CHAPTER 4

CASE-BASED REASONING

The working phenomenon of the supply chain management system is multifaceted and dynamic by nature. All entities of the SCM carry out their roles and responsibilities in the SCM operation. These activities occurred in SCM operation consists a variety of decisions at all stages. Today the trade activities are reliant on a worldwide economic, equipped and business infrastructure. With the increase of complexity of SCM activities it requires more innovative decision taking mechanism. The decision-taking process is directly oriented on reasoning and learning from the environment for managing the market risks. For this purpose we choose the case-based reasoning approach to construct a complete supply chain system to recognize, supervise and construct well-versed decisions to diminish disruptions to businesses sand consequential impacts.

4.1 WHY CASE-BASED REASONING?

The case-based reasoning approach consist the library of past cases which consists the description of the problems and their solutions. The conventional reasoning approaches are not proficient of utilizing the precedent experience of solving existing problems. Hence these past problems’ solving experience is completely exhausted. This experience is not skilled of guiding the managers to obtain the resolution of forthcoming problems. The case-based reasoning is found more suitable with following reasons as given below:

- Cases of earlier solved problems exist.
- Historical cases are viewed in the future for solving new problems.
- Recalling existing experiences is practical in deriving new solutions.
- The proposed solutions are being referenced with existing examples.
- The experiences are more valuable in comparison of stored rules.
With the help of these facts, the case-based reasoning is looked as more useful in supply chain management planning and risk management. The CBR is being considered as the very powerful tool in supply chain management system. It reassembles as the human being reasoning.

4.2 CASE-BASED REASONING

The case-based reasoning is more emerging problem solving approach oriented on the utilization of past problem solving experience. This approach maintains the case base consisting of multiple cases. The case includes the following details as given below:

- Description of the problem
- Justification of the problem
- Result of the solution
- Solution of the problem.

The description of the problem consists multiple attributes having a specific value (Integer, Double, String and Boolean). The conventional system used the concept of knowledge-base system. The knowledge-based system (KBS) may be defined as the computer program that competent of rationale and utilize the comprehension to resolve composite problems. KBS acquires the knowledge and represent it using a variety of information illustration techniques. This type of system consists basic components as given below:

- Knowledge base contains the information regarding the environment.
- Acquisition mechanisms provide the mechanism for acquiring knowledge.
- Inference mechanisms handle the way of firing the rules.

But this approach is not as efficient due to the following reasons as given below:

- The process of knowledge elicitation is difficult by nature that is known as the knowledge elicitation bottleneck.
- The KBS implementing process requires extraordinary skills and complex by the natures.
- The KBS are not capable of accessing or managing large volumes of information.
- The speed of accessing the information from the knowledge is very slow.
- The KBS are difficult to maintain and modify.
These problems are being resolved by using the case-based reasoning approaches. The CBR maintains own case base having the multiple cases having the information regarding the solution of the existing problems. This approach is directly used in many real-time problems as given below:

- The tutors (online or offline) utilize the memory of past questions’ answer. During its operation, the new question is being asked then the answer of the similar question is being utilized to provide the answer during the coaching.

- The physicians keep the record of every patient in his hospital. In these records the physicians maintain the information about their patients’ diseases and the treatments is being followed. On arrival of the new patient he summaries major symptoms and find the similar case for diagnosis purpose and start the treatment for the patient.

- The drilling engineer utilizes the experiences of impressive nosh-up situations. He reminds the decisive dimensions matches for blow out case. Remembering these previous matches it helps to avoid the mistake made during a previous blow-out once again.

- The financial consultant uses the memory of previous troubles faced in performing tricky credit conclusion task in the company. For this purpose the consultant reminds the past experience of similar trouble regarding with current loan application. These experiences are being refused in handling the new loan applications.

- The travel recommender maintains the collection of the suggested travel packages to various tourists. They suggest them different type of packages and guide the appreciated tour package regarding tourists’ needs. He suggests the similar tour package on discussing the tourist through reminding the past tour packages suggested.

All these real-time applications show the usages of the case-based reasoning. This approach enhances the utilization of case libraries. This approach affords straightforward manner of the knowledge acquisition phenomena. It is very easy to expand and maintain the knowledge base with the times without propagating the past errors occurred in the
past. It resolves the problems in also known domains and supports the learning facility during problem solving [94].

