International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:9, No:1, 2015

Continuous FAQ Updating for Service Incident Ticket Resolution

Kohtaroh Miyamoto

***Abstract*—**As enterprise computing becomes more and more complex, the costs and technical challenges of IT system maintenance and support are increasing rapidly. One popular approach to managing IT system maintenance is to prepare and use a FAQ (Frequently Asked Questions) system to manage and reuse systems knowledge. Such a FAQ system can help reduce the resolution time for each service incident ticket. However, there is a major problem where over time the knowledge in such FAQs tends to become outdated. Much of the knowledge captured in the FAQ requires periodic updates in response to new insights or new trends in the problems addressed in order to maintain its usefulness for problem resolution. These updates require a systematic approach to define the exact portion of the FAQ and its content. Therefore, we are working on a novel method to hierarchically structure the FAQ and automate the updates of its structure and content. We use structured information and the unstructured text information with the timelines of the information in the service incident tickets. We cluster the tickets by structured category information, by keywords, and by keyword modifiers for the unstructured text information. We also calculate an urgency score based on trends, resolution times, and priorities. We carefully studied the tickets of one of our projects over a 2.5-year time period. After the first 6 months we started to create FAQs and confirmed they improved the resolution times. We continued observing over the next 2 years to assess the ongoing effectiveness of our method for the automatic FAQ updates. We improved the ratio of tickets covered by the FAQ from 32.3% to 68.9% during this time. Also, the average time reduction of ticket resolution was between 31.6% and 43.9%. Subjective analysis showed more than 75% reported that the FAQ system was useful in reducing ticket resolution times.

***Keywords*—**FAQ System, Resolution Time, Service Incident Tickets, IT System Maintenance.

I. Introduction

THE costs of IT system maintenance (for both software and hardware) pose significant problems for enterprises as systems grow in size and complexity [1]. Often large numbers of employees must be assigned globally [2] to the support and maintenance teams [3].

Each service incident is typically managed and controlled by a set of "tickets". Typical tickets include RFI (Request for Information) tickets when end users request answers to questions and RFC (Request for Change) tickets when end users request changes in the target system. Many details are defined in the ITIL (Information Technology Infrastructure Library) [4].

The importance of knowledge management is widely acknowledged [5], so to reduce costs [6] we decided to consider the use of a knowledge management approach for FAQs to help ticket-resolution practitioners share their knowledge with each other.

Kohtaroh (Cole) Miyamoto is a researcher at IBM Research – Tokyo, IBM Japan (e-mail: kmiya@jp.ibm.com).

A. Typical Use-Case

Fig. 1 shows a very simplified example of how a ticket is handled. Initially, the ticketing system issues a new ticket for each service incident. Based on its category, the ticket is assigned to an available ticket-resolution practitioner. The practitioner will attempt to refer to the FAQ database to support the resolution process. If the FAQ doesn't help resolve the problem, then the practitioner will remediate the issue by diagnosing the details and eventually resolve the ticket. The solution is then provided to the end user. Periodically (in a manual system), practitioners may identify a recurring problem that required a longtime to resolve, (i.e. due to its complexity) and update the FAQs in the database.

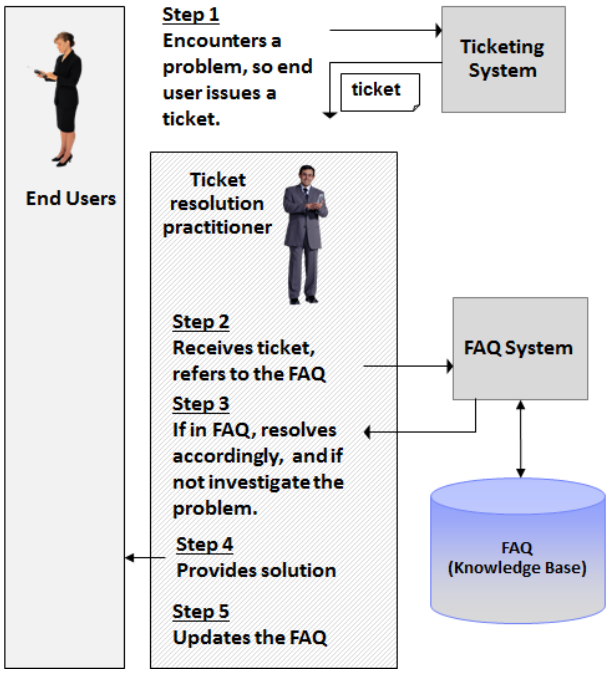

Fig. 1 Use Case Example of Ticket Handling

B. Definition of Tickets

Service incident tickets (tickets) usually include many types of structured information and some unstructured text data. The structured information is basically predefined values for specific types of data. The unstructured text data is free-format written text describing the incident in greater detail. Though there are no firm standards for this, we have identified some

typical patterns. A simplified but typical pattern appears in Table I.

