DEZIDE

DEZIDE WHITE PAPER

Capture, organize and optimize expert knowledge using Bayesian Belief Networks

WHITE PAPER

CONTENTS

(click to jump)

Reach us through these channels:

Visit our website at http://www.dezide.com

Email us at info@dezide.com

Follow us on LinkedIn https://www.linkedin.com/company/100934/

But the really cool thing about Dezide is the patented algorithm that considers causes, probabilities, time and money when making its suggestions for the best corrective action.

Kenneth Grindsted
Lead Engineer, Vestas R&D

Executive Summary

At Dezide we help improve installation, service and repair processes by providing efficient troubleshooting knowledge to service centers, field service technicians and even end customers.

We have 20 years of experience with helping businesses of all sizes capturing, organizing and optimizing expert knowledge. We work with clients in industries ranging from wind, mining, and electronics to consumer printing and telecommunications.

We help solve the digitalization challenge within service and support, and our customers trust us through many years of partnership where the solution continues to help reduce Mean Time To Repair, reduce re-visits to fix the same issue, use fewer spare parts and retain knowledge when people leave the company.

We support current and future service and support strategies and help our customers capitalize on their knowledge as a service for their own and competing products in businesses with an increased focus on Industry 4.0 where service is key for pushing revenue and shareholder value even higher. We see a growing trend in the industry to provide service for customers even on competing products and Dezide provides that opportunity.

The software solution is built to make it easy to add troubleshooting capability as a component in a web service connected environment, and we have integrated Dezide with self-service, ticketing, CRM, CMS, ERP –both commercial and custom built.

Our customers have achieved amazing results including 60% reduction in calls to HQ, 50% faster troubleshooting time, 35% reduction of Average Handling Time and 30% deflection of calls due to self-service.

Dezide is a triple-A solidity rated company, with clients ranging from the Liebherr Group to the world's largest wind turbine manufacturer, Vestas.

Capturing Troubleshooting Knowledge - a Capability

In the context of business enterprises, we see that knowledge tends to be interpreted as possession of experience (tacit knowledge) as well as possession of factual information (explicit knowledge) – or where to get it. In an age of Big Data, IoT and a heavy focus on digitalization we believe that it's crucial to understand the difference between data, information, and knowledge – especially when capturing knowledge from the experts in a computer system.

Dezide started based on an idea of being able to capture expert knowledge and make it available to everybody in an optimized way using Bayesian Networks, and today we see Dezide help transform service from a cost consuming necessity to a revenue-generating business unit through new business opportunities.

But why do we need formalized troubleshooting?

Well, we can identify at least three major reasons:

1. Problem-solving tasks take up huge amounts of time. Many companies have hundreds or even thousands of technicians on the roster just for solving machine issues. There is an enormous potential for saving time and money.

2. It enables a structured approach to problem-solving, in which we build and maintain knowledge bases on a computer. We capture our current knowledge in a computer model, which we refine over time. In turn, these knowledge bases put expert knowledge in the hands of non-experts/ less skilled personnel, greatly enhancing their potential for problem-solving.

3. Problem-solving tasks are difficult, and computers are far better than humans for optimizing these tasks. Computers can do billions of calculations per second, and with an appropriate math ematical model, the computer can potentially perform far better reasoning than any human.

As we gain experience, we can solve the problems faster, but because we tend to use ad hoc approaches like "trial and error," learning is a slow process. We tend to forget solutions over time, and as a result, we end up starting from scratch each time. What is needed is a structured approach where previous efforts are saved, reused and improved upon.

In general, we can say that the simpler the problem is, the easier it is to make a troubleshooting system that guides inexperienced people to a solution, and the more complicated the problem is, the larger is the potential benefits.

However, regardless of how complicated the domain is, there is a great potential for optimizing problem-solving tasks by structuring, reusing and improving existing knowledge.
That is why we need capture, organize and optimize expert troubleshooting knowledge.

What is Troubleshooting?

Decision-theoretic troubleshooting (or simply **troubleshooting**) can be seen as an exact science for "detection." However, at the core of decision theoretic troubleshooting is not only the goal of arriving at the true diagnosis or conclusion of a given problem but also a desire to do so as efficiently as possible.

The faster (or cheaper) we arrive at the proper conclusion, the better. But decision theoretic troubleshooting goes further than that because it establishes a principled way to solve the problem while concurrently narrowing down the potential causes of the problem. As such, it is detection and problem solving mixed together in the most optimal manner. In modern terms, this is called **decision theory**.