4.3 CBR CYCLE

The case-based reasoning approach solves the problem by following the schematic cycle as shown in figure 4.1. Aamodt and Plaza et al. defines following four REs in the CBR cycle as given below:

- In Retrieve phase, the similar cases are being found on the basis of the similarity functions and retrieved with regard the new problem from the case base.
- In Reuse phase, the information stored in retrieving cases are being reused to derive the suggested solution to solve the problem;
- In Revise phase, the suggested cases are revised and evaluated regarding the constraints of the upcoming problem to generate the proposed solution. This phase analyses the suggested solution to prevent the propagation of the errors in the next phase.
- In Retain phase, the proposed solutions are being stored in the case base for utilizing the past experience of solving the future uncounted problems.

![Figure 4.1 CBR Cycle](image-url)
As shown in figure 4.1 the CBR cycle complete with 4 REs (Retrieve, Reuse, Revise and Retain). There is no need of any interference from the users. It helps in the decision-taking process in the decision support system.

4.3.1 Components of CBR system

The basic propose of the CBR system to support the decisions through the past experience. The CBR system contains the following components as given below:

- **Case base**
  
  This Component maintains the past cases of problem solving. With the help of the indexing techniques the CBR system stores the cases in case base and enable the fast way of accessing cases during its operation.

- **Case retriever**
  
  This component applies various types of the similarity functions to retrieve similar cases. The nearest neighbor algorithm is frequently applied for judgment of related cases from case base.

- **Case reasoner**
  
  This component performs the adaptation task to the retrieved case with a new case to generate the proposed solution. Then it performs the revision phase to prevent the error propagation in the final solutions.

The Case retriever and Case reasoner together design the case-based reasoning mechanism to construct the derivative solution. There exists the interactions between CBR components shown as in the figure 4.2.

![Figure 4.2 CBR Components](image-url)
It may be considered as a black box on the top level of abstraction that interacts with following exterior facets:

- New problem may be measured as new cases.
- Final Solution after revision with constraints of new problem.

The case-based reasoning mechanism is internally divided into two parts- case retriever and case reasoner. This reasoning process normally involves mutually calculating the differences sandwiched between retrieved cases and the recent case, and updating the resolution to imitate these differences properly.

### 4.3.2 CBR tasks

From the retrieve phase to retain phase the case-based reasoning system perform multiple functions for the purpose of deriving the final solution for upcoming problem. These functions or processes may be known as the CBR task. These CBR tasks perform specific tasks in the described its cycle. The CBR task is performed by applying one or more methods for achieving its goals. To carry out a task, it needs familiarity about the wide-ranging appliance domain as well as information about the recent problem and its circumstance.

The major CBR tasks are the retrieve, reuse, revise and retain which is further decomposed into subtask to complete the CBR cycle. The descriptions of the CBR task with their functions are given as below:

- **Retrieve:** This major task performs the function of retrieving the similar case from the stored case base with the help of following subtasks as given below:
  - **Identify features:** Before selecting the similar case this task identifies the detail of the upcoming problems. For this purpose, this task consists following subtasks as given below:
    - **Collect descriptors:** First the case reasoner collects the descriptors of stored case base of the CBR system.
    - **Interpret problem:** After collecting them, it understands upcoming problem.
    - **Infer descriptors:** It concludes the features and helps in retrieving the similar cases.
Search: After identification of features of upcoming problems, it searches the similar cases from the case base. For this purpose it performs following subtask as given below:

- **Follow direct indexes**: Every case in case base has a unique index. In the initial stage it tracks the indexes of the case directly.
- **Search index structure**: Then it searches the index structure of the case base. It matches the indexing of cases regarding upcoming problem constraints.
- **Search general knowledge**: This task collects the all important details regard indexes.

Initial Match: This task manipulates the similarity between the cases. The similarity measurement helps in finding more matched case regarding the new case. For this purpose it performs following subtask as given below:

- **Calculate similarity**: This task computes the similarity between the cases using various similarity measurement techniques. It can calculate the both global and local similarity between the cases.
- **Explain similarity**: This subtask explains the factor of finding similarity between the cases.