TABLE I
TICKETS

Name	Mandatory?	Description
ID	Yes	Unique identifier to distinguish each ticket
Title	Yes	Short description of the ticket
Resolution Time	Yes	The actual working time spent for resolution of the ticket. (Details below.)
Category	Yes	The category of the ticket. The granularity and the definitions differ among ticket domains.
Sub-Category	No	If the domains are broad, sub-categories may be defined.
Priority	Yes	Ticket handling is prioritized.
Description	Yes	The initial source of the description is the end user, but the practitioner adds key details.
Opened time	No	When the incident was reported.
Closed time	No	When the incident was fully resolved.
Root Cause	No	Root cause identification is extremely useful for later analysis.

"Resolution time" is not merely the elapsed time between the "Opened time" and "Closed time", but should represent the actual working time spent in resolving the ticket [7]. For example, practitioners often need to ask for more details from the end user and the time spent waiting for the responses should not be included. Also, it does not include the time the ticket is queued (waiting to be processed), either because no practitioner is available or because it has been assigned to a specific practitioner who is working on other tasks. Lastly it does not include non-working hours. The time recording is mostly automated but there is still some manual recording. This may appear to be a burden for the practitioners work, the "Resolution time" for each ticket is quite important for various kinds of ticket analyses.

Some of the ticket analysis outputs include resource optimization for the practitioners and ticket trend predictions. However, in the work reported here, we are focusing on the long-term relationships between the resolution times of the tickets and the evolving conditions of the FAQ (Frequently Asked Question) database.

C. Problem to Solve

We have previously shown that preparing a FAQ can help reduce the ticket resolution time [8]. But when we proposed our technology for use in real projects, the practitioners expressed some serious concerns. The primary concern was that the domain knowledge evolves over time. Any FAQ can soon become obsolete, and as the FAQ coverage becomes low the FAQ may increase resolution times, resulting in more dissatisfied practitioners and customers.

To address these concerns we carefully studied past tickets to develop a novel method of automatically updating the FAQ database from the ticket information.

Here are the FAQ metrics we defined for our method:

1) Knowledge Coverage

How does the ticket coverage of the FAQ decline without periodic updates? Can our continuous update method prevent or reverse the decline and keep the ticket coverage at a high level?

2) Resolution Time Improvement

Our earlier work showed how ticket resolution times qre improved by preparing a FAQ. Can we further improve the ticket resolution time?

3) Subjective Evaluation

The practitioners tend to benefit from the FAQ in resolving tickets when the FAQ is fresh. However, over time the FAQ data tends to grow, which can make it more difficult to use the FAQ effectively. The user interface of the FAQ system is not the primary focus of our current work, but it is important to collect subjective impressions about the utility and applicability of the FAQ system. As a metric, we can ask about the long-term subjective satisfaction levels of the practitioners?

This paper is structured as follows. In Section II we introduce our methods for designing the FAQ data structure and explain the FAQ system user interface, our ticket clustering techniques, and the FAQ matching. Section III shows the results of our techniques in a long term project. Section IV covers related work. The final Section V gives our conclusions and discusses future work.

II. Methods

A. Dialog Based FAQ System

The concepts of our FAQ data structure are shown in Fig. 2. The characteristics of this design are:

1) Hierarchical Structure

The FAQ is structured in hierarchical layers, with the information going down from a general layer to more detailed information.

2) Dialog-based interaction

The basic user interface format is a "dialog-based" interaction where it is possible to navigate the focus through the nodes. The user of this system is prompted to make selections from each node's links. Each selection moves the current focus to a new node.

3) Directed Acyclic Graph

Technically, this is a "directed acyclic graph", where the graph nodes are connected with directed links and cycles are not permitted (to prevent infinite loops).

4) Start Node and Answer Node

The FAQ is entered via the start node and the final node will be an answer (from the FAQ).

5) Ticket-Aligned Category and Sub-category Nodes

The Category and Sub-category nodes depend on the ticket domains, so they are not universally pre-defined. The current definitions and links are stored in the ticket system configuration. The FAQ categories and sub-categories should be an exact match to the categories and sub-categories of the tickets for each target domain, but in practice this is often difficult to achieve.

6) Keyword Nodes

Each keyword node defines a characteristic word (usually a noun) that links to a FAQ topic. They may be technical proper nouns such as “Internet Explorer” or “Open Office”, or just basic terms such as “stock” or “error”.

7) Node Merging

The nodes can merge from multiple links of nodes.