When we build a computer system based on decision theory, it often takes the form of an expert system which gives advice to humans in certain contexts (e.g. to advice a doctor that it would be most beneficial to make a CT-scan of the patient) or which semi-automates some task that is normally carried out manually by humans (e.g. to provide a customer with technical support about his printer over the phone). We typically create the system as a synthesis between human expert knowledge and the enormous computational power of a modern computer. This fusion is enabled by an underlying mathematical model, and if this model is sound, the resulting system can potentially reason far better than any human.

Basically we describe a problem and its solution via three components:

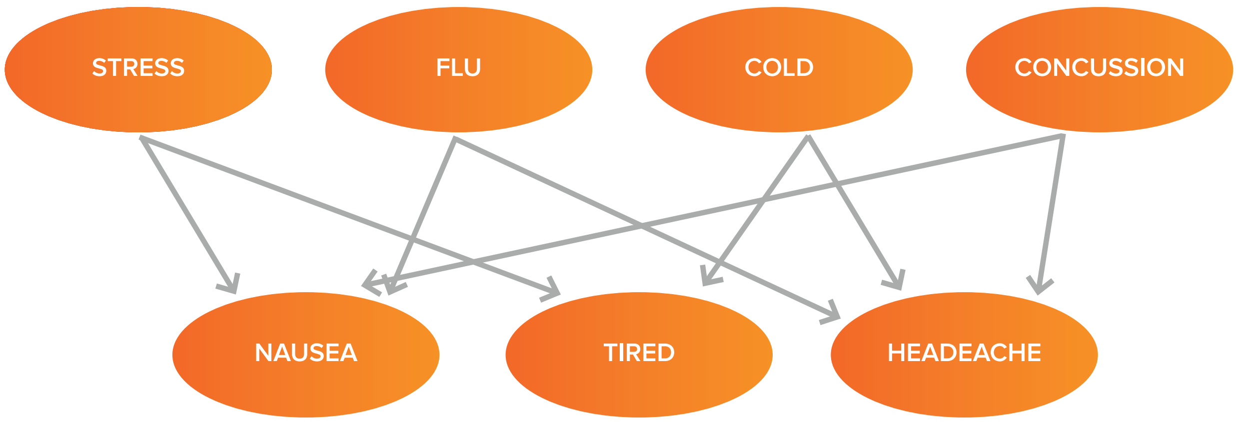
1. **Causes** of the problem,
2. **Observations** which may narrow down the potential causes
3. **Repair actions** which we may perform to fix the problem.

For each observation or action, we also associate a cost which describes the resources required to make the observation or perform the action. As a final component, the model describes the probabilistic relationship of the domain via some form of a probabilistic model over causes, observations, and actions.

In Dezide, that model is **Bayesian Belief Networks**.

Models of a Problem Domain Using Conditional Probabilities

Bayesian Belief Networks are graphical models of a problem domain using conditional probabilities to represent relationships between events – e.g., the probability of developing lung cancer given the fact that the patient is a smoker. Bayesian networks are a newer AI technology that enables one to reason in complex situations with many confusing, uncertain and conflicting sources of information. Bayesian networks are based on Bayes' formula for reversing the causal direction of conditional probabilities, allowing one to reason about causes based on information about the effects or symptoms.



A Bayesian network consists of a number of events that may be observed, and at any time a calculation can be made to find the probabilities of the non-observed events based on the available information. As opposed to ad-hoc technologies such as fuzzy, rule-based and case-based reasoning systems, Bayesian networks are mathematically proven to be correct – even in very complex situations. A Bayesian network is the best available technology for handling very complex decision scenarios.

As opposed to other AI technologies such as neural networks/machine learning, fuzzy systems, and case-based reasoning, Bayesian networks provide a theoretically sound approach that scales well with the amount of information, unaffected by some of the information being uncertain or conflicting. Most other methods are based on ad hoc computational algorithms that break down when the complexity or uncertainty becomes high.

Bayesian Belief Networks for Handling Complexity

BBNs are a fundamental reasoning technology particularly suited for handling complex and uncertain domains. A BBN is a model of the domain where events and their relationships are represented in a graph with associated probabilities.

BBNs have become increasingly popular in the industry over the last couple of decades. Over the years they have been used for information retrieval, troubleshooting and help systems, credit fraud detection, diagnosis of medical problems and many other applications.

