InfoFrax : CBR in Fused Cast Refractory Manufacture

Deepak Khemani [1], Radhika Selvamani B. [1], Ananda Rabi Dhar [1] , Michael S.M. [2]

[1] A.I. & D.B. Lab, Dept. of Computer Science & Engineering I.I.T.Madras, India
khemani@iitm.ac.in
bradhika@peacock.iitm.ernet.in
anandat@peacock.iitm.ernet.in
[2] Carborundum Universal Limited, Madras, India (CUMI)
mikesm@ho.cumi.co.in

Abstract. This paper describes a CBR application in manufacturing industry, a domain where CBR has by and large proved its applicability and success. The paper details a thorough understanding of the field of fused cast manufacturing basically seen from the perspective of glass furnace, where quality of glass produced is straightaway related to the refractory blocks used in furnace linings. The applicability of CBR paradigm is revisited in the present context. The CBR process needed is conceptualized and designed. The paper states in detail the evolution of the system starting from tackling hurdles of the knowledge acquisition and refining, facing a number of pitfalls in the prototype phase, to final implementation of InfoFrax, a CBR system specially devised for the project, and overall description of the same covering architecture, and usage. The paper also reports the immediate effects of the software in form of direct user feedback, expectations from the existing system and some directions of future work already underway in the project.

1 Introduction

Foundry related manufacturing often lies at the crossroads of art and science. This is because the behavior of materials at high temperature and during phase transition is poorly understood. The process of manufacturing involves a large number of parameters that have to be set on the shop floor. Often experience plays a crucial role in successful processes and in troubleshooting. We look at a case based reasoning (CBR) application in such a setting.

Carborundum Universal Ltd. (CUMI), Palakkad, India, is in the business of manufacture of fused cast (electro cast) refractory blocks used as lining for glass melting furnace tanks. This line of manufacture is such that the technology is unique, not freely available in the world, specific to the application and the equipment, and is not fully mastered yet [11]. A number of cases have been outlined where the observations made by the operating personnel have changed the basic principles of design and manufacturing practices because no definite knowledge base is available in either textbooks or journals. Each of these cases, which are production activities, is by itself an experiment and the industry will have to build on these experiments to improve continuously. This is ideally suited for an application of the case based reasoning system [9]. [1]This project described in this paper involves setting up a CBR system on the floor, collecting operational data, and maintaining a repository of all kinds of blocks. Details for each block manufactured, good or bad is added to the memory along with the corrective action taken if any. When a defective block is encountered the CBR system retrieves the best matching corrective action. With sufficient data accumulating, the system will, in the future, be in a position to support predictive maintenance and even trigger changes in the Design Handbook. It demonstrates CBR applicability in domains where analytical knowledge is incomplete and yet experienced personnel can make decisions that succeed.

2 Domain

Refractory blocks are manufactured in the electrocast foundry. The first evaluation is done by quality control at the end of the manufacturing process. The block may be accepted wherein it becomes a success case. Or it may be rejected wherein it becomes a failure case and it is remade with a change in parameters, mostly suggested by experts. The blocks are thus labeled O.K., or rejected by the Quality department. Following acceptance, they are put to use in lining glass furnaces, where they are subjected to relentless corrosion at high temperature over a period lasting tens of months. Glass manufacturers place orders on CUMI for lining their melting tanks. The lining is made up of around 300 to 400 blocks depending on the size of the tank.

2.1 Refractory Manufacture

Refractory blocks are made of a mixture of Alumina, Zirconia and Silica in certain proportions depending on the location of the block in the tank. The mixture of these raw materials are melted in an arc furnace and poured into moulds, and annealed. The blocks are subsequently removed from moulds and finished. Fig. 1 shows the production process along with a few illustrative parameters. The common technical defects, which lead to rejection of blocks, are cracks, spalls, and surface porosity. The average weight of each block is around a tonne and the cost of materials and production is quite high. The manufacturing lead-time is almost a month considering the long annealing periods of almost a fortnight. Hence rejections are quite costly in terms of material and time.

Design activity precedes the manufacturing activity. The design department of CUMI has a design handbook, which gives the design attributes for the manufacture of the blocks. The design handbook gives guidelines, which can be modified if failures are traced to specific attributes.. The process of changing the design handbook is done very formally based on a corrective action record (C.A.R.), which records the solutions provided by experts in case of rejected blocks which were remade.