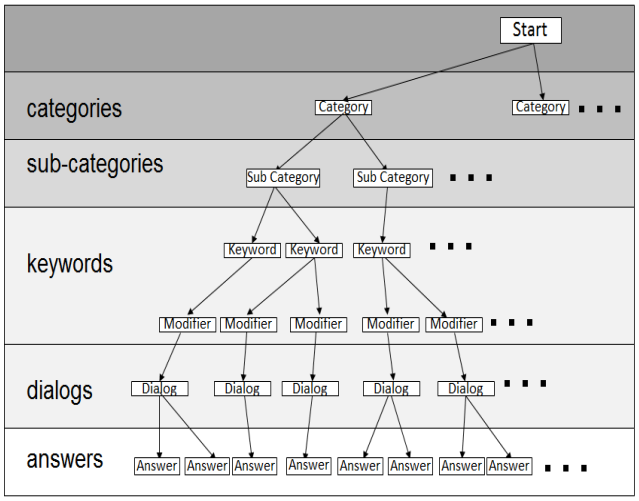

Fig. 2 Tickets by Category

To support this FAQ format, we created a dialog-based FAQ system. The grand design of this system has these characteristics [9]:

1) Dialog based FAQ navigation

Even if a user practitioner does not know where to start, the “start node” is a default entry before navigating to an answer.

2) Authorization Level

There are 3 levels of user authorization for each user. One level is the administrator level to navigate, add dialogs, make arbitrary changes to dialogs, and do administrative-level configuration changes. The second level is an editor level where dialogs can be changed or added. The reader level only allows navigating among and reading the dialogs.

3) Hybrid search (Keyword based and dialog based navigation)

Users who are familiar with the system want to jump quickly and directly to the answers. To support faster access, there is a hybrid search feature so users can combine keyword searches with a dialog-based approach.

4) Best Alternative Node

During the navigation of the FAQ, the focused node will show a list of candidate nodes to choose from. Unfortunately, the exact candidate solution is not always included in the list. We have studied this case carefully and found that even if the exact candidate is not shown it is quite common that an alternative node can show the answer to the question at hand. So when there is no exact match in any of the candidates, they system would automatically identify the best alternative node. In order to perform this identification, the system would utilize information available on the internet and examine the feature, name and release timing information. And then navigate the user to that node.

5) Rich Editor User Interface

The usability of the FAQ editor is very important. Users must be able to make changes easily, for example changing the nodes or links in the FAQ. We devised a feature-rich and intuitive editor user interface that uses intuitive mouse operations such as drag and drop and left and right clicks [9].

6) Complete Logs

The navigation paths for each user are stored in a database. The system uses this database to allow the user to resume suspended navigation. Users can refer to previous paths and start navigation from any previously visited node, which is often faster than starting at the start node.

7) Ticket Number and FAQ Relationships

When an FAQ node resolves a ticket, the ticket number is recorded in the FAQ. This data helps determine the ticket coverage ratio of the FAQ and of each node. This will be discussed in later sections.

B. Initial Ticket Clustering

The basic source of FAQ knowledge is the experience of resolving tickets. If the number of tickets is too small, the RoI of FAQ creation may be too low. FAQs are typically only after some number of resolved tickets can be referenced to create the initial set of FAQ database.

Our method of creating the initial FAQ from the set of initial tickets uses categorization, keyword clustering, keyword modifier clustering, urgency scoring, and FAQ formatting, as shown in Fig. 3.

Here are more details about these steps:

1) Category Grouping

The tickets are initially grouped by categories and the sub-categories to map to a corresponding FAQ category and sub-category structure. If the FAQ structure is a precise match with the ticket category and sub-category definitions, then there are no technical issues. However if for various practical reasons such as tickets covering multiple systems with diverse category definitions or any other similar situation (as often happens in real-world projects) there are discrepancies, then the system matches the tickets to categories based on the relevance of names and features.

2) Keyword Clustering

The categorized tickets are clustered by keywords. The keywords are distinctive words (usually nouns) that appear in each the tickets of each category. We also prepared a domain-level synonym dictionary to map each group of synonyms into are presentative keyword. Our keyword clustering technique is an extension of LDA (Latent Dirichlet Allocation). While LDA needs a number of clusters as input,

our extension can calculate an appropriate number of clusters based on input criteria.

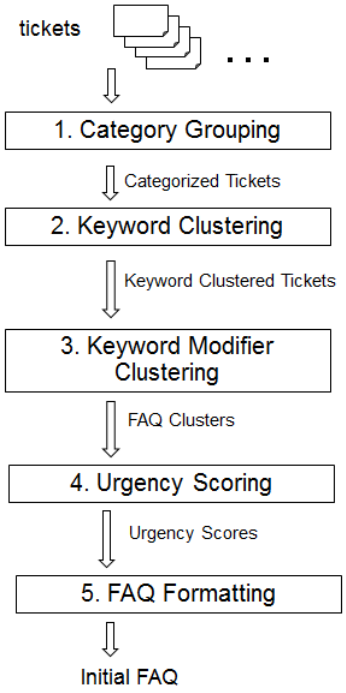

Fig. 3 Initial FAQ Creation Method

3) Keyword Modifier Clustering

The keyword modifiers are nouns or verbs that modify the keywords [10]. Again, we prepared a domain-level synonym dictionary to map each group of synonyms into are presentative modifier keyword. Examples include “display”, “transfer”, or “delete”.