Select: After calculating the similarity between the matched case, the similar cases are being selected through following subtasks as given as below:

- **Use selection criteria**: It uses the appreciated selection criteria for choosing the more similar case.
- **Elaborate explanations**: This subtask describes the factors for selecting the particular case from the case base.

Reuse: This major task performs the function of adapting the information from the retrieved case to generate the proposed solution with the help of following subtasks as given below:

- **Copy**: This subtask copy the retrieved case following details through the help of sub tasks as given below:
- **Copy solution**: This subtask fetches the information stored in the retrieved cases.

- **Copy solution method**: Along with information stored in the retrieved case, it fetches the methods associated the information stored in the retrieved cases.

  - **Adapt**: This subtask becomes accustomed with retrieved case & perform following subtask as given below:
    - **Modify solution**: This subtask transforms the information stored in the retrieved cases.
    - **Modify solution method**: Along with information stored in the retrieved case, it transforms the methods associated the information stored in the retrieved cases.

- **Revise**: This major task performs the function of revising proposed solution to derive the final solution with the help of following subtasks as given below:
  - **Evaluate solution**: At this stage the proposed solution is being evaluated to generate the optimized solutions with following subtasks as given below:
    - **Evaluate by teacher**: At the initial stage of Revise phase, the proposed solution is being evaluated by the trainer with expertise skills.
    - **Evaluate in real world**: After evaluation by the trainer, the proposed solution is being evaluated in the current environment.
    - **Evaluate in model**: With evaluation with working conditions, the model is being evaluated to prevent the propagation of errors.
  - **Repair fault**: After the evaluation of the proposed solution, the next task is to repair the faults as given below:
    - **Self-repair**: First way to renovate faults is through by self by the system. It removes the faults and errors found in the proposed solution.
• **User-repair:** This subtask allows the user to remove the faults and errors faced in the proposed solution to guide the CBR system to produce the final solution free of faults and errors.

• **Retain:** This major task retains the final solution in the case base for future purpose with the help of following subtasks as given below:
  
  o **Integrate:** Before storing the case consisting final solution of upcoming problem it integrates the solution to the case through following subtasks as given as below:
    
    ▪ **Return problem:** Again the case reasoner comes back to the problem before storing the final solutions.
    
    ▪ **Update general knowledge:** The knowledge of the entire CBR system is being updated and it supports learning.
    
    ▪ **Adjust indexes:** Before storing the final solution, the case indexes are being updated to store the final case.

  o **Index:** This task manages the indexing of the cases stored in case base with executing the following tasks as given below:
    
    ▪ **Generalize indexes:** This subtask oversimplifies the index of the cases in the case base.
    
    ▪ **Determine indexes:** This subtask manipulates the index of the new cases in the case base.

  o **Extract:** Finally the final solution is being stored in following subtasks as given below:
    
    ▪ **Extract solution method:** It extracts the methods involved in the solutions.
    
    ▪ **Extract justifications:** It extracts the validation of the solutions for generating optimized solution.
    
    ▪ **Extract solutions:** It extracts the final solutions for future purpose use.
    
    ▪ **Extract relevant descriptor:** It extracts the significant descriptor of the solutions.
These CBR tasks are hierarchically related and perform a specific function initiating from selecting the similar case for generating the optimized solution with learning capability over the time in specified or unknown domains. The case-based reasoning task supports full utilization of the experience without allowing propagation of errors [95].

4.4 RELATIONSHIP TO OTHER APPROACHES

There exists a number of reasoning approaches in the Artificial Intelligence. These approaches enable the users to take the decision in different situations. Let’s find out the similarities and dissimilarities with other existing approaches.

4.4.1 Memory-based reasoning

Memory-based reasoning is often measured as a subtype of Case-based reasoning approach. MBR solves problems by retrieving stored precedents as an initial point for newfangled problem-solving. Its prime center is in the retrieval process using parallel retrieval schemes to facilitate retrieval without a conservative index assortment. These parallel models support quick access to stored precedents but facing the problem of selecting criterion for knowledge access.