---

[1] This project is funded by a research grant from Carborundum Universal Ltd. Madras, India (CUMI)

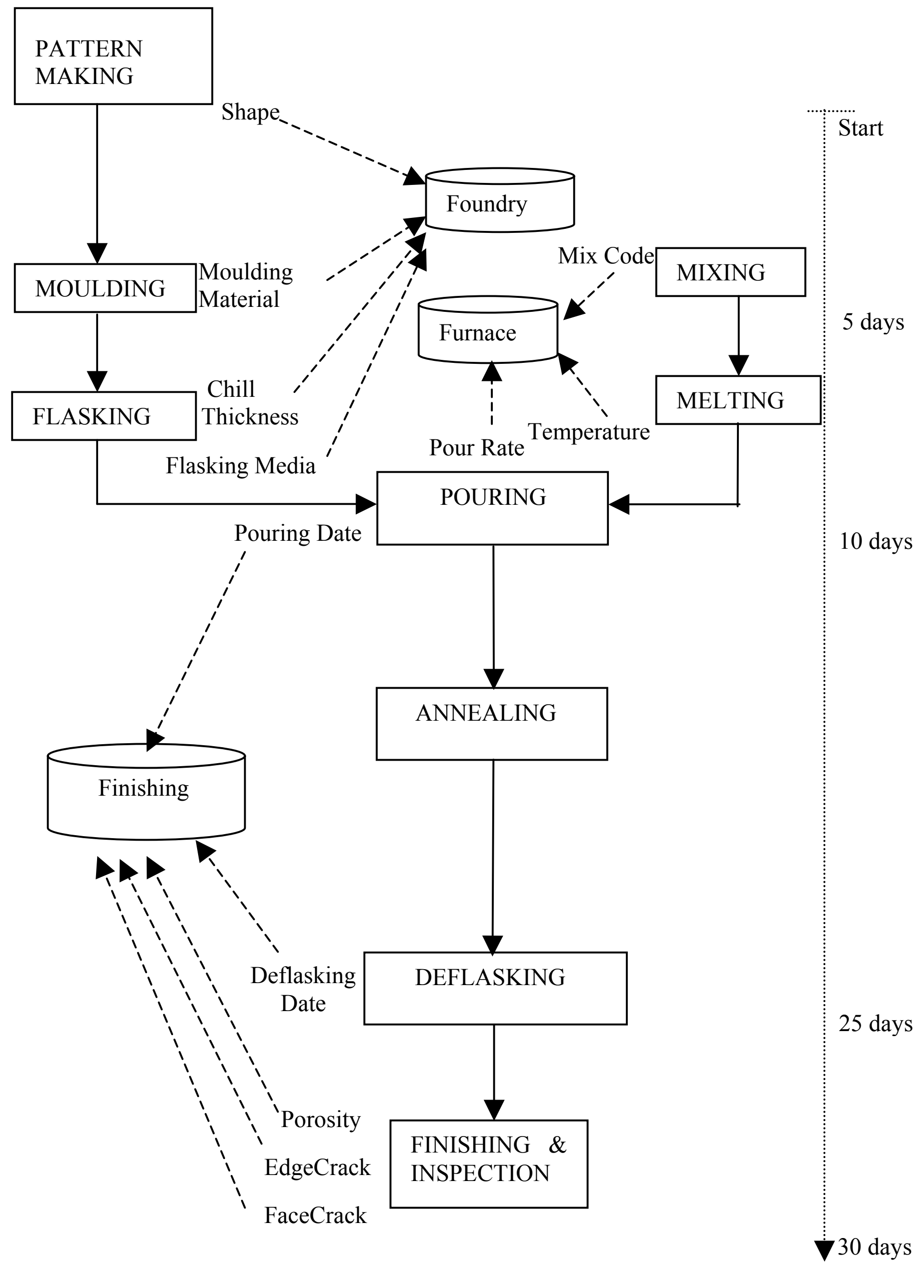

Fig. 1. Production Process and Data Collection

2.2 Glass Manufacture

Glass manufacturers melt a charge mix in a furnace and draw out molten glass to be formed into different types of containers or sheets. This is a continuous process, which means that the furnace should be operating on a 7X24 basis. The furnaces have a life of 4 to 8 years depending on the type of glass drawn. Minor problems occurring in the furnace or the auxiliary structures are corrected while the furnace is in operation. These repairs are called hot repairs. The production loss is not too severe in these types of repairs. However, any major problems would require a cold repair, which would mean a shut down. A shut down and restart operation would take 8 to 12 weeks. This type of furnace stoppage is one of the reasons for production loss. Another type of loss is due to the glass defects. The glass could have defects such as bubbles, foreign particles or scratches, which leads to rejection and production loss. These defects in glass are sometimes attributed to the components of the furnace including refractory blocks. Alternatively they would be due to incorrect operation of furnaces. The parameters related to the operation of the furnace viz. the temperature,