4) Urgency Scoring:

We calculate the urgency U(i) of each cluster i based on 3 factors (trend score, resolution time score, and priority score) of the tickets in each cluster I, as shown in (1). The trend score Trend() shows the degree to which the volume of related tickets is rising over time. We used the least squares method [11] to calculate the slope of the trend line. The resolution time scoreResol() is based on the total resolution time The priority score Prio() is based on the average “priority” value of the tickets. W_T, W_R, and W_P are the weights of the scores for Trend(), Resol(), and Prio(), respectively.

$$U(i) = w_T Trend(i) + w_R Resol(i) + w_P Prio(i) \quad (1)$$

5) FAQ Formatting

A powerful editor, which interfaces with specialized tools, is used to create and update the FAQ [9]. The complete user interface is beyond the scope of this paper.

C. FAQ Updating

After the initial FAQ creation, the FAQ needs to be periodically updated. Some obvious reasons are new trends in the problems recoded in the tickets or removing obsolete data or adding new information to the FAQ.

From the technical perspective, FAQ updating is clearly different from initial FAQ creation. It is not good practice to reformat the FAQ system too heavily when you take into consideration the usability consistency. A practitioner who expects to find a previously referenced answer may be unable to find it again if the entire structure is completely reformatted. Therefore the ticket information from the previous FAQ should be similarly accessible in the updated FAQ. The FAQ formatting process should involve minimal updates.

Our method of FAQ updating is shown in Fig. 4.

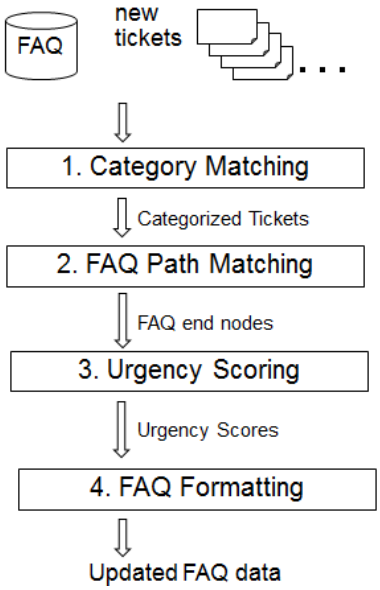

Fig. 4 Method to Update FAQ Data

Here are more details about the steps in updating the FAQ:

1) Category Matching

This process is similar to the corresponding “Initial Ticket Clustering” step.

2) FAQ Path Matching

In this step, the ticket information is matched to the FAQ Path, which is any FAQ navigation route from the start to an end node [12]. Since the FAQ path and the ticket information are in hierarchical structures, we can assume that higher in the hierarchy the higher the weight of the corresponding matches. The weight of each match is calculated based on the keyword match obtained from a tf-idf method [13] where tf is the term frequency (the number of frequencies in the document) and idf is the inverse documents frequency (the relative rareness of the word in the FAQ domain). If the score of the match is above the criterion, the FAQ path with the highest score is the matched FAQ path. Otherwise, a new node is defined. A simple illustration which combines both category matches and FAQ path Matching is shown in Fig. 5.

