MARY KAY O'CONNOR PROCESS SAFETY CENTER
TEXAS A&M ENGINEERING EXPERIMENT STATION

22nd Annual International Symposium
October 22-24, 2019 | College Station, Texas

Unified Hazard Assessment – Bringing Together HAZOP, LOPA, Hazard Registers, and Bowtie in a Unified Structure

Edward M. Marszal
Kenexis

*Presenter E-mail or Presenters E-mails: edward.marszal@kenexis.com

Abstract

Process hazards analysis (PHA) studies, especially Hazards and Operability (HAZOP) studies and Layer of Protection Analysis (LOPA) are ubiquitous in the process industries, but the information generated during these studies is not being used to its fullest extent. Management desires to have multiple scenarios rolled up into easier to use hazard registers, and to be able to visualize them with graphical approaches like bow-tie diagrams. Unfortunately, the data that is currently contained in most HAZOP and LOPA studies is not structured to allow easy hazard register generation and bow-tie visualization. The use of the "cause local – consequence global" approach to HAZOP has made the facilitator's life easier at the expense of others who need PHA data but can't find what they need because the resulting documentation is not logical. This paper discusses the use of standardized data structures that are already revolutionizing PHA documentation, and the extension of these standardized data structures will allow the development of tools that can display a single set of data as a HAZOP worksheet, LOPA worksheet, or bow-tie diagram. Furthermore, the hazard analysis can be modified using any of these visualization techniques and have the results carried back across all diagram and worksheet types. Adoption of a unified hazard assessment data structure will benefit all users of PHA data and facilitate use of PHA data by other critical stakeholders, automatically.

Keywords

HAZOP, LOPA, Bow-Tie, Big Data

Introduction

Chemical process facilities spend a considerable amount of resources on process hazards analysis studies. Initially, many organizations performed PHA in a minimal-cost minimal-compliance fashion, in order to "check the box" for regulatory compliance. Many bad habits were formed in this early phase of PHA adoption that are still with us today, but which can hopefully be rectified with a combination of better software and better motivated PHA facilitators. Since the early adoption phase of PHA, many sophisticated organizations not only committed to the PHA process, but also expanded the use of PHA information, leveraging the information to facilitate engineering and management tasks that were performed by other process stakeholders. Over the last 50 years, the process industries have gone a long way in implementing risk-based and performance-based decision-making processes to achieve higher levels of safety while minimizing the cost and effort of implementing the safeguards that these risk studies identify and define.

One of the first extensions of PHA information was LOPA. While PHA had come into wide adoption after the Process Safety Management (PSM) rule was promulgated by the US Occupational Health and Safety Administration (OSHA), not much was done with the information beyond simply following up on the recommendations that were generated during the study. One of the first evolutions occurred in the late 1990's after the International Society for Automation (ISA) released their ISA 84.01-1996 standard, *Application of Safety Instrumented Systems for the Process Industries*. This performance-based standard for the design of safety instrumented systems requires the selection of a performance target that defines the required level of performance of each safety instrumented function (SIF), which is essentially a control loop that performs a safety action. The determination of this performance target, referred to as the safety integrity level (SIL) is a natural extension of the PHA. The LOPA process which became popular as the methodology to select this performance target starts with PHA information such as the causes, consequences, and safeguards of a hazard scenario and expands them in a semi-quantitative fashion to ensure tolerable risk is achieved by assigning the portion of the total required risk reduction that needs to be provided by the each SIF.

Beyond LOPA, managers of process plants were interested in summarizing the information from PHA into lists of significant hazards. These managers would regularly refer to these hazard lists and ensure that all of the safeguards impacting the risk of these scenarios were properly design, implemented, and maintained. These lists of significant hazards and information relating to them are typically referred to as hazard registers. While hazard registers became a powerful tool for management of risk, understanding the hazards and communicating them throughout the organization and even external stakeholders was difficult due to the highly technical nature of the information and the dense use of acronyms and jargon. In order to aid in the understanding of hazard scenarios, the visualization technique of bowtie diagrams was developed. When a hazard scenario is visualized as a bowtie diagram, a graphical representation of the development of the scenario is developed. The bowtie diagram reads from left to right, as shown in Figure 1.

