hands of both players, the last bit shows the turn of game) for the primary key of position database. These databases have links to each other and it is possible for a participant to use comments and candidate next moves on positions that appeared in other game records during discussion. For example, Figure 6 shows mutual links of game record database and position database. Each of the game record A and B has an identical position. In case of game record A, the position data has a comment "The best move is W58G". In case of game record
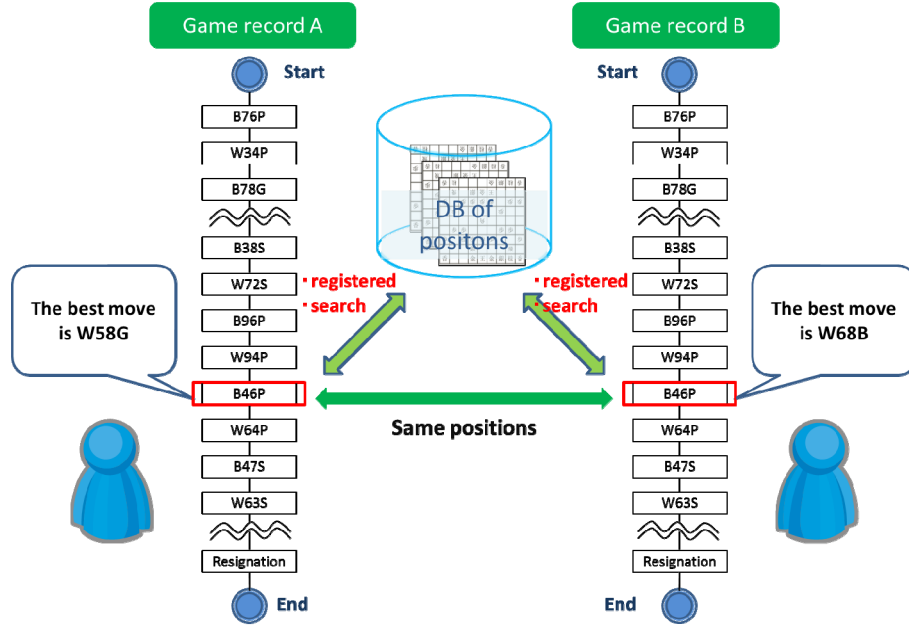
Figure 7. Mutual links of game record database and position database

B, the position data has a comment "The best move is W68B". Without a position database, players cannot find the comment on game record B during the post-game discussion of game record A.

With this database architecture, players and observers can also read comments and variations given by other players or observers. As a result, their post-game discussions will be more active.

B. Variations

Variations are stored in the variation table with game record ID and variation ID. These IDs are primary keys. When a user saves one variation in the variation table, a variation ID is given which shows the position where the variation is given. Variation ID 0 is reserved for the actual moves. Figure 7 shows an example of game record and its variations in discussion.

C. Candidate Moves

A candidate move means a recommended move or a move likely to be played by players at each position. Actual moves on game records and best moves given by AI programs are also regarded as candidate moves. Users can find a skillful move played by other users or standard moves, by retrieving them from the position database.

D. Comments

In SAKURA, comments are categorized into two types: comments on positions and comments on game records. These two types of comments have different significance. Learners mainly need comments on positions. For example, "which player has advantage at this position, and why?", or "what are the next candidate moves and how is their
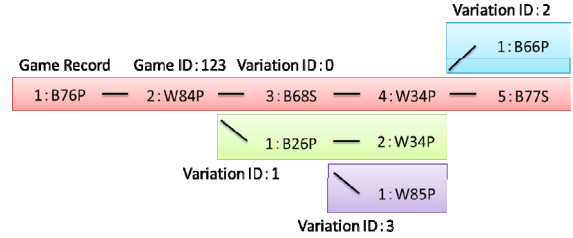

Figure 6. Example of game record and its variations

comparison?" However, some comments are not useful for learners. For example, during the discussion, players often give comments like "I failed to find this move" or "I am proud of this beautiful move of pawn." These comments are subjective and should be associated with game records, not with positions. It is because the position data is independent of players. Therefore, comments on game records are stored in the game records database, whereas comments on positions are stored in another database. With this design, learners can read comments only on each position when they study.