4.4.2 Analogical reasoning

The CBR users view the case-based reasoning approach as primarily analogical. As discussed, the CBR approach solves new problems by applying analogous previous episodes. The models of analogy and CBR inspect the similar cognitive procedure. The CBR approach consists the processes that are related to occurring equally earlier than and subsequent to mapping. The relationship between the case-based reasoning and analogical reasoning is defined as below:

Case-based reasoning = retrieval + analogy + adaptation + learning.

4.4.3 Information retrieval system & database

The case-based reasoning systems become accustomed retrieved cases & the retrieval process in CBR differs from information retrieval systems and standard databases. The database and information retrieval system do not support the adaptation phase. The CBR adaptation phase is supplementary energetic. Database systems and IR systems encounter the problem of formulation of user queries. The CBR systems are being designed to
establish from an input description using features of the cases in memory and use the appropriate retrieval algorithm. These factors show the similarities and difference the CBR & IR system. The case-based reasoning approaches may be seen as the simple database system. The CBR system maintains the case base that’s actually database holding various cases. But the database does not support the adaptations and revision of the data stored. Hence we cannot consider the case-based reasoning as a single database for storing case base. The CBR system is capable of preventing the propagation of the errors from one stage to others.

4.4.4 Learning methods

The case-based reasoning supports the learning facility along with time pass. The CBR support both types of learning- inductive and explanation-based. Unlike other approaches it defines the concepts by generalizations and abandons the exemplars based on the generalizations entirely on saving cases. This approach provides the specific advantages in retaining specific cases. It makes decisions supplementary explicable & makes the decisions supplementary unpredictable due to the user examination of the cases directly to evaluate their applicability. The CBR approach also contrasts with knowledge-poor inductive learning methods due to emphasizing on the semantics of a domain, through similarity and retrieval criteria and case adaptation knowledge.

4.4.5 Rule-based reasoning

The rule-based reasoning may be defined as the combination of the predefined rules. The rule consists the conditions that define which actions to be performed. The condition is true then the specific action is performed otherwise other action is fired. But case-based reasoning is totally different from the rule-based reasoning. Instead of firing the rule, it solves the upcoming problem through finding the similar cases of the new problem and provides the solution through utilization of past experience.

4.4.6 Human reasoning

The case-based reasoning is found as the best option to implement the human reasoning. Other approaches are unable to capture the working of the human reasoning. As the human beings provide the solution to the upcoming problem by remembering the past
experience, the case-based reasoning does the equivalent in its working. As the human reasoning, the case-based reasoning stores the past experience in the memory. On arrival of new problem, it finds the similar cases and utilizes the information associated with previous solutions as the human beings think [96].

These are mostly used reasoning approaches in the Artificial intelligence applications for taking the decisions in different type of situations. This approach overcomes the limitations of the existing knowledge-based system. Mainly case-based reasoning system reduces the knowledge acquisition process which limits the performance of the whole reasoning system. This section highlights the similarities and dissimilarities between the CBR and other reasoning approaches.

4.5 ADVANTAGES OF CBR APPROACH

The case-based reasoning are proficient of providing the enhanced solutions to the problems faced in existing CBR through enhanced elicitation techniques with facilitating the collaboration among KBS and databases. The CBR approach is attracting awareness since it seems to unswervingly concentrate on the problems. The case-based reasoning support following features as given below:

- The CBR does not necessitate an explicit domain model and this elicitation becomes an assignment of congregation case histories.
- The CBR implementation is so easy that diminish of identifying noteworthy description having the case description.
- This approach manages the case base through employ of the database techniques in bulky case base.
- The CBR systems can learn by acquiring new knowledge as cases thus making maintenance easier.
- It enables the reasoner to recommend solutions to a problem speedily.
- It allows the reasoner to propose solutions in both known and unknown domains.
- It generates the solution in the absence of the generalized algorithms for solving the upcoming problems.
- It prevents the propagating the errors through remembering previous experience.
- The case reasoner utilizes the information in case associated with stored cases by
pointing out what qualities of a difficulty are significant. These facts indicate the advantages of case-based reasoning approach over other reasoning approaches. But these advantages are dependent on following constraints as given below:

- The experience of case reasoner
- The case reasoner’s capability to appreciate newfangled situation regarding old experiences
- Its proficiency at Reuse phase
- Its proficiency at evaluation and repair in Revise phase.
- Integration process for new experiences into its remembrance properly [97].