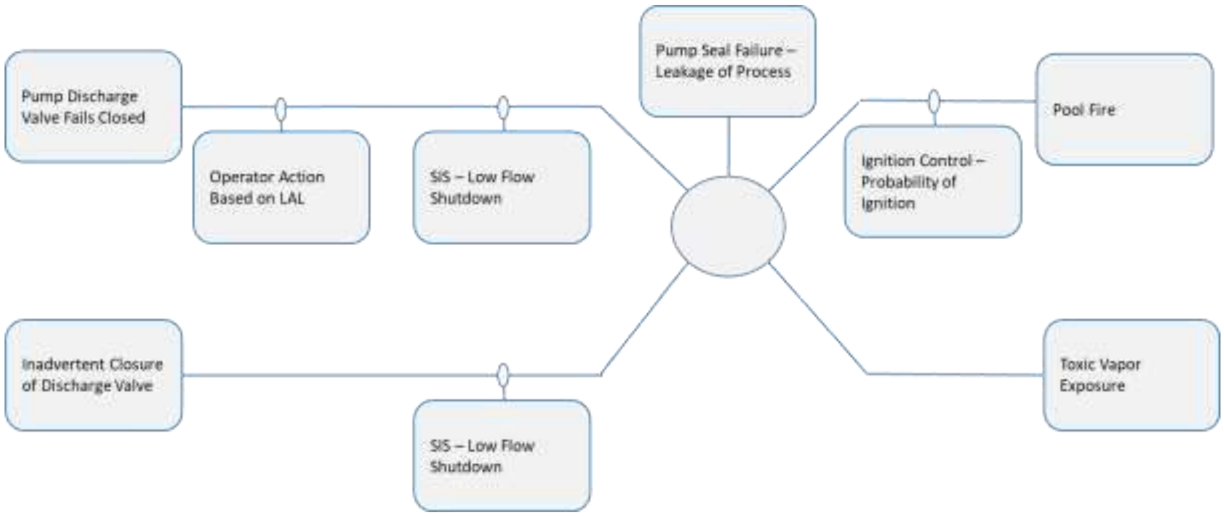

Figure 1 – Typical Bowtie Diagram

On the right a series of causes of the hazardous event are listed. Then a line is drawn from each cause to the "knot" in the bowtie. The knot in the bowtie represents that realization of the hazardous event. In the process industries the "knot" is typically a loss of containment of process chemicals. Along each line from the cause to the bowtie knot are "barriers". Barriers are events, systems, or situations, that prevent the cause from result in the loss of containment scenario – which in bowtie analysis jargon is typically referred to as the hazard or hazardous event. Barriers between the cause and the loss of containment are referred to as "preventive" barriers as their successful operation prevents the loss of containment from occurring. On the right side of the bowtie knot is the description of what occurs after the loss of containment. This will consist of a series of consequences that can occur as a result of the loss of containment event, with lines drawn from the "knot" to the consequences. As with the causes, all of the consequences may have barriers that will either prevent a mitigate (i.e., reduce the magnitude) of the consequence if they effectively operate. The key benefit of the presentation of a risk scenario as a bowtie diagram is that the visualization accelerates and facilitates understanding of the scenario as the combination of the visual cues and textual information provide a much richer representation of the hazard scenario information.

The preceding discussion clearly shows that there are myriad valuable uses for the information developed during PHA. It would seem that you should be able to simply press a button, and switch from viewing the data as a PHA, to viewing it as a LOPA, then a hazard register, and finally as a bow tie diagram. Further, it seems like you should be able to make edits to the bowtie diagram and have them cascade back the to LOPA study, and the you should be also be able to view the quantitative aspects of the LOPA while you're viewing the bowtie diagram. Unfortunately, this is not currently the case.

At the current moment, each of these different uses of PHA data use different software applications, each of which use different data structures to represent data, and even worse, the data structures used in the different applications for different study types are completely and irreconcilably different from each other such that data cannot be shared transferred between applications. In order to address this situation and maximize the utility of the data that the process industries are investing a few changes will need to be made in how PHA is performed and documented. First, a change in the mindset about how PHA facilitators document studies will be required. Facilitators must be willing to document studies based on where the hazard manifests itself as opposed to the location of the cause of the hazardous event. But most importantly, industry needs to move to a common, consistent, standardized data structure for PHA. The authors have named this standardized data structure and process the Unified Hazard Assessment.

Problems with Existing PHA Documentation

While the demand to make use of existing PHA data is large and growing, safety professionals who are attempting to make use of the after the completion of the PHA study are having trouble mining the reports and arranging the data they contain into a format that they can use for other purposes. The reason that most PHA studies are hard to use for other purposes is that the method used to generate the data optimizes the speed at which a study is conducted, and not the usefulness of the data. In order to make PHA systematic and fast, many facilitators started to implement rules for how a study should be carried out. There is one rule that is very commonly used and is the root cause of why PHA data has evolved into a data representation that is difficult to leverage for other purposes. That rule is “causes local – consequences global”. Using the “causes local-consequences global” approach, a facilitator will lead the discussion of a study such that for a given deviation (or other discussion prompt, if HAZOP is not used), the discussion of the scenario would begin with identification of a cause for that deviation. The study would limit its scope such that only causes that occur in the node being analyzed would be considered and causes that occur outside of the node would be assessed and documented elsewhere. Once the cause is defined, the “causes local – consequences global approach” would then require the team to chase the consequence of the deviation wherever it would manifest itself