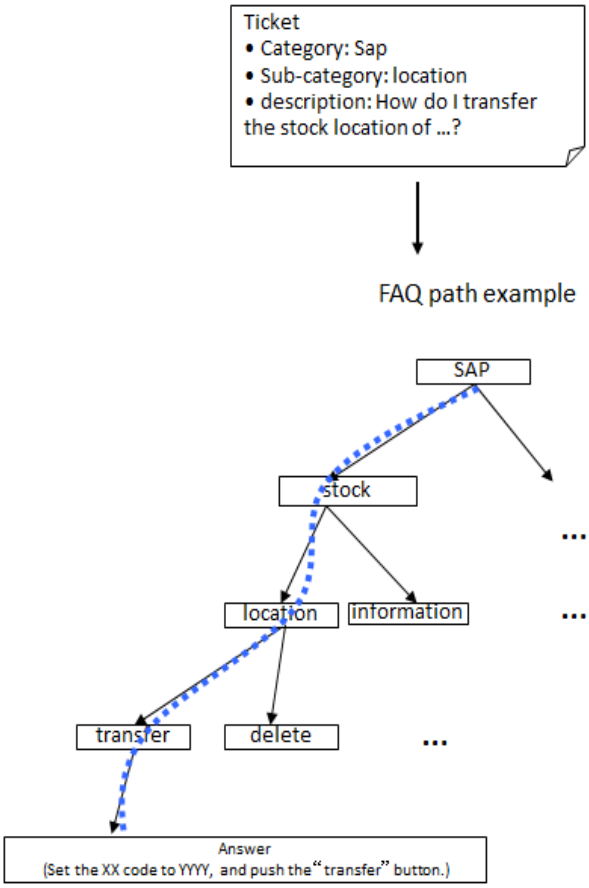

Fig. 5 FAQ Path Matching

3) Urgency Scoring

The urgency score is updated similarly to the corresponding "Initial Ticket Clustering" step. We observe that the urgency score for each cluster changes over time. Therefore, our method periodically recalculates the whole set of tickets for higher accuracy.

4) FAQ Formatting

Based on the ticket clusters and their scores the FAQ may need a new node an update of an existing node. To update nodes the practitioners use their areas of expertise to identify the nodes that need to be updated based with relevant ticket information. The practitioners are associated with their areas of expertise so the FAQ system shows the nodes to be updated in priority order (based on urgency).

III. Results

A. Conditions

We tested our technology in a project for a period of approximately two and half years. Table II shows the conditions of the project. (We are restricted on clarifying the client specific information of the project.) The size of the project was fairly large with a maximum of 60 practitioners working to resolve tickets. The number of practitioners have changed but peaked at 60 people. The domain was mostly restricted to SAP-related problems and the type of the tickets was mostly RFI (Request for Information) with estimated 1/4 RFC (Request for Change) tickets. The categories correspond to the names of the applications and we were able to completely sync the category and sub-category names with the FAQ structure.

TABLE II
Test Conditions

Item	Value
Total Period of Time	Approximately 2 years and 6 months
Period of Time before initial FAQ	Approximately 6 months
FAQ update frequency	Every 3 months
Total Number of Tickets	1,102
Domain of Tickets	SAP
Type of Tickets	RFI& RFC
No. of Practitioners	60 people (at peak)
No. of Categories	7
No. of Sub-Categories	64

For the creation and maintenance of the FAQ system, approximately 10 SME (Subject Matter Expert) practitioners were assigned by the project leader to each application (category), with multiple assignments for a few of them. The number of SME practitioners was limited to avoid inconsistency and the overhead of system usage training.

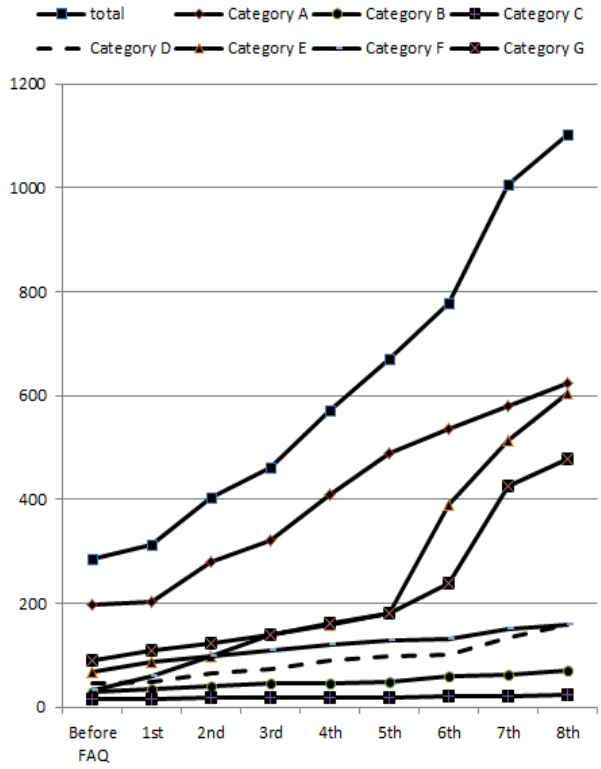

Fig. 6 Accumulated Tickets by Category

Fig. 6 shows the categorical and total accumulation of tickets over time. The x axis represents the quarter number starting from before the FAQ distribution and on to when the FAQ

become available. There was some variation in the numbers of tickets, with some periods of rapid increase. For example, the number of Category E tickets suddenly grew from the 5th to the 6th quarter. This was due to a new version of the Category E application, when user interface changes led to more inquiries. It is generally difficult to accurately predict the number of future tickets in advance.

B. Number of FAQ Nodes

The FAQ nodes were updated on a quarterly basis. Fig. 7 shows the total number of nodes in all of the layers by their status (unchanged/updated/new) for each quarter. The number of new nodes gradually became saturated as the knowledge in the FAQ approached a level of maturity completeness. At the same time, many nodes were updated frequently to reflect newly found knowledge and insights, while a large proportion of the nodes remained unchanged.