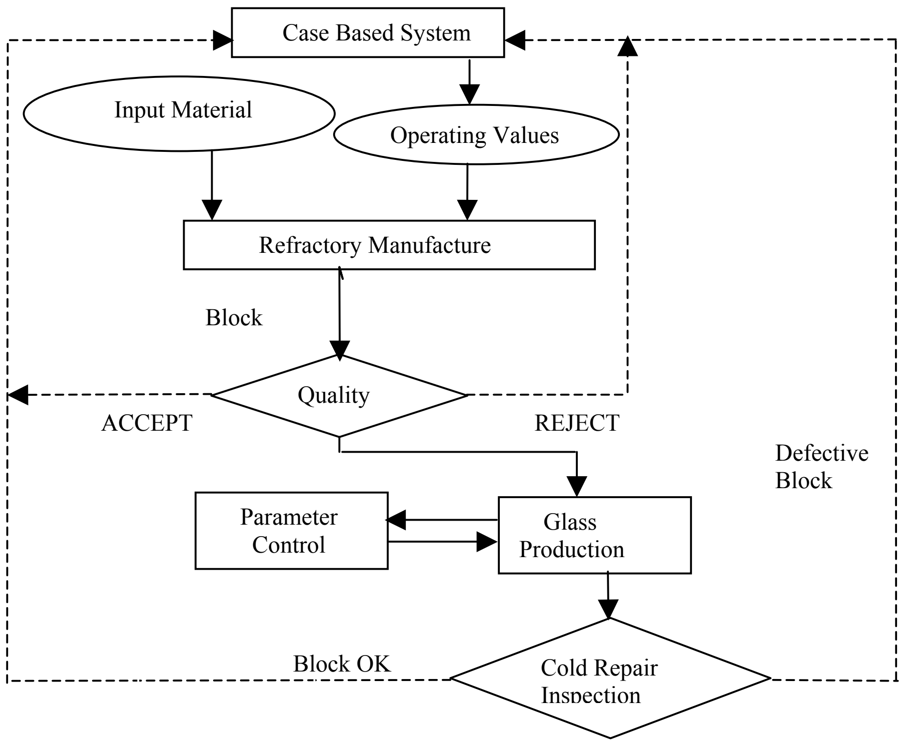

Fig. 2. Role of CBR in Refractory and Glass Manufacture

the pressure, charge mix composition, consumption of fuel per ton of glass drawn, temperature of inlet air and exhaust air, draw rate, etc. and the problems encountered as mentioned above can be recorded as case description.. However, if such a CBR system is to be developed there will be a need for data to be collected for many furnaces over a long period cycle. The work discussed here is restricted only to build a CBR system which will aid in solving quality problems faced in manufacture of the refractory.

In a CBR system that covers the life of a block one would have to capture the history of each block as it is manufactured and also its performance in the furnace. Over a period of time a corporate memory that captures the experience of the factory in manufacturing refractories and the performance in the furnace will be built up. The main aim is to build a system that allows the direct reuse of experience without trying to do a technical analysis. This is depicted in Fig. 2 as integrated CBR cycle for both Refractory and Glass Furnaces.

3 Why CBR?

The problem in context resembles a CBR paradigm, which could be counted for some basic reasons. The manufacture domain is ill understood and it is not easy to articulate the knowledge in the form of rules. The time line in Fig. 1 illustrates the long process which involves a large set of attributes dispersed in different shop floors. If the parameters for every block manufactured are stored in a database, how does one exploit this data? Traditional database retrieval requires a precise match of values of attributes. One can relinquish precision and use ranges, but this would lead to over abundant recall. This is because a priori there is no knowledge about which attributes should be relaxed. CBR allows for flexible relaxation because it uses a notion of similarity. It offers much more precision, but this comes at a cost sacrificing the efficiency built into database system.

4 Domain Acquisition and Prototype

4.1 Traceability of a Case

Many experts opine that the mould plays an important role in determining the success of the block. But the mould is destroyed and it loses its identity when the fused melt is poured in at a temperature of 1700°C. The relationship of the attributes of the mould to the block is lost. This poses a problem in uniquely identifying the attributes of a manufactured block with respect to the mould from which the block was manufactured. This is achieved by adding common attributes with the furnace data base schema to facilitate a correct join operation. A similar link has to be established as the block moves from the furnace to the finishing department.

4.2 Incomplete Domain

Shop floor experts decide on solutions based on past experience and reasoning when faced with a problem. In the design of a case structure the need is to take into consideration as many attributes as possible. The influence of these attributes on the final product cannot be clearly determined. The experts agree that all the attributes are important, but are not able to express in unambiguous terms their influence. Hence it is required to capture all attributes, even those, which seemingly do not contribute to the quality of the block.

Often the defective block is remade with no modifications and the block turns out to be good. There is need for statistical analysis to support basic process modification. The process of changing the design handbook is done very formally based on a corrective action record (CAR) when the corrections are validated to be successful. However, the description of the defects and solutions as recorded in the C.A.R. book currently are too general to be reused. Also the design knowledge has grown over the years and is not structured. There are increasing number of exceptions to the guidelines.

4.3 Large Number of Attributes

CLAVIER system, of Lockheed Missiles and Space Company [5] uses CBR effectively to capture the successful patterns of placement and helps replacing similar parts to be cured. Domain experts were able to identify position in the furnace as a crucial feature, thus helping design an appropriate case schema. However, in Electro Cast manufacturing process the number of attributes generated in each stage of manufacture is quite large and posed a challenge for description of a block with all its associated features for the whole manufacturing process. The number of attributes is about 150, which is significantly larger than most implemented systems.