These constraints manipulate the performance of the case-based reasoning system in solving the new faced problems.

### 4.6 CASE RETRIEVAL PHASE

The case retrieval is the initial and key phase of the case based reasoning system. The case retrieval phase may be defined as the process of finding the similar cases with corresponding to the upcoming problem. There exists the specified strategy which judge the case whether the particular case is apposite for the selection or not is called as the retrieval strategy. The selection strategy ensures and validates exacting case of being retrieved on the basis of distance between the present case and the accessible cases stored in the case base of the CBR system.

The repossession process of case from case retrieval phase is being completed into the following stages as given below:

- **Recognize features**
  At this step, the problem is being recognized & all clarifications relating to the problem is being self-possessed. The attributes of the upcoming problems define the mechanism for solving the problems.

- **Investigate**
  Depending on the index based information of stored cases in case base, the required solution is being searched. This phase finds out the solution of the problem in the initial phase.
Initially match
Initially on foundation of comparison we compute the similarity between the newfangled case & stored cases in case base.

Select
On the basis of similarity between cases, the retrieval strategy takes the decision regarding more approximate case for producing the projected solution.

Following these steps, most appreciated cases are being selected for new problem. The case retrieval phase is dependent on various constraints for choosing the case retrieval method. The main factors for case retrieval are discussed as below:

- Quantity of cases to be searched for the latest problem that manipulates the number of required case in different type of applications.
- Accessibility of domain précised knowledge defines the mechanism of accessing the information stored in the case base.
- Simplicity in formative of weightings for entity features of the finicky cases for indicating which factor is more important.
- Indexing all cases which should be whole case or the identification features’ importance. The saved cases are indexed by the exacting labels, the new situation is documented as an explanation addicted to that index and traverse apposite indexing paths to locate the relevant cases.

Majority the fastidious indexing scheme assortment is the majority anxiety for the case retrieval in case-based reasoning. These indexing schemes investigate memory using levels, and decide the unsurpassed of the retrieved cases. The indexing process generates the gathering of pertinent cases with deference to the new case.

After indexing scheme, conception of the similarity measurement among the cases is a means of filtering comparable case from the productivity of indexing procedure. The similarity measurement is a moderately tricky chore to execute. The similarity may be defined as the quantity that reflects the potency of affiliation between two substances. The rate of this measurement is recurrently having variety of either -1 to +1 or normalized in 0 to 1. The cases can be formed for each other on the foundation of similarity measurement ideals & they may be grouped using k-means clustering. The basic benefit of grouping of cases is that the uniqueness of each group can be
documented. It explains the behavior of the groups or clusters. Grouping also may give more efficient organization and retrieval of information. It also helps in predicting the behavior of the new case & simplifying the data that we have into supplementary reasonable affiliation. These factors demonstrate the effects of similarity measurement for case retrieval process. There are a lot of techniques that employ the perception of similarity measurement in various case retrieval algorithms.

### 4.6.1 Various case retrieval algorithms

There exists a numeral of case retrieval algorithms relevant in case based reasoning system. These algorithms are based on the similarity metric that allows similarity between cases stored in the case base. The nearest neighbor retrieval algorithm & induction retrieval algorithms are two principal algorithms are used in this progression. Nearest-neighbor retrieval is an uncomplicated approach that calculates the similarity among the cases through indexing. The case is selected on the significance of weighted calculation of its attribute. When the value of weighted calculation of its features is superior to other cases, then scrupulous case is designated from the case base.

For example, there are numerous case (case1, case2, case3 and case4) selected though indexing from the case base then case4 will be measured as the nearest neighbor among these cases from the case base due to

\[
\text{similarity} (\text{NewCase}, \text{case4}) > \text{similarity} (\text{NewCase}, \text{case1}), \\
\text{similarity} (\text{NewCase}, \text{case4}) > \text{similarity} (\text{NewCase}, \text{case2}) \text{ and} \\
\text{similarity} (\text{NewCase}, \text{case4}) > \text{similarity} (\text{NewCase}, \text{case3}).
\]