E. Information Related to Users

Variations, candidate moves, or comments can be given with information related to users: proposer, rating, and user's category. The proposer means the person who proposed variations, candidate moves or comments. The rating means player's skill level of shogi. The skill level is expressed by integer values. If the player has high shogi skill, his/her skill level is represented with larger numbers. The user's category is defined by us as one of the following six:

anonymous players, professional players, ladies professional players, apprentices of professional players who are challenging to get professional licenses, amateur players, and artificial intelligence programs.

If variations, candidate moves, and comments are stored in the database with information related to users, players and observers can confirm who proposed them. In addition, players and observers can refer to the data, in order to determine which variations, candidate moves, or comments are the best.

F. Links of the Game Records Database and the Positions Database

The position database also has links from each position record to game records that include the position, in the game records database. The link is represented by a game ID, a variation ID, and the position number. As a result, users can find the game records where the position appears.

G. API

Mutual links between databases of SAKURA make it complex to operate the databases. Therefore, SAKURA provides API for external processes operating the databases.

The following is a list of API functions required during discussions.

- Saving game records
- Saving positions
- Saving variations of a game
- Saving candidate moves
- Saving comments on game records
- Saving comments on positions
- Updating game records
- Searching for game records
- Searching for positions
- Deleting game records
- Deleting variations of the game
- Deleting comments on game records
- Deleting comments on positions

Additionally, SAKURA's API also provides user management functions.

The API is implemented with PHP and it communicates with client software, shogi server, AI programs, etc., via http. API receives POST data from these systems and sends XML data to them.

V. ARTIFICIAL INTELLIGENCE PROGRAMS

Artificial intelligence programs are developed by many researchers, and they have excellent abilities for playing shogi. Especially, the top-level artificial intelligence programs often win against professional shogi players. In addition, some of the moves played by artificial intelligence have been adopted also by several professional shogi players in their tournament games, recently. This fact shows that artificial intelligence programs discovered new standard moves.

SAKURA provides an interface with Bonanza Feliz, which is also an artificial intelligence program. With the interface, Bonanza Feliz provides advices during post-game

discussions, and new candidate moves at positions stored in the database. However, Bonanza Feliz has a problem. It can only give candidate moves and evaluated scores for an actual game record, but cannot give them for its variations or arbitrary positions. The problem is caused because Bonanza Feliz follows the game with only actually played moves, from the opening, step-by-step, and gives new variations based on their consideration, simply because it does not support full USI. Hence, we will develop an interface with artificial intelligence programs supporting USI, which supports exchange of arbitrary position data between programs on the Internet.

VI. EVALUTION OF DATABASE AND API

A. Purpose of Evaluation

We will develop an interface which provides links to artificial intelligence programs supporting USI. Using the interface, AI programs can give the best move and evaluated score for a given position in SAKURA's position database. In this case, artificial intelligence programs access the database via API enormous times, repeatedly. Hence processing-time of API significantly affects the total time of exploration. To confirm that the API is efficient enough to realize such automatic processing by AI, we made an experiment to measure the processing-time of API and evaluated the API performance.

B. Target API for Evaluation

Artificial intelligence programs execute three processes: searching positions, calculating candidate moves and their evaluated scores, and storing them on databases. Therefore, artificial intelligence programs require three functions of API (searching for positions, saving candidate moves, and saving comments on positions) to give the best moves and evaluated scores for positions.

We have selected these API functions for the experiment of performance evaluation.

C. Outline of the Evaluation

We conducted the evaluation on an Ubuntu server, which is virtual server on Windows 8. The CPU type number was Intel R Xeon R CPU E5-2620 v2 @2.10GHz, and Ubuntu was assigned 2 CPU cores and 2 GB memories.

API functions of saving candidate on positions, saving comments on positions, and searching for positions require arguments of position data and candidate moves, position data and comments data, and only position data, respectively. For example, in case of the API function of saving candidate moves, position data and candidate moves are fetched from game records (the function follows game records and fetches moves from each of the positions). The position data is also used for other two API functions: saving comments on positions and searching for positions. As for the comments data for saving comments on positions, we generated random character strings and used them as comments data, because the texts of comments data do not affect the performance.

The following list is the procedure of the evaluation.
1) Prepares an empty table of positions.
2) Executes the API function of saving candidate moves.
3) Executes the API function of saving comments on positions.
4) Executes the API function of searching for positions.
5) Goes back to step 2. The loop repeats 10,000 times.