4.4 Approach to Solution

Knowledge acquisition can be viewed as the process of extracting, structuring and organizing knowledge from one or more sources. This process has been identified by many researchers and practitioners as a (or even as the) bottleneck that currently constraints the development of Knowledge Based Systems [4].

There are three types of features involved in successfully describing, retrieving and identifying a case. These are the operational parameters used in production, the quantified quality inspection results, additional parameters needed to join data from different departments. During the knowledge acquisition phase of InfoFrax, all the parameters as involved in the process of manufacture were noted down from operating personnel and other documents available. The experts then identified the parameters that were important according to their experience and knowledge. From successive interviews and analysis an initial collection of parameters was constructed. These production parameters constitute the explicitly relevant features of a case. But

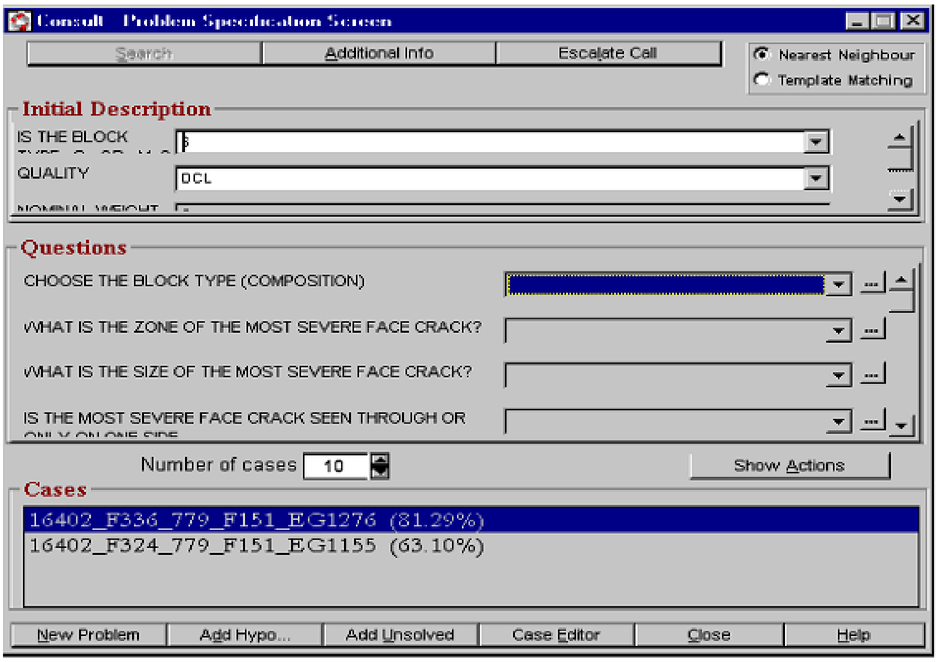

Fig. 3. Screen Shot of Case Retrieval in Consult^{TM}

the case structure could not be manually validated because it was vast and distributed among various process shops. This necessitated the prototype phase.

The prototype phase [8] had two components. The main goal of constructing the prototype was demonstrating the idea of case based retrieval from the database for troubleshooting.

Data acquisition is not a one-time activity. So to ensure validity of the data actually available an initial data capturing system was installed. Based on this exercise data capture sheets were designed for operating personnel. These were necessitated primarily because the process was physically removed from the data entry points and tracking was difficult. The analysis of the data collected in addition to knowledge refinement also made obvious the importance of some of the features. For this prototype phase an in house developed data capture software was used.

The principle of CBR, which the manufacturing expert does not visualize is that a case will be retrieved even if there are missing attributes or when some attributes do not match. With partial number of attributes for a complete case being used in the search, closest matching cases are retrieved. At no time will the CBR not retrieve a case while searching for a matching case, as against the principle of database, where no data will get retrieved when there is no exact match.. For this prototype case an off the shelf CBR package Consult^{TM} was used [1]. The version of Consult^{TM} supported only cases with 30 attributes. However, it was felt that a buy-in with the management would be obtained by demonstrating the prototype with 30 attributes before developing software to handle 150 attributes. This meant that the knowledge engineers had to group a few attributes to represent a case. In Consult™ [Fig. 3], which is an interactive system, the attributes may be of either dynamic or static type. The static attributes will be present in all cases, whereas the dynamic attributes may or may not be used in a particular case. For instance, some of the most important attributes of a case will be the defects, but all blocks will not have the same number or type of defects. The nearest neighbour match [2] is used to retrieve the best matching case.