The Nearest-Neighbor algorithm is essentially sloping on the similarity value. For every case, initialize value of total similarity to 0. For each case retrieved from database calculate the value of \( \text{sim}(f_{\text{NewCase}}, f_{\text{casek}}) \) first by following formula:

\[
\text{sim}(f_{\text{NewCase}}, f_{\text{casek}}) = \frac{\Sigma_{k=1}^{n} (f_{\text{NewCase}} \ast f_{\text{casek}})}{\Sigma_{k=1}^{n} \sqrt{(f_{\text{NewCase}})^2 + (f_{\text{casek}})^2}}
\]

Using \( \text{sim}(f_{\text{Newcase}}, f_{\text{Casek}}) \) then calculate the similarity value over all significant weight as given below:

\[
\text{Similarity} (f_N, f_k) = \frac{\Sigma_{k=1}^{n} W_i \ast \text{sim} (f_N, f_k)}{\Sigma_{k=1}^{n} W_i}
\]
Where \( f_N \) & \( f_K \) are value of features related for new case & particular case that is stored in the case base. The \( w_i \) is the significant weight of a feature & sim is the similarity function of features. Next evaluate the entirety similarity values of all the cases and discover the adjacent case for newfangled case.

The major benefit of the nearest neighbor retrieval algorithm is that it is greatly straightforward in the implementation phase. An additional benefit of the nearest neighbor retrieval algorithm is that preindexing is not required. The preindexing process consumes more time in designing the case base. But the major inadequacy of this algorithm is that it is slow in handling large case base. In the nearest neighbor algorithm, the CBR system faces the slow speed for retrieving the case. As the size of the case base grows, then performance of this approach reduces with the size of the case base.

The induction retrieval approach fails in case of incomplete information required in the case. This approach determines precise features in the entire the superlative work in perceptive cases and generates a decision tree type structure to organize the cases in memory. This approach seems extremely effectual in a situation having the requirement of single case feature as a solution, and case feature is dependent upon others. This algorithm produces the high-speed repossession speed. It fails due to facing the incomplete & data missing cases & it is also reliant on time consuming preindexing process. In mostly real time applications, the information fields are volatile so these fields on totally self-regulating on each other [98]. These algorithms are not sloping on domain precise knowledge. Due to all these facts, these algorithms are not so a great deal efficient in the case retrieval in case based reasoning system. The knowledge-intensive similarity measurement techniques may be applied to overcome the problems faced in both nearest-neighbor algorithm and Induction retrieval approach.

### 4.7 CASE REUSE PHASE

One of the major advantages of using the case-based reasoning is the utilization of the information and their associated methods to generate the proposed solution. Some time it may happen that there is a more significant gap between the upcoming problem and the retrieved cases. This cause the complex way to reuse the knowledge associated with the retrieved cases. So those cases should be adapted for minimizing the differences between
the new case and selected cases from the first phase of the CBR system. This phenomenon is known as the case adaptation process.

The retrieval phase generates the initial form of the final solution. This process only selects the similar cases. The case adaptation process transforms these retrieved cases into an appropriate solution by altering the retrieved case information. Hence the case adaptation is one of the major phases of the case-based reasoning as it generates the solution by utilizing the patterns associated the retrieved cases.

The adaptation phase is essential for the superior and rational case-based reasoning process. It is conscientious for discovering a solution to resolve a newfangled problem using the theory of \( k \)-nearest neighbors (\( k \)-NN). Statistical adaptation method is a standard technique for feature-based case adaptation because of its domain-independent and effortlessly to be implemented having small adaptation correctness capabilities.

![Figure 4.3 Case Adaptation phase](image)

There are various approaches available to perform the case adaptation process:

- The case retrieved may be used as a solution to the present problem without updating or with updating in the situations having the solution is not completely suitable for the existing circumstances.
• The steps that were followed to attain the previous solution may be repeated without updating or with updating in the situations where the steps taken in the earlier solutions are not entirely acceptable in the present circumstances.

• Where multiple cases have been retrieved, the solution may be derived from these multiple cases, numerous substitution solutions may be offered.

The adaptation can use a variety of techniques in the case-based reasoning system. The following constraints make a direct impact on the selection the strategy for case adaptation as given below:

• The similarity between the current and selected case during the retrieval phase.
• The characteristics of new case differ from selected cases.
• The common or known rules used in carrying out the adaptation.