The key strength of BBNs is their ability to perform accurate reasoning in the presence of significant uncertainty. It has been said that BBNs mimic human reasoning, but the truth is probably that they often improve upon human reasoning. Further, a very compact representation of knowledge is possible with BBNs, and due to the rational handling of uncertainty, not only are the explicitly represented cases covered, but also corner cases and conflicting cases are handled without additional modeling effort. Further, BBNs are exceptionally good at simultaneously factoring previous experience with present observations, and facilitate the incorporation of contextual information into the reasoning process. These strengths allow BBNs to tackle complex reasoning domains in a variety of circumstances and permit the user to extract explanations for recommended decisions under conditions of uncertainty – a key advantage over other reasoning technologies.

Two major challenges have prevented BBNs from obtaining wide industry acceptance:

- They have traditionally been very difficult to construct, requiring a deep understanding of BBN theory.

Further, all administration tools have required a low-level expression of BBNs, making it difficult to quickly build useable models of complex real-world systems.

- Second, BBNs have traditionally been very compute-intensive with unpredictable tractability, limiting applications to less complex and less interesting domains.

These challenges have been addressed and overcome with Dezide. The Dezide Administration tool requires no knowledge of BBN theory, thus enabling domain experts to develop models by themselves. Secondly, the BBN models produced by Dezide are guaranteed to be tractable. When these models are deployed, they are executed by the Dezide decision engine generating an optimal step-by-step trouble-shooting sequence.

The Dezide decision engine executes BBN models in a highly efficient manner, finding an optimal sequence of steps balancing belief of the step being helpful with its cost. Taking many factors into account, the Dezide engine produces a sequence of solution-oriented and/or information-gathering steps.

The engine executes BBN representations of specific decision support problems and the BBN representations are highly parameterizable for different audiences and applications, and easy to integrate into other processes and systems. Therefore, the Dezide engine can be used in interactive, semi-automated and completely automated settings. Further, parameterization can facilitate highly personalized decision support experiences, tailoring the service to the users' preferences for risk, level of sophistication, prior knowledge of the domain, and contextual need for the decision.

Dynamic Guides - Causes, probability, time and cost

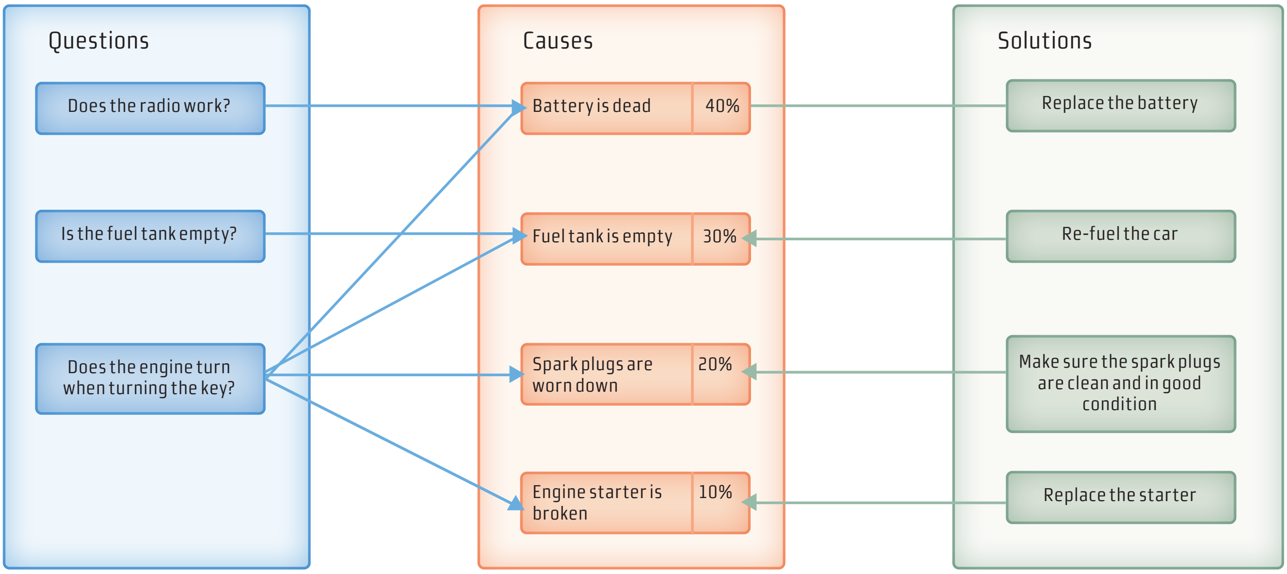
Dezide has its own concept based on BBN's for capturing expert knowledge. The concept builds on the following three central elements that are used for building intelligent troubleshooting guides:

- **Causes**
  - The root causes is a specific item that is known to be an actual cause of the problem. A cause can be anything from "the device isn't turned on" to " component X failed on board Y." The depth of causes will depend on who the intended audience is and what level of detail is needed

- **Solutions**
  - A solution is a specific task known to solve a cause. A solution is any corrective step taken within the troubleshooting process that requires performing an actual task

- **Questions**
  - Questions are observations used to identify, remove or clarify causes and play a very integral part in any complex troubleshooting situation. In complex guides that contain a wide range of causes, questions can help zero in on the solution that needs to be performed to solve the problem

Causes, solutions, and questions are linked together to form a troubleshooting guide. It is easy to build even large troubleshooting guides and the guide building process can be learned in a few days.



An advanced troubleshooting guide containing 50 questions can be constructed in one or two days. To create a traditional static decision tree for this would require manually structuring a very very large tree with 50 branchings and determining the optimal order of the questions to be asked manually! Dezide makes this calculation automatically based on the probabilities and cost factors (time and money) provided by the knowledge engineer.

Dezide ensures that the optimal troubleshooting sequence is used by always presenting the step that is most efficient given the current situation. The most efficient step is chosen such that it minimizes the Mean Time to Repair because this provides the largest cumulative savings.

Dynamic guided troubleshooting is the core competency of Dezide and we believe that we provide the very best commercially available technology for troubleshooting based on Bayesian Networks with IoT integration for optimal troubleshooting efficiency.

Dezide consider causes, probability, time and cost when making suggestions for the best corrective action to perform at any given time in a troubleshooting scenario and we allow the user to skip steps that he is unable to perform.

We use “Symptom Questions” for identifying an effect created or induced by the causes and for revealing if something is broken, malfunctioning or showing signs of abnormality.

We use “Configuration Questions” to represent the context of the guide which affects the causes of a guide by inherent, external properties as they govern how likely causes are in a certain situation or environment.

The Guide Administration tool includes the Guide Validation System which makes sure that all guides are built properly, do not contain duplicated solutions or other errors that will result in bad user experiences.

The Guide Validator is the foundation for building high-quality guides that are in a coherent state with no missing links between causes and actions, no incorrect probability settings, no conflicting constraints and much more.

Anyone Can Build Bayesian Networks with Dezide

Dezide has researched and patented several breakthrough innovations in the area of Bayesian networks allowing the commercialization of Bayesian networks in our software.

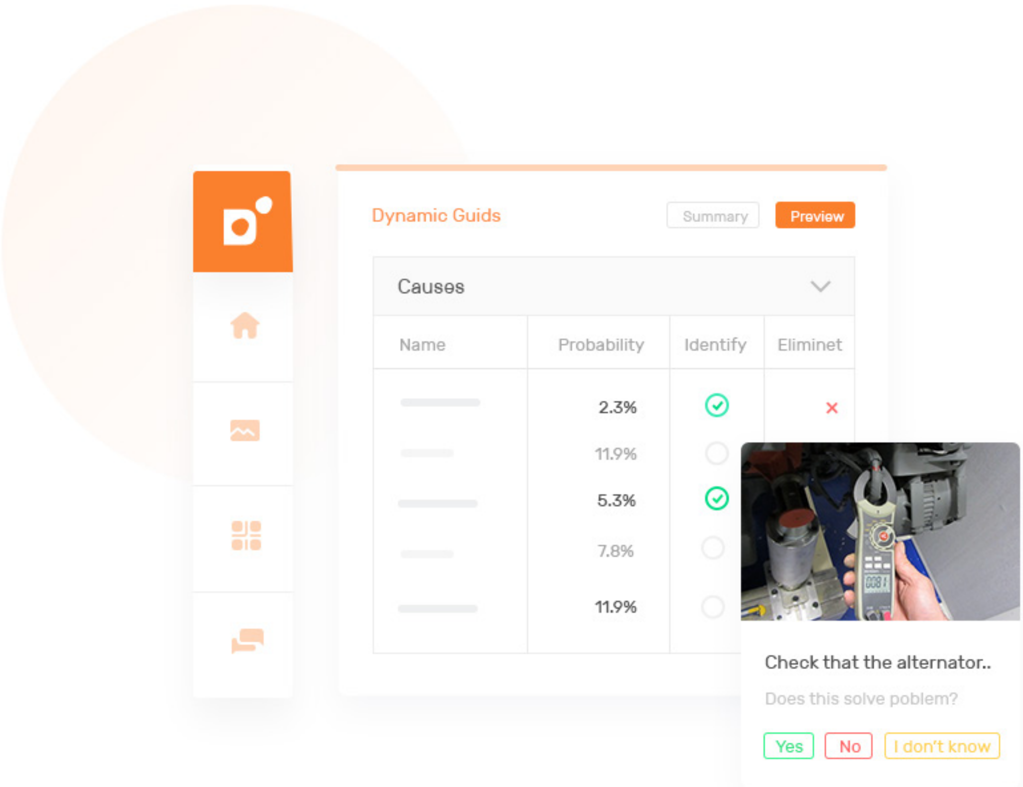
These inventions have broken previous barriers with Bayesian networks such that:

- Anyone can learn to build the knowledge base in Dezide with no prior knowledge of Bayesian networks
- A high degree of scalability is obtained to allow Dezide to handle very complex problem scenarios efficiently

The underlying Bayesian network is used to provide an inexperienced user with expert guidance on how to resolve a complex problem or reach a diagnosis. The Bayesian Network will continually be recalculated to find the best next question to ask or the best next solution to try out. When the user responds, the answer will be inserted as evidence into the Bayesian network – which will become more knowledgeable about the situation. The technology behind Dezide guarantees that the optimal sequence with the fewest number of troubleshooting steps will be found.

Dezide can be perceived as a dynamic, self-adapting decision tree that keeps learning over time. Traditional decision trees are simple static structures that don't adapt to the situation and learn over time.

Traditional decision tree approaches require the user to answer all questions, but Dezide allows you to skip a question or solution at any time if you are unable to answer. In this case, Dezide will try another approach to finding the optimal path to a solution.



Dynamic Guide Complexity

People always ask us how big guides we can build in Dezide before we start seeing performance issues and maintenance of the guides becomes difficult, so for the interested, here are a few points on guide complexity and size:

- Complexity is linear to the number of causes and actions
  - O(N_C + N_A)

- Complexity is polynomial to the number of questions
  - O(N_Q^2)
    - The Bayesian algorithm calculates the optimal sequence of steps for each answer to a question

- If there are only causes and actions in a guide then there is virtually no model size limit

- If we build a guide with approximately 150 questions we will start seeing performance issues

- But, if we use data collection / IoT integration for automatically answering some of the questions using e.g. machine state data, then there is no limit

Generally, breaking large guides into sub-guides significantly increases performance. The complexity of a large guide with complexity O1 broken up into a super guide becomes:

- O(sqrt(O1))
  - This means that if the guide contains primarily causes and actions, the complexity becomes less than linear which is very good.

If the guide contains a lot of questions (that are not answered through data collection/IoT), the complexity becomes linear, which is still pretty good. In general, splitting a large guide up into a super guide structure positively affects the complexity quite a lot.

So, consider the guides as Lego-blocks and build a hierarchy of guides representing re-usable " components/modules" and there will never be an issue with performance and the knowledge base will remain easy to manage in the long term.

Guide Optimization

Dezide includes a sophisticated self-learning module used for improving the performance and accuracy of the intelligent guides based on actual usage of the deployed guides.

Data collected from the online and offline troubleshooter is analyzed by the Guide Optimizer which will provide a newly updated set of guides that more correctly reflect the “real world”.

The guides are originally built based on the experience of subject matter experts. When this experience is combined and refined with the learning from usage data from end users, the guides are improved and become better at solving issues faster and more reliably.

This is an example optimization report from Dezide, where changes in cause probabilities are visible.

Causes						
Name	**Original Probability**	**New Probability**	**Change**	**Original Marginal Probability**	**New Marginal Probability**	**Change**
Relay broken	0.1 %	12.2 %	12.2 %	0.1 %	12.2 %	12.2 %
Oil temperature below 30 degrees celcius	90.5 %	38.2 %	-52.3 %	90.5 %	38.2 %	-52.3 %
Butterfly valves are closed	0.1 %	5.0 %	4.9 %	0.1 %	5.0 %	4.9 %
oil sensor valves are not calibrated correctly	0.1 %	0.0 %	-0.1 %	0.1 %	0.0 %	-0.1 %
Not enough oil according to correct oil level	0.0 %	2.9 %	2.9 %	0.0 %	2.9 %	2.9 %
No signal or wrong signal from oil sensors	0.1 %	9.1 %	9.0 %	0.1 %	9.1 %	9.0 %
Upgrade needed according to CIM 2783	0.1 %	7.2 %	7.1 %	0.1 %	7.2 %	7.1 %
Sensor installed incorrectly	0.0 %	9.6 %	9.6 %	0.0 %	9.6 %	9.6 %
Gear oil is of wrong type	7.1 %	11.7 %	4.6 %	7.1 %	11.7 %	4.6 %

Competing approaches

The below table compares various types of technologies used by vendors of AI technologies for technical troubleshooting concerning complexity and performance when used in different situations.