5 Design of InfoFrax

Values for the case attributes are collected in the various shop floors and assimilated into a record in the block database. To exploit the use of a RDBMS package to store and retrieve cases a standard record schema was designed. This record schema includes attributes for all the possible defects that could occur. For example up to six face cracks could be described using a qualitative defect vocabulary that was defined. Similarly other kinds of defects have slots too. This resulted in a case structure with about 150 attributes. The first task was to verify that such large cases could be handled effectively.

5.1 Feasibility Check

The simplest form of nearest neighbour algorithm would require comparing an input case with all existing cases. The case closest with respect to some distance measure is selected. A feasibility check was done to retrieve and match 2000 cases of 150 attributes, to verify that the response time will be within acceptable limits. The implementation is on Windows based Pentium Ш PCs.

5.2 Case Structure

The case structure constitutes of attribute value pairs of the following kinds.

1. Production attributes: The parameter values used in the various stages of production extracted mostly from the mixing, foundry, and furnace shop floors.
2. Quality assessment: The data related to the actual block produced, including description of defects, and inspection conclusions.
3. Administrative attributes: Information related to customers, design documents, and labels used for tracking the block through various stages.

The case structure is stored in a schema file starting with a serial number followed by a name uniquely identifying the attribute, the type of the attribute, the enumerated values or the range of values depending on the type, and ending with the weight assigned if it is a match type. For numeric attributes the possible range is initially given by the domain experts, letting the system dynamically expand the range whenever a value encountered lies outside the so far decided range. The match function scans the file to get attribute type and weight. The similarity over integers or

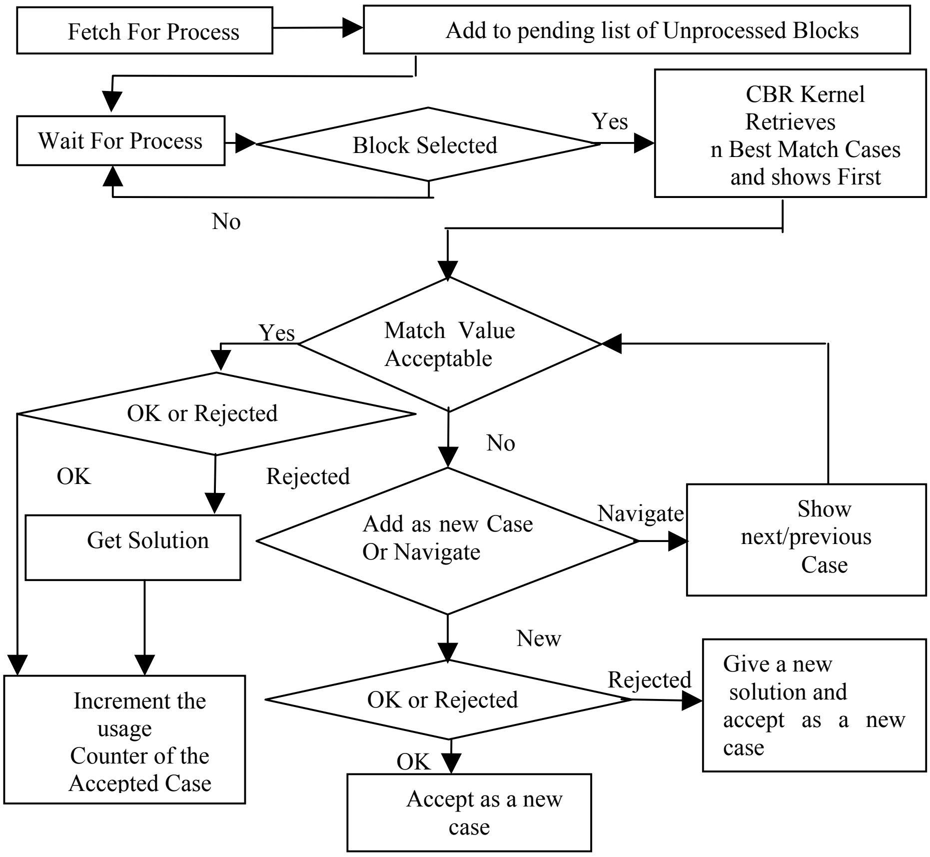

Fig. 4. CBR Process Flow chart

ordered enumerations is computed using the following formula. Let Ij be the input value for the jth attribute Aj, and Vj be the value for the same attribute in the case to be matched and Rj be the range of the particular attribute, then local match for the attribute Aj is given by sim(Vj, Ij) = 100 – (100 * abs(Ij-Vj) / R). Unordered enumerations are exactly matched according to the value.

5.3 Case Retrieval

After a block has been produced, and a block data record is assimilated the CBR phase is invoked. The CBR kernel sequentially compares the input block with all the cases in the database, and constructs a temporary database of n best cases. If the block has been rejected and there is no previous case of that type it is added to the case base with the corrective action as solution, which may be retrieved later on to get solution for a similar rejected block, which needs to be remade [Fig. 4].