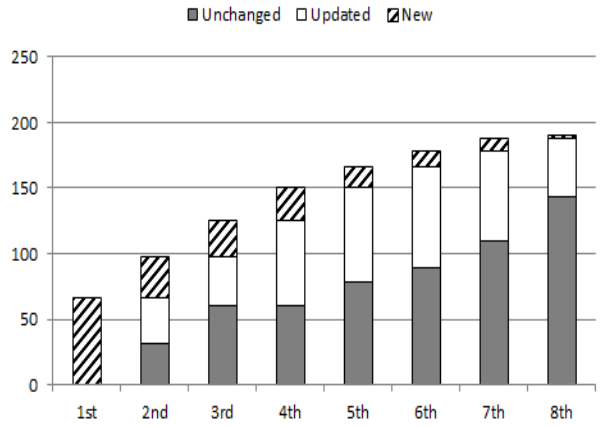

Fig. 7 FAQ Nodes

C. Ticket Coverage

The FAQ database was updated each quarter. The system tracks the FAQ node that was used for each resolved ticket. This data is used for Fig. 8, which shows the percentage of tickets that were covered by FAQs in each quarter. The solid line shows the overall coverage for the tickets.

We were also interested in how the coverage would decline if the FAQ data was not updated. It is quite difficult to accurately calculate the coverage of FAQs on the basis of "what if the FAQ stops changing" for an actual running project, since we cannot risk sacrificing the resolution time because of the damage to the business.

Our assumption is based on the record of which nodes were newly created, updated, or unchanged. First, we examined the coverage of the tickets by unchanged nodes over the study period. We also realize that updated nodes are difficult to analyze since the significance or the degree of the update may vary for each node and it is not possible to calculate the significance of each change. Therefore, we simply evenly treated the updated nodes as half unchanged and half new.

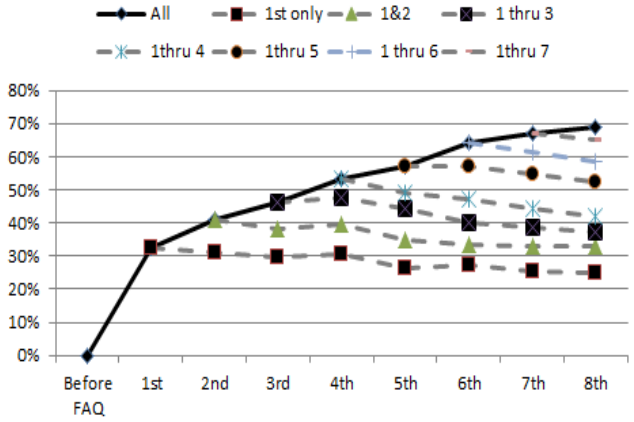

Fig. 8 FAQ Coverage of Tickets

Our results revealed that each quarter had declining coverage for the unchanged nodes (with the downward sloping dotted lines in Fig. 8). After the first quarter of FAQ availability, the total ticket coverage was 32.3% and after the 8th quarter it was 68.9%. This analysis makes it clear that the periodic updates to the FAQ data helped raise the ticket coverage, which would have declined without the updates.

D. Resolution Time

Fig. 9 shows the average resolution time for each quarter. The first 2 quarters did not have any FAQs and the resolution time was increasing and becoming a major problem. The initial FAQ helped reduce the resolution time significantly. As can be seen in Fig. 8 the coverage of tickets by the FAQ grew and we were able to keep the average resolution times short. Since ticket resolution is a human effort we know that there is a limit to how much the ticket resolution times can be reduced.

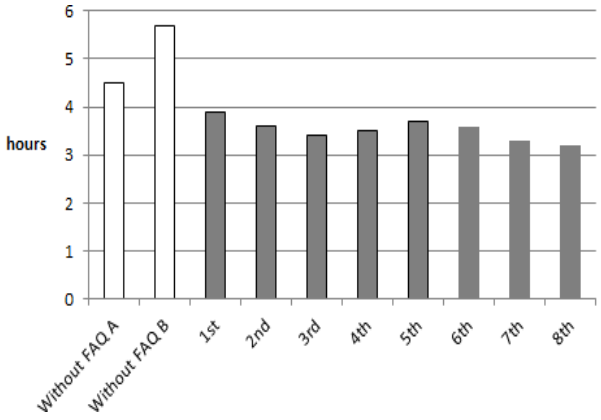

Fig. 9 Resolution Time

E. Subjective Evaluation

Though the user interface and the FAQ's usability are out of the scope of this paper, the practitioners' are the users of this system and we must recognize that their subjective evaluations are important. At the end of the project, we asked each practitioner to give us feedback on the effectiveness of this FAQ system. Fig. 10 breaks down the answers to the simple

question “Is this (FAQ) system effective in resolution time reduction?” More than 75% of the practitioners reported “strongly positive” or “positive” results, demonstrating the system’s effectiveness.