Property	Desicion tree	Rule-based system	case-based system	Standard Bayesian net	Dezide advisor
Storage complexity	High	Low	High	High	Low
Execution complexity	High	Low	High	High	Low
Complexity and difficulty of authoring	High	High	Very high	Very high	Medium
Handling of uncertainty in domain	None	Some	Some	Excellent	Excellent
Handling of uncertain and missing user reply	None	Some	Some	Excellent	Excellent
Maintenance of guides	Difficult	Difficult	Difficult	Difficult	Easy
Optimization of parameters	No	Yes	Yes	Yes	Yes
Modularity and re-use	Difficult	Difficult	Difficult	Difficult	Easy
Handling of unexpected situations	Weak	Weak	Weak	Excellent	Excellent
Parameterizability	None	Some	High	Some	High
Handling of conflicting data	None	Weak	Weak	Excellent	Excellent
Technology fit	Small domains	Small domains	Medium domains	Small domains	All domains

Dezide compared to Case Based Reasoning

We are often asked about the difference between AI driven expert systems based on Case Based Reasoning (CBR) and Bayesian Networks, so here is our take on explaining that difference as we see it.

CBR works by collecting a large number of "cases" from the problem domain. A "case" consists of a specific problem, a number of symptoms, the root cause, a number of repair steps that were not successful, and the repair step that actually solved the problem - all of this information refer to the same specific problem case. The idea is that if you collect a large number of such cases, you can start to do pattern matching and actually determine the best matching of the old cases when you are looking at a new case. If you are looking at a new case where you have symptom X, Y, and Z, then you may have 10 cases in the database matching this. You will then look at these 10 cases to see which repair steps were most successful and suggest these first.

CBR has a number of problems when compared to Bayesian Networks:

- **More time-consuming to build:** We believe the process of developing the knowledge base is more time-consuming since administrators have to develop a decision tree for the problem domain first and then generate cases based on this tree.

- **More time-consuming to maintain:** There is no concept of modules or "objects" in a CBR database. So it is hard to build once and reuse many times. Also, if a changed component leads a change in how a symptom indicates causes and repair steps, then the administrator will have to go through a potentially enormous amount of cases where the symptom is present. Possibly all these cases will have to be discarded as there may not be an easy way to re-validate the cases. In the Dezide Advisor system, you can very quickly modify how a symptom is linked to causes and repair steps.

- **Size of knowledge base becomes enormous: :** the size of the case base grows exponentially. If you have a relatively complex system with 100 symptoms, 100 causes, and 100 repair steps, you will need to cover relatively well the state space which is 2^100 x 2^100 x 2^100 ~ 4 x 10^30 (if symptoms are binary). Obviously, this will be almost impossible to do. To cover this situation well in Dezide Advisor, you need to represent the 300 elements and only those relationships that make sense. If all relationships are relevant this could be around 20,000 relationships - still not impossible but of course a large task. Normally only a small percentage of the possible relationships are actually necessary to define, perhaps 5-10%.

- **Symptom management:** CBR has a problem with situations where differences between symptoms are not equally important. We can have a situation where two cases are the same for all symptoms, except one. All the similar symptoms are not very important (it doesn't matter if they are different) - but the symptom that is different is extremely important. These two cases would normally be matched as near-identical by a standard CBR system if not handled specifically. Of course, it can be done, but it takes much extra time.

- **Uncertainty:** if you know the uncertainty of causes or of relationships between causes and solutions, you can specify it precisely in Dezide Advisor. This cannot be done in CBR systems, leading to more incorrect and uncontrollable behavior.

1. **Efficiency of service:** because a troubleshooting guide is founded on a consistent probabilistic model together with costs, the guide can minimize the cost of solving the problem. CBR systems have no such formal underpinning and therefore they have to resolve to some form of semi-intelligent, ad-hoc guessing. The end result can easily be 50-100% faster troubleshooting with intelligent guides.