5.4 Linear Search

The retrieval algorithm that is used in the CBR prototype is a simple sequential search irrespective of the current case and its complexity. The index chosen is a primary index automatically being generated synchronously with the knowledge acquisition phase entrusting an RDBMS with the task of indexing, sorting, and retrieving. Two different arguments are there to support the preference of a similarity computation on a linear case base over an indexed search on a structured case base. Firstly, the number of features related to each case is quite high and secondly, the domain experts are ambiguous about the relevance of the features, thus making any structuring on the case base costly and inefficient. Also tuning of the case retrieval is made easy by just allowing weight adjustment in the attribute schema, which otherwise in a structured case base would have needed a complex strategy.

5.5 Visualization

The user is shown the input block data and the best matching cases juxtaposed on the screen with each attribute showing values for both [Fig. 5]. A color coding scheme is used to highlight attributes with mismatch between the input block and the retrieved case. The lower the local match for the values of an attribute, the more strikingly different a colour is used to highlight the values. This directs the user's attention quickly to attributes where the case does not match the block. The user can then navigate back and forth through the n retrieved cases and decide if any of them fits the bill. If it does, the block is added as an instance of the case, and a counter is incremented for statistical purposes. If not, the block can be stored as a new case in the case base. If the block being processed is a rejected block, the system ensures that a corrective action for the block is prescribed and entered before proceeding further.

5.6 Weight Adjustment

The retrieval of cases is based on matching by the 'k' nearest neighbor algorithm [2]. Let the weight schema be represented as $W=\{A_1 W_1, A_2 W_2, ..., A_n W_n\}$ where A_1, A_2 and A_n are attributes and the W's are the associated weights. Let the input problem's match attributes with values be represented as $C_i = \{A_1 V_1, A_2 V_2, ..., A_n V_n\}$ where the V's are the associated values. Let C_y be one of the cases in the case-base and which can be represented as, $C_y = \{A_1 Y_1, A_2 Y_2,...., A_k Y_k\}$. Then the weighted 'k' Nearest Neighbor computes the similarity between C_j and C_y, S_{jy} as,

$$S_{jy} = \sum W_j * \text{sim}(V_j, Y_j) / \sum W_j \quad j = 1 \text{ to } k$$

i.e. the similarity over each attribute is computed, weighted, summed up and finally divided by the sum of weights to get the normalized similarity. The contribution of each attribute is proportional to the weight. The case author decides whether or not the block being processed is an instance of a stored case. Ideally the match function should agree with the human by giving a suitably high score to the selected case.

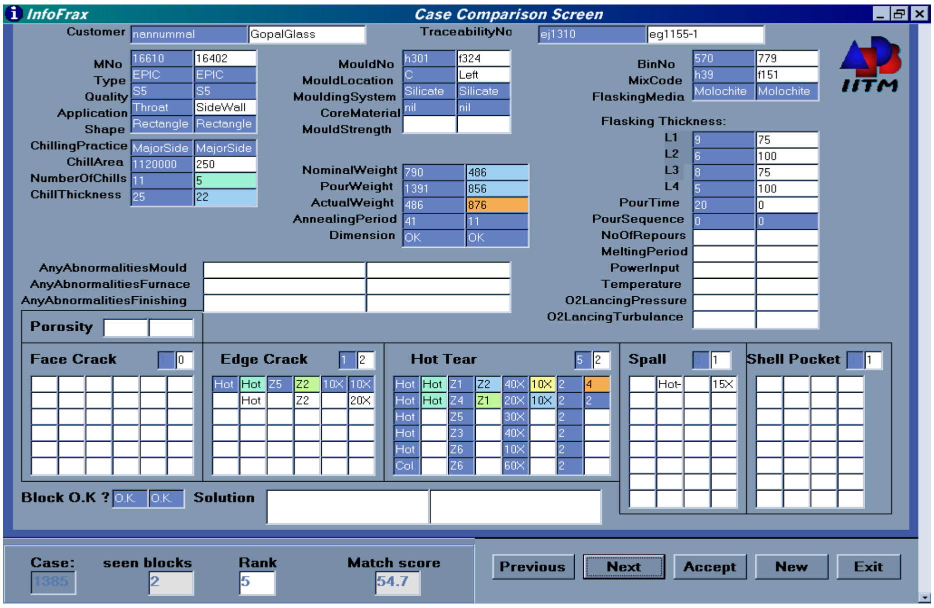

Fig. 5. Case Comparison Screen in InfoFrax

Whether or not this happens is influenced by the weights assigned to different attributes. The current implementation of InfoFrax provides a weight changing functionality. If case author feels that the system consistently disagrees with him on the best match case, and the match value, he can increase or decrease the weights of some attributes. The colour coding used in visualization can help him decide which weights to increase weights and which to decrease. The match function normalizes the sum of all weights to 100.