The dissatisfied practitioners were more carefully questioned about their concerns, and we found that they were the subject matter experts and had little need for the FAQ in their ticket resolution work.

We also received many comments throughout the project and considered them as carefully as possible. Most of the comments were operational questions that we were able to address by providing some basic assistance. Occasionally we received reports of system-level problems that required changes in the system code or configuration. Some simple usability issues were handled by updates to the user interface.

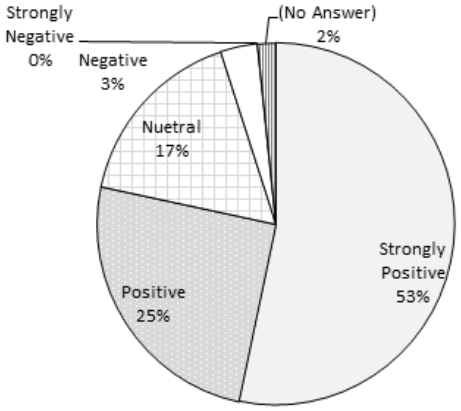

Fig. 10 Subjective Evaluation Result

IV. Related Work

The service incident ticket management system can be categorized as an instantiation of procedural knowledge [14]. Procedural knowledge is often described in the cognitive science literature as implicit knowledge about performable skills, such as walking or throwing a ball [15]. However, FAQs are for specific tasks defined explicitly and precisely SMEs, who must also be capable of describing the procedures.

Procedural knowledge in certain expert fields or areas has been studied in several different contexts such as cognitive science [15], artificial intelligence [16], enterprise workflow management [17], and domain knowledge searches for medicine and life science [18]. An intriguing application of procedural dialogs in the medical field was described by Bickmore et al. in [19]. In this study, they created a virtual medical assistant that engaged in a procedural dialog with patients to explain health documents. Their study indicated that patients with low health literacy were relatively more satisfied with the health document explanations from a virtual agent compared to a human being.

The concept of FAQ navigation is quite similar to the User Interface concept of Wizards that are commonly used in operating systems and many software packages, especially for installations. Perer and Shneiderman [20] described an extended wizard concept so users can navigate in applications. Their SYF system is a modular component that can be used to augment other applications with interactive guides.

Though many domains were identified where procedural knowledge can be applied, we did not find any research that tried to quantify the continuous effectiveness of FAQs in terms of time savings over time, or any work on correlating service incident tickets to FAQ usage.

There are NLP approaches to FAQ search [21]-[22], where end users can type in natural language sentences as the queries to search FAQs for answers. Basically, keywords in the query are matched against those in the FAQ. How the FAQs are searched and retrieved beyond our scope here, but our focus on search and retrieval is a hybrid approach of keyword matching from the query keyword and the text segmentation of FAQs.

V. Conclusion

Our past studies showed that a FAQ has a strong effect on ticket resolution time for a short period of time. Our objective in this new work was to evaluate if the ticket coverage, resolution time, and subject analysis can be continuously improved over time. Our project over 2.5 years showed effective improvements. Specifically, ticket coverage increased from 32.3% to 68.9% and the resolution time reduction percentage grew from 31.6% to 43.9%. Our subjective evaluation found that more than 75% of the practitioners found this FAQ system to be useful for ticket resolution.

One of the features we have not yet covered so far is to evaluate the user interface in terms of navigating to the answer in the FAQ. We currently support a hybrid navigation method where the user can search with dialogs or keywords. The concept behind this design is that the dialog-based approach is more useful for novice users, but the drawback of this approach is that it may require many interactions so that users can find it tedious. Keywords are faster, but require deeper prior knowledge of the solutions. In the future we want to quantitatively study the efficiency of our approach.

Another feature we have not covered is a best alternative node recommender, where the system will automatically find the best alternative node for any newly defined node. This newly defined node might be defined or derived from manual input, as obtained from keywords used in queries or in tickets. Our study shows that there are many cases where the alternative node can help guide the user to an effective answer.

Acknowledgments

We would like to thank all of the people who have participated in this project for so generously sharing their precious time with us. We would also like to thank many people at IBM Research who have given us valuable feedback in the technical aspects and in organizing this paper.

References

[1] A. April and A. Abran, “Software maintenance management: evaluation and continuous improvement”, Wiley-IEEE Computer Society Press, 2012.

[2] M. Ohno, K. Miyamoto and K. Yoshikawa, “Extracting works for offshore outsourcing through service request log analysis”, Frontiers in Service Science 2014.