This results in a data structure that can be referred to, in terms of data science, as “cause-indexed”. This means that the cause is the critical data item by which the balance of the information in the study is sorted and related. Unfortunately, when PHA data is desired to be used for other purposes, the starting point for searching for information is almost never the cause. In the case of a LOPA, especially one that is used to pick the SIL performance target for a SIF, all of the causes that generate a given consequence scenario that a SIF is intended to protect against must be combined. For LOPA, it would be most beneficial to index the PHA data by consequence (i.e., “consequence-indexing”) as opposed to by cause. In fact, cause-indexing generates myriad problems and additional hours of effort during LOPA because the causes (or initiating events, as they would be referred to in a LOPA) are scattered all around the PHA study instead of being contained in a single location. For example, if a LOPA scenario is interested in the overfill of a process vessel, like a compressor knock-out drum, the causes of that overfill often occur not only in multiple deviations in the PHA node that contains the vessel, but frequently causes of the overfill are documented in different nodes upstream and downstream of the vessel. Optimally, all causes of a high level should be documented in the high-level deviation of the node the contains the vessel, but due to facilitation and documentation techniques that are commonly in use today, this is virtually never the case.

The problem of the causes of a hazardous event rarely being documented where expected is further compounded by the order in which deviations are addressed by the PHA facilitator. Most facilitators begin with deviations related to the "flow" parameter, this then devolves the study into what the authors call a "poor man's FMEA", because this approach inevitably results in the facilitator and/or team finding every valve, pump, and compressor in the circuit and having it open/close or start/stop. This then results in the vast preponderance of hazards being documented as flow issues while the actual hazard is related to an entirely different parameter altogether. Going back to the example, the causes of high level in a compressor knockout drum would often not be documented in the high-level deviation of the node that contains the vessel. Instead, the causes are scattered, and usually the most important cause will be found in the low-flow deviation of the node that is the destination of the liquid from the knockout drum.

Another large problem preventing the more wide-spread use of PHA data in other applications is inconsistent data structure. Many facilities have been performing PHA studies for dozens of years, having gone through several rounds of revalidations. The unfortunate truth is that many of these studies were done using different software tools, and even for those studies that were performed using a single software tool, the data structures can vary because different PHA facilitators customized the worksheets to show different kinds of data, usually based on their personal preferences. While initially, it seems that having a PHA tool allow for the customization of the data structure to allow a study to specifically meet the requirements of particular facility is a great benefit, ultimately, the inconsistency between studies prevents being able to "roll up" and dashboard information at a single site, let alone across a multi-site organizations.

Basics of Structuring Data

Before delving specifically into the problem of defining a consistent structure for PHA data, we should begin by discussing how data is structured in general. While this section will rely on terminology that was developed for relational databases, it is applicable to other methods for data storage and retrieval that are vastly superior to the use of traditional relational databases for hazard analysis studies. The primary concepts of relational databases for storing data are tables, records, fields, relationships, and identifiers (ID's).

A table is a structured list of data that all describe a given item, for instance, a PHA study will generally have a table that describes all of the study nodes. The best way to think about a data table is by visualizing a spreadsheet. A table contains many rows and columns of information where each column of information is a specific piece of information, such as a Node Number or a Node Description. A database will generally contain many different tables that describe the different aspects of a system. For example, a PHA database might contain a table for nodes, deviations, and causes. The reason that different tables are required is because, often, there are multiple instances of one type of item that all relate to a single instance of another. With PHA, for instance, a single node will have many different deviations associated with it and a single deviation can have many causes. In order to address this phenomenon, we set up multiple tables of information, and organize the linking between the tables using relationships.

In database science there are three types of relationships – One-to-many, many-to-one, and many-to-many. A one-to-many relationship means that for a single item in the primary table, there can be many items in the secondary table that are associated. In other words, the primary table is the "parent" in the relationship can have many "children" in the secondary table. In a typical PHA, a single node will have many deviations and each deviation will have many causes. A relational database will manage these relationships first by understanding how they are defined, and then by tracking the IDs of the records. Consider causes. When one looks at a PHA report a single deviation will contain multiple rows that contain causes of the deviation. While it looks like a single table to the viewer, in reality, the software is combining the two different tables into one visualization. The deviation table, in fact, does not actually contain any information about causes at all! Instead, each cause contains the identifier (ID) of the deviation that it is associated with. This activity is referred to as cause-indexing. What the PHA software does in order to display that view that most of us are familiar with is to first display the deviation from the deviation table and subsequently search the causes table to find all of the causes that are associated with a given deviation. It does this by comparing a "Deviation ID" field in each cause record with the ID of the deviation that is currently being displayed. A many-to-one relationship is the opposite of one-to-many and works very similarly. A many-to-many relationship between tables is the most complex situation where a single record in the primary table can have multiple associated records in the secondary table, but, individual items in the secondary table can also have relationships with multiple records in the primary table. The best example of this in PHA is recommendations. For each PHA cause (scenario) can have multiple recommendations associated with it, but each of those recommendations can also be associated with multiple causes. In this complex case an entirely separate table needs to be created simply to store the relationships between the primary and secondary table.

As alluded to earlier, each table will contain multiple records. Each record is like a row on a spreadsheet and contains all of the different attributes of a specific entry on the