Alternatively there is a clear case for parameterized learning of weights. When a user selects a case not ranked first by the system, one can use the information to adjust weights. Again the local match plays a role. We plan to implement a weight learning algorithm in the next phase of implementation.

5.7 Architecture.

The three major subsystems in InfoFrax are the Data Capture sub system, Information retrieval sub system and the kernel [Fig. 6]. The data capture system is the first to be implemented for initial data validation and is responsible for capturing the manufacturing data in three major process shops namely the foundry furnace and the finishing. These screens need to be user friendly because the major data input in the system is handled by the system. Data entry for all symbolic attributes, and most of the numerical attributes is through drop down windows to avoid errors. In addition, an attempt has been made to ensure that each data item is entered only once. Complete and consistent data entry is ensured by an interlocking mechanism among the screens in different shops.. The information retrieval system and the kernel, are responsible for the storage and retrieval of the manufacturing experience. Though developed and used in parallel they have been implemented separately for the following reasons. The information retrieval system interacts with the user and facilitates easy access to the required knowledge. For instance it lets the user decide which block is to be processed and displays the best matching cases for the particular block with the match score.

Case maintenance features for adding new cases or updating the usage of the existing cases is also provided in the screen. The kernel implements the algorithms required for searching the case base and retrieving the best cases. Speed is the primary requirement of this sub system because analysis identified the database access operations to be the major cause for delays in retrieval of the whole system. The kernel is shielded from users by the information retrieval system, which also synchronizes the data transfer between itself and the kernel.

The current implementation uses Visual Basic on top of MS Access. The kernel is written in VC++.

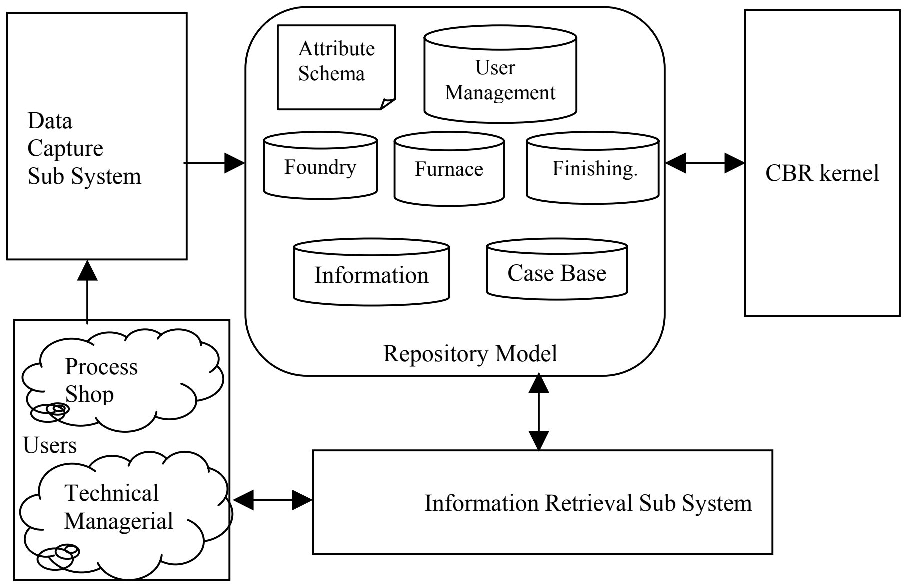

Fig. 6. Software Architecture

6 Deployment

The users of the system exist in various organizational levels of the company. Shop floor people are responsible for the data entry operations. The technical and managerial persons are the consumers of the information stored in the system. The case base is exploited in two ways. Firstly, to find a best matching case for a manufactured block. Its main utility is to help decide upon corrective action in case of rejected blocks. Presently it is done without any adaptation. Secondly, in a design mode to look at past parameters and to produce blocks with some given specifications. A single case author usually is allowed to maintain the case base with new cases or usage information for each block manufactured. Both positive and negative cases are stored as experience.

Table1. Example of a Case Compare in Tabular Form

Case	Type	Quality	Block	NomWt.	PourWt.	L1	L2	L3	L4	%
Input	EPIC	S3	Reject	534	670	5	4	6	5	
Case1	EC	S3	Reject	502	653	16	13	16	14	89.81
Case2	EPIC	S3	OK	534	668	9	4	7	10	88.72

6.1 User Feedback

The three modes of InfoFrax are Data Capture, Case Comparison and Design Consult The end users have benefited by being able to see all the 150 parameters associated with each block and comparing with similar blocks aided with graphics aids. The design experts in the domain have reported that they are now ready to consider using the software for development inputs for manufacturing newer type of blocks. The design experts have been able to identify additional uses of the intermediate block database to aid them in specific analysis, such as to find out how many blocks have had a particular type of defect. The marketing department is more confident that they can approach their customers with value added service packages along with the sale of refractory blocks such as lifetime guarantee on the performance of the blocks in the furnace. They have also claimed that it will help them in resolving conflicts with the customer when grappling with problems, which can be due to parts, which they have supplied vis-à-vis the components supplied by others. However, in the refractory and glass manufacturing industry any initiatives such as this will take a few years to yield results because the life of the furnaces is between 4 to 8 years [7].