These factors help in the process of selecting the case adaptation strategy for better utilization of stored information associated with retrieved cases [99]. After this process the internal assessment of the solution is being performed to the proposed solution as shown in figure 4.4.

![Figure 4.4 Case adaptation logic](image)

### 4.8 CASE REVISION PHASE

The next phase of the case-based reasoning is to evaluate the proposed solution by the case reuse phase. The main aim of this phase is to evaluate the proposed solution. The
revision phase performs the significant analysis the collapse of projected solution. After finding the crash in the projected solution, it maintains all these failures. The case revision phases perform following subtasks as given as below:

- **Evaluate solutions**
  The evaluation process takes the outcome from applying the solution in the authenticated working environment. This task is being performed outside the CBR system. The results from applying the solution possibly will consume some quantity of time to execute depending on the type of application. This step provides the required feedback for solution repair.

- **Repair fault**
  After finding the faults in proposed solutions, the next step of this phase is to fix the errors of the recent solution and retrieving the explanations for them. It repair the faults found in the proposed solution using an explanation-based learning technique. This form of learning moves finding of errors in a post hoc fashion to the elaboration plan phase where errors can be predicted and handled and avoided. This task uses the failure explanations to adjust the solution to avoid the failures in the future [100].

This phase does not allow to propagate the errors occurred in the duration of solving precedent problems.

### 4.9 CASE RETAIN PHASE

The last phase of the case-based reasoning is to store the final solution generating through the case reuse phase for the future use. This process of storing the case in the case base is known as the case retain phase. The case retain phase enable the CBR system for learning capabilities. Mostly the CBR system retains the case is to only evidence the information with reference to the target problem requirement and its finishing solution. In case of unreliable solution, then supplementary information would be required to store into the case library such as the required changes made to the retrieved solution. Such information should be saved with more care. In the case retain phase, the consequence of the problem-solving process is added to the system’s universal information. There can be multiple approaches used for retention the case consists in storing in the case base...
successfully solved problems (in the form of cases). Other approaches retrieve solved cases with additional information indicating to which point the proposed solution was satisfactory for the problem or whether it answered certain objectives.

- **Integrate the solution**
  Before storing the case consisting final solution of upcoming problem it integrates the solution to the case by returning problem back & updating general knowledge. Then for the final solution, the case indexes are being updated to store the final case.

- **Index the final solution as the case**
  This task manages the indexing of the cases stored in case base with executing the generalization of indexes and determining their indexes.

- **Extract the solution**
  Finally the final solution is being stored in case by extracting solution method and extracting justifications & solutions with the relevant descriptor of the solutions

This phase is directed related to the retrieval phase. It defines the performance of the CBR system directly. This phase extends the size of the volume of the case base that makes an impact on the search process of the required cases regarding the upcoming problem. So the process of retaining the final solution is so a critical process. To control the case base growth, the efficient case maintenance mechanism is needed to manage the CBR system. It may be done with help of deleting the old cases. For this purpose, the Random Deletion is the oldest and simplest deletion mechanism which can basically diminish cases. This method faces the complexity in preserving the proficiency while the high utility value of cases is deleted. The competence-based maintenance method is being applied for designing the deletion policy strategy for CBR. There are three main steps in this method. First this method formulates the problems, then it resolves the coverage and reachability set based on coverage value and finally reduce case base size [101]. This method is more useful in the case maintenance purpose.

### 4.10 CBR TOOLS

In the last sections we study the basic of case-based reasoning and their advantages. The concentration in CBR is owing to the sensitive nature of CBR and it may intimately be
similar to human reasoning. Currently the CBR tools have made the developers to design the application following the 4 REs phases. The tools have made a contribution in designing the case-based reasoning application in the real world. There are following tools are available for developing the CBR applications as given below:

- ART* Enterprise
- Case-1
- CaseAdvisor
- CasePower
- CBR-Express
- CasePoint
- Eclipse - The Easy Reasoner
- ReMind
- CASUEL
- ReCall
- jColibri
- myCBR
- freeCBR

These tools are being applied in designing the CBR applications with more efficient and reliable. The CBR tools follow the schematic of the architecture as shown in Figure 4.5.