2. **Precision of modeling:** assume one symptom can indicate three different causes to a varying degree. In a CBR system, there is no way to say which cause is more relevant than others to this symptom. In Dezide's troubleshooting guides you can set individual symptom probabilities for each cause as needed. In the end, this extra precision is important for gaining the high troubleshooting efficiency.

3. **Adapting to unknown situations:** Say a guide has 10 symptom questions. These 10 question can be answered in more than 1000 different ways. Dezide's troubleshooting guides automatically (dynamically) adapt to each situation in an optimal manner. CBR systems have difficulties taking advantage of a particular situation defined by the symptoms. For example, the CBR system may have many cases that are ranked equally based on the symptoms. This is another reason for the high troubleshooting efficiency of intelligent guides.

4. **Optimization:** each troubleshooting guide can be optimized with statistical machine learning. This adjusts probabilities of causes as well as symptom probabilities. Since CBR systems have no formal underpinning, they can only take advantage of statistical information in limited and mathematically inconsistent manners. The end result is that the Dezide troubleshooting guide will get better at solving the problem after it has been optimized, whereas a CBR system has no way of knowing if it improves the troubleshooting process.

5. **Short phone codes:** a phone code is a compressed representation of a troubleshooting sequence which can be used to recover the entire troubleshooting sequence outside a closed facility. With intelligent troubleshooting guides, we can compress this sequence enormously, leading to very short phone codes even for 20+ step troubleshooting sequences. Furthermore, a code never expires and will keep working even though the guide has been updated with new actions and causes. This is simply impossible with CBR systems.

So in our view, CBR lends itself more to relatively simple domains where you have limited knowledge of relationships between causes, symptoms and repair steps, and where you are not very concerned about the precision of the system. It does potentially give you a more automated approach to build the knowledge base, but the end-results will also be of much lower quality, efficiency, and precision.

All in all, we believe that there are compelling reasons for not choosing CBR systems. They were among the first AI systems, but they have unfortunately not evolved a lot during 20+ years of modern probabilistic AI research. In today's world, they lack the power, the flexibility and the skill of a modern technology like Dezide intelligent troubleshooting guides.

IoT Integration / Equipment Data

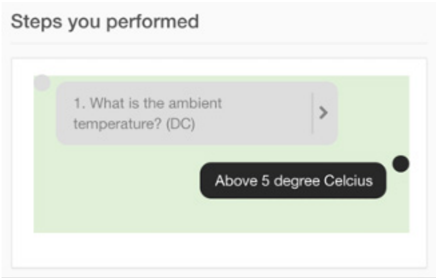
The Data Collection Framework (DCF) aims to provide a simple but powerful and flexible mechanism for data collection that can be inserted automatically into troubleshooting guides. The framework is designed to be used when encountering a step in a guide that can be answered automatically, and the client applications may request an answer to the step from DCF.
Every time there is a question in a troubleshooting guide there is an opportunity to automate the answer to that question, thereby saving time for the technician. Automatically answered questions optimize the troubleshooting sequence significantly as it takes the technician closer to a solution very fast. Every automatically answered question is never presented to the user – it all happens in the background with no user interaction. The user is, however, able to see what answers were provided by the system.
The framework obtains data through individual Data Collectors that are built to get relevant pieces of information like product configuration and product state (asset state data is extremely powerful in troubleshooting situations). The data is evaluated in an interpretation layer that transforms the raw data into meaningful answers in the guides. The knowledge base administrators configure everything.
Between the data collectors and the guides lies the data interpretation layer. This layer is where the translation from raw data to guide answers takes place. For example, voltage readings, environmental readings, CRM information, and any other data source is translated into understandable human terms like “perfect throughput” or “bad weather.”

Most customers use the Data Collection Framework (DCF) to dramatically increase precision and shorten Mean Time To Repair. By injecting data from CRM, SCADA, or live equipment data, the asset is basically saying, “here I am, here is my state,” and the troubleshooter will pick up that data and guide the user onto the best possible path to the solution.

By abstracting away difficult manual evaluations of complex data, the DCF ensures that we troubleshoot uniformly – we solve the same issue, in the same way, every time (and we save a ton of time in training, as new employees don’t have to learn how to make sense of complex data)

The DCF is the integration point for using IoT equipment data, CRM data, contextual data like ambient temperatures and conditions, and much more.

The DCF framework is available for all Dezide clients to build and maintain their own collection of Data Collectors



A green background indicates a question answered automatically by the Troubleshooter

The Liebherr Case

Liebherr is one of the world's largest manufacturers of construction machinery and an extraordinarily large range of products across eleven product divisions. Liebherr has been using Dezide in the Mining division for troubleshooting Mining Excavators and Mining Trucks since 2012.