[3] K. Miyamoto, T.Nerome, and T. Nakamura, “Document Quality Checking Tool for Global Software Development,” Service Research and Innovation Institute (SRII 2012), 2012, pp. 267-276.
[4] B. Orand andJ. Villarreal, “Foundations of IT Service Management with ITIL 2011: ITIL Foundations Course in a Book,” Create Space Independent Publishing Platform, 2011.
[5] T. H. Davenport and L. Prusak, “Working Knowledge: How Organizations Manage What They Know,” Harvard Business School press, 1998.
[6] W. Newk-Fon Hey Tow, J. Venable, and P. Dell, “Toward More Effective Knowledge Management: An Investigation of Problems in Knowledge Identification,” Pacific Asia Conference on Information Systems (PACIS 2011), paper 194, 2011.
[7] J. Skene, F. Raimondi, and W. Emmerich, "Service-Level Agreements for Electronic Services," IEEE Transactions on Software Engineering, Vol. 36 Issue 2, 2010, pp. 288-304.
[8] K. Miyamoto, R. Anand, and J. Lee, “Improving IT Service Incident Resolution by Using an FAQ System”, Service Research and Innovation Institute (SRII 2014), 2014.
[9] R. Anand, J. Lee, K. Miyamoto, L. Mei, and Q. Li, “The Dialog Manager: A system for managing procedural knowledge,” IEEE International Conference on Service Operations and Logistics, and Informatics (SOLI 2013), 2013.
[10] N. Indurkhya and F. J. Damerau, “Handbook of Natural Language Processing”, Second Edition, Series: Chapman & Hall/CRC Machine Learning & Pattern Recognition, February 22, 2010.
[11] “Statistical Methods. Least Squares, Kolmogorov-Smirnov Test, Design of Experiments, Optimal Design, Regression Analysis, Student's T-Test”, Booksllc. Net, 2013, ISBN 10: 1157454089
[12] A. Aizawa, "An information-theoretic perspective of tf–idf measures", Information Processing & Management, Vol. 39, Issue 1, January 2003, pp .45–65.
[13] K. Miyamoto, R. Anand, and J. Lee, “Automatic Updating FAQ Knowledge from Service Incident Tickets”, Frontiers in Service Science 2014. 2014.
[14] M. Demarest, "Understanding Knowledge Management," Long Range Planning, Vol. 30, No. 3, pp. 374-384 (1997).
[15] M. Schneider, B. Rittle-Johnson, and J. R. Star. "Relations among conceptual knowledge, procedural knowledge, and procedural flexibility in two samples differing in prior knowledge," Developmental psychology 47.6, 2011.
[16] M. G. Marchetta and Q. F. Raymundo, "An artificial intelligence planning approach to manufacturing feature recognition." Computer-Aided Design 42.3. pp.248-256, 2010.
[17] A. Lammer, S. Eggert, and N. Gronau. "A Procedure model for a SOA-Based integration of enterprise systems." Information Resources Management: Concepts, Methodologies, Tools, and Applications, 2010.
[18] N. Vibert, C. Ros, L. L. Bigot, M. Ramond, J. Gatefin, andJ. F. Rouet, “Effects of domain knowledge on reference search with the PubMed database: An experimental study.” Journal of the American Society for Information Science and Technology, 60(7), 2009, pp. 1423-1447.
[19] T. W. Bickmore, L. M. Pfeifer, andM. K. Paasche-Orlow, “Health document explanation by virtual agents.” In Intelligent Virtual Agents, Springer Berlin Heidelberg, 2007, pp. 183-196.
[20] A. Perer and B. Shneiderman, "Systematic yet flexible discovery: guiding domain experts through exploratory data analysis," In Proceedings of the 13th International Conference on Intelligent User Interfaces, 2008, pp. 109-118.
[21] D. Contractor, G. Kothari, T. A. Faruquie, L. V. Subramaniam, and S. Negi, S, "Handling noisy queries in cross language faq retrieval," In Proceedings of the 2010 Conference on Empirical Methods in Natural Language Processing, Association for Computational Linguistics, 2010,pp. 87-96.
[22] Y. Liu, C. Yin, H. Ogata, G. Qiao, and Y. Yano, "A FAQ-Based e-Learning Environment to Support Japanese Language Learning," International Journal of Distance Education Technologies (IJDET), 2011, pp. 45-55.

Kohtaroh (Cole) Miyamotor is a researcher at IBM Research – Tokyo. He resides in Tokyo but he has lived for more than 10 years abroad (UK and USA) so he is bilingual in English and Japanese. He has graduated with a master degree on Mechanical Engineering from Sophia University in 1990. Since then for almost a quarter of a century he has been working at IBM Japan. When he first joined IBM he started as a software developer and worked on development and project management for several software projects. In 2000, he moved to IBM Research – Tokyo (AKA Tokyo Research Lab) and started to do research on accessibility and human computer interaction. In this field he has been issued with many patents and has made some presentations at some distinguished international conferences. Recently he has changed his research theme to service research and his main interest is in cost reduction of service maintenance fields and front office innovations.