When each function was executed, we measured the elapsed-time of each function, instead of the CPU time. The number of data in SAKURA's database was increased by repeating the loop. Therefore, we checked the increase of elapsed-time associated with the increase of data in the database.

We repeated the processes three times.

D. Result

The results are shown in Figure 8, 9, and 10. Red lines in figures show the regression lines, with regression formulae as shown. The measured data are scattered because they are not the CPU time, but the elapsed-time. The regression coefficients for the number of trials of each formula can be neglected. The elapsed-time of saving candidate moves, saving comments on positions, and searching for positions are 0.0423 sec., 0.0369 sec., and 0.033 sec., respectively. Therefore, when artificial intelligence programs execute the process, the operating time is estimated as 0.1122 sec. Artificial intelligence programs require more than 10 seconds for computing candidate moves and evaluated score (however, the time depends on the setting of artificial intelligence programs). We confirmed that the operating time of the database is smaller than AI's own computation.

VII. Conclusions and Future Work

This paper described SAKURA's design and implementation, especially the database and API. SAKURA's database consists of the game records database and the positions database. These databases have links to each other and it is possible for participant to use comments and candidate next moves on positions which appeared other game records, during their discussion on shogi games.

We have developed the API to operate SAKURA's database. The API has two roles: avoiding inconsistent operations and making the client codes simpler.

For the future work, we will develop an interface cooperated with artificial intelligence programs supporting USI. With the interface, artificial intelligence programs incrementally give candidate moves and evaluated scores for SAKURA's positions database. We expect it will be able to discover new standard moves. On this purpose, we have made the evaluation to measure processing-time of the API. The experiment shows that processing-time of the API does not affect the performance of the exploring process.

Acknowledgment

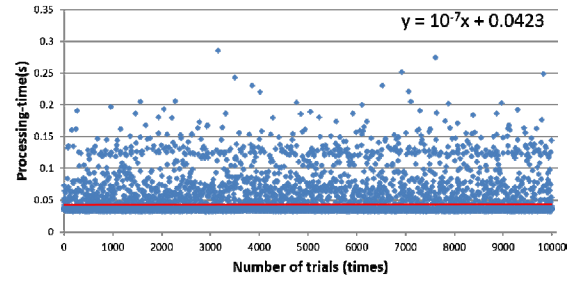We appreciate comments from Ms. Madoka Kitao (lady's professional shogi player) and Mr. Tomohide Kawasaki (81dojo.com).
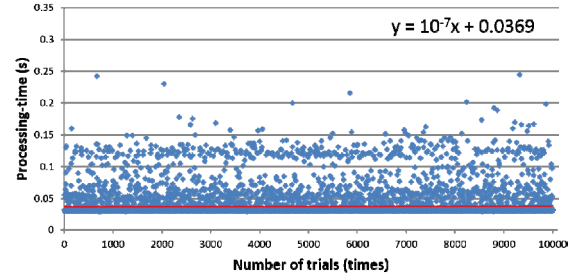

Figure 8. Saving candidate moves

$y = 10^{-7}x + 0.0423$

Figure 9. Saving comments on positions

$y = 10^{-7}x + 0.0369$

Figure 10. Searching for positions

$y = 2.0 \times 10^{-8}x + 0.033$

References

[1] Japan Shogi Association : http://www.shogi.or.jp/

[2] Shogi Club 24 : http://www.shogidojo.com/

[3] 81 Dojo : http://81dojo.com/

[4] KIFU Database (in Japanese) : http://wiki.optus.nu/shogi/

[5] ChessliB : http://folk.uib.no/pfvaf/chesslib/

[6] ChessBase Database : http://shop.chessbase.com/en/categories/123

[7] Kyokumenpedia : http://kyokumen.jp/

[8] Tagashira, Y., Tarumi, H., Toshima, S., Ohara, S., Kinuhata, T., and Hayashi, T.: A Networked Discussion Tool for Shogi, Proceedings of the Late-Breaking Papers in Collabtech 2014 (Online), pp. 23-30 (2014)

[9] Bonanza Feliz : http://www.geocities.jp/bonanza_shogi/

[10] The Universal Shogi Interface(USI) : http://www.glaurungchess.com/shogi/usi.html

[11] Computer Shogi Non-stop Playing Field -- Floodgate (in Japanese) : http://wdoor.c.u-tokyo.ac.jp/shogi/floodgate.html