LIEBHERR

Dezide has been branded as the "Liebherr Troubleshoot Advisor" (LTA) both internally in Liebherr and externally with affiliates and customers.

Liebherr is using Dezide as:

1. An internal tool for Field Service Engineers accessing the tool using both the online and offline interface when performing installation, commissioning, service and troubleshooting on mining equipment. Liebherr uses the standard Dezide Offline Troubleshooter branded using LTA styling.

    a. A Windows Phone App is available, built entirely by Liebherr IT on top of the Dezide Web Service API for providing an intuitive troubleshooting experience for their Windows Phone users.

2. An external tool for affiliate Field Service Engineers performing on-site service at customer locations.

3. A training tool used in the training classes held at the mining equipment factory in Colmar, France. Liebherr Field Service Engineers, affiliates and customers come here for training and they use the LTA as part of the training program.

4. A service offering to end customers. The LTA has been made into a viable product that Liebherr is including as a part of their service offering to customers. The customers can purchase access to limited parts of the knowledge base, where they are allowed access to content relevant to the machines they have purchased.

Results

Liebherr has currently built more than 2000 Dynamic and Static Guides. The Mining division has been so successful in achieving a reduction in troubleshooting time by more than 50% and a significant increase in first time right, that other divisions in the Liebherr Group are now joining the project.

Liebherr is having a hard time finding and training skilled people to service the machines and it is very expensive and inconvenient when people leave. They have found that being able to retain knowledge in LTA is a massive advantage.

Finally reselling the knowledgebase to customers represents huge revenue opportunities.

Summary

DEZIDE HAS A DEMONSTRATED HISTORY OF COMBINING THE KNOWLEDGE OF YOUR LEADING TECHNICAL EXPERTS WITH DEZIDES AI-POWERED TROUBLESHOOTING TECHNOLOGY TO DELIVER THE MOST **EFFICIENT** ADVICE TO THE ENTIRE GLOBAL WORKFORCE.
EFFICIENT ADVICE THAT WILL GET NEW TECHNICIANS UP TO SPEED FAST, TRANSFER KNOWLEDGE INSTANTLY, REDUCE TROUBLESHOOTING TIME, REDUCE THE NUMBER OF SPARE PARTS USED AND REDUCE THE ENVIRONMENTAL FOOTPRINT TOO.

Highlighted benefits:

Resource Gap and Skills

The service industry in general lacks technicians and more importantly, skilled technicians. We have seen a paradigm shift from a mechanical industry to a more electrical and digital focused industry, which puts a big strain on training and addressing the skill and resource gap in the industry.
Dezide shortens training time for new technicians by providing a platform that offers a structured, organized, and step-by-step approach to assist new technicians. By breaking down complex tasks into smaller, more manageable steps and providing a structured approach to troubleshooting (and thereby learning), technicians can become familiar with their tasks more quickly and with fewer mistakes. Additionally, Dezide can help technicians retain information and apply it to their daily tasks. Dezide helps shorten training times for new technicians, resulting in more efficient onboarding and better overall productivity.
Dezide is designed to shorten training time for new technicians by using bayesian networks to provide technicians with a logical and systematic way of diagnosing and resolving technical problems quickly and accurately.

Faster and more precise troubleshooting

Using Dezide allows new technicians to quickly and precisely identify the root cause of an issue. Specifically, Dezide guides technicians through a step-by-step process of asking questions and gathering data to arrive at a solution. This allows technicians to quickly identify the problem, diagnose it, and take corrective action. This saves time, increases efficiency, and reduces the amount of trial and error that may be required when using traditional troubleshooting methods.

Fewer non-defect parts replaced

Our customers are measuring that the technicians using Dezide replaces fewer and much cheaper components than the technicians using traditional methods. They are seeing that with Dezide only the faulty parts are replaced, and no unnecessary steps or resources are used as much less non-defect parts are used. The reduction in consumed spare parts reduces warranty costs and the overall environmental footprint as well.

Contact

We love to engage with our growing community of people passionate about troubleshooting and knowledge management. People who share our vision for capturing and sharing expert knowledge.

Reach us through these channels:

Visit our website at http://www.dezide.com

Email us at info@dezide.com

Follow us on LinkedIn https://www.linkedin.com/company/100934/

DEZIDE ApS

Gasværksvej 5
DK-9000 Aalborg
Denmark

Phone: +45 9655 73 70
info@dezide.com
www.dezide.com