Figure 4.5 Architecture of the CBR engine
4.11 DISTRIBUTED CASE-BASED REASONING

The conventional CBR systems seem as the single agent working with single case base as the central knowledge resource. The classic examples of this type of CBR system are Entree and FAQ Finder. The main drawbacks of the conventional CBR system is limitation of available resource involved in the case-based reasoning process. The distributed CBR enables the users to utilize the resources that are located distributed with the objective of improving the performance of CBR systems. The distributed CBR can be utilized in the different areas by following two key criteria:

- Knowledge organization within the system.
- Knowledge processing by the system.

Figure 4.6 displays the 4 categories of the case-based reasoning with intelligent agent as given below:

![Image of the 4 categories of the case-based reasoning with intelligent agent.]

Figure 4.6 Distributed CBR

4.11.1 Single agent single case base

This category consists the single intelligent working with single case base. This configuration is very easy to implement and managed. All information and experience related to the last problem solving evidence. It is only suitable for simple application limited to single intelligent agent and limited case base.
4.11.2 Single agent multiple case base

This category consists multiple case bases that can be accessed to the particular problem solving intelligent agent. During its operation the intelligent agent accesses these multiple and hierarchical case bases containing cases at different levels of abstraction. In this approach, the intelligent agents take the complete benefit of these numerous case bases. It is also known as multi-case base reasoning. The main concern of this approach is to reason about how to apply cases between various case bases. Another concern is to reason about when and how to draw on particular case bases. Finally this approach plans in which case base the case should be stored due to the availability of multiple case bases. These concerns are the main problems faced in this category.

4.11.3 Multi-agent single case base

In multi-agent systems, the multi-agent system shares out single case base. The multi-agent CBR system shares the single case base where a multiple number of problem solving agents drawing on the information held by a single case base. This approach uses Case Based Markup Language as the procedure for enabling distributed CBR over every network. This XML format for case representation enables granular updating. The main problem faced in this approach is that the servers do not have to dispatch the complete part of the case base to the client every time for change in the retrieved case. The single agent single case base systems that rely on using a single case base source for problem solving are limited by the scope of this knowledge resource. But this problem is totally resolved in this approach.

4.11.4 Multi-agent multiple case base

This category consists the groups of the intelligent agents that are capable of accessing the multiple case bases. Such groups of the intelligent agents are known as the multiple agent system. It means the there are more than intelligent agents working for common problem and the individual intelligent agent of this group is capable of accessing the cases from the multiple case bases.

These four categories are mainly used configuration of the intelligent agents with the case-based reasoning system. All categories has own cons and pros in using the case base. In these days the single agent single case base category is not suitable for handling
the challenges faced in the industry computing applications. The supply chain management system contains multiple entities such as the manager, manufacturer, suppliers, retailer, customers and distributors. The role of these entities is being implemented as group of intelligent agents- the ManagerAgent, ManufacturerAgent, SupplierAgent, RetailerAgent, CustomerAgent and DistributorAgent. This different type of intelligent agents work may work on different locations. The single case structure is not only suitable for solving the all information associated with different type of the problem. Hence we cannot save the whole past problem experience in the single case structure. Hence for this situation, the multi-agent multiple case bases approach is being found more comfortable for implementing the supply chain management system.

Figure 4.7 Distributed CBR Architecture

Finally, proactive learning can be used for acquiring proficiency sculpt that allows an agent to gain knowledge of how to recognize which other agents are best appropriate as cooperation partners. Competence models can improve both the size and the robustness of the ensembles of CBR agents, since using the most competent agents in an ensemble reduces its size while maintaining the ensemble effect [102].

4.12 DISTRIBUTED CASE BASE

The distributed CBR strategies are aimed to enhance the performance and maintainability of CBR systems. The vast majority of CBR systems have taken a single agent, single case
base approach to problem solving. These strategies define the retrieval and retaining the cases stored in distributed case base. In a multi-agent system, the whole problem solving process may not be run at a single physical location and may be distributed among the set of agents. The system performs the successful problem-solving episodes in which each agent stores its own local case in its case base. Finally, there are also latent gains in provisions of system maintenance as it may be easier to become accustomed with local case bases independently of each other.