7 Conclusions and Future Work

InfoFrax is now able to successfully capture the huge amount of data prevailing at various distributed process shops thus initiating the first step in knowledge creation. The CBR methodology has been integrated with the knowledge capture providing a tool for the factory personnel to use the huge amount of data captured and organized in the first step. Since case based expert systems, unlike other systems that need great effort and technical assistance in the phase of maintenance, are easy to maintain, the knowledge maintenance is also done effectively and easily. In addition the experts are allowed to tune the CBR system using the weight manipulation facility,. There is a rule-based system providing the necessary design knowledge for the manufacture of the blocks. The CBR stores the manufacturing experience and provides solutions in case of failure after manufacture. It is expected that as cases in the case base accumulate, the design rules will be influenced via knowledge discovery methods.

7.1 Outcome of Solution

At present a case comprises a problem description and solution. The solution proposed in case of rejection is always remaking the block with either unaltered or altered settings. If the outcome or effect of the solution is captured in future cases as soon as a block is remade, it could potentially help the user decide upon the solutions for the blocks having similar problems, which are to be remade in future.

7.2 Multiple Views

Weight tuning though handled by experts may suffer from wrong assessment due to personal choices and incomplete knowledge. The problem can be minimized by considering the opinions of different experts in different process shops. It could be possibly done by allowing the case author to visualize all views and consequences in the form of a one screen-display of multiple cases retrieved using multiple similarity assessment suggested by different experts. For example, by magnifying the weights for a particular department their viewpoint will be emphasized.

7.3 Modified Representation

Currently the first block of a kind encountered becomes the case prototype. However this could be inaccurate. Assuming that conceptually a case defines a region in the n dimensional attribute space, the first instance being near the periphery of the region would lead to an unsuitable case representative. This is crucial because nearness is computed with respect to the representative. One direction of future work is to work towards a representation that is cumulatively defined by all similar cases encountered.

7.4 Improving Quality

Improvement in quality will be possible with deployment of CBR in predictive maintenance by further discriminating amongst successful cases based on quality related parameters. This may require a second stage of matching with a weight schema that gives more importance to quality parameters.

7.5 Concluding Remarks

There was an initial skepticism amongst production managers towards the proposal of constructing a knowledge based expert system in a domain they viewed as complex and ill defined. But the case based reasoning approach has successfully demonstrated a technique of harnessing knowledge and experience accessible to the collective workforce. Nevertheless, as we have seen in this section, the installed system can be seen to be a basis on which further systems can be built. The repository that is generated will be a resource for developing other functionalities and exploring the ways of exploiting the information. We feel that any living system of this kind opens up a new horizon for all levels of users, developers, and theoreticians.

8 References

[1] Balaraman, V. and Vattam, S., " Finding Common Ground in Case-Based Systems", In *Proceedings of the International Conference KBCS-98, India.*. Sasikumar, M.: Durgesh Rao.: Ravi Prakash, P.: and Ramani, S. eds. 1998.

[2] Balaraman, V., Chakraborti, S., and Vattam, S. , "Using CBR Inexact Search for Intelligent Retrieval of Data", In *Proceedings of the International Conference* K*BCS 2000.India*. Sasikumar, M.: Durgesh Rao and Ravi Prakash, P. eds. 2000

[3] David B. Leake, *Case-Based Reasoning: Experiences, Lessons, and Future Directions*, Menlo Park: AAAI Press/ MIT Press, 1996.

[4] Efraim Turban, *Expert Systems and Applied Artificial Intelligence ,* Macmillan, 1992

[5] Hennessey, D. and Hinkle, D. "Applying Case-based Reasoning to Autoclave Loading", *IEEE Expert*, Vol. 7, Num. 5. 1992. pp. 21-26

[7] Mike Nelson, "Recent glass refractory application trends", *Glass Industry Development International* 2002, pp.55 - 57

[8] Michael, S. M., and Khemani, D., "Knowlwdge Management in Manufacturing Technology", *In Proceedings of the International Conference on Enterprise Information System 2002,*Ciudad Real,, Spain, Vol. I, pp. 506-512

[9] Rissland, E.: Kolodner, J.: and Waltz, D. 1989 Case based Reasoning. *Proceedings of the DARPA Case-based Reasoning Workshop*, 1-13, San Mateo, CA,: Morgan Kauffman

[10] Sivakumar A., & Ravi Kannan. In *Justification of the Project*, Internal memo, CUMI Chennai, 1998.

[11] Srinivasan, R. "Refractories for Glass Tank Furnaces: Exudation Behavior of AZS Refractories", *Society of Glass Technology, Indian Section, Quarterly Newsletter*, 1, Vol.V Number 2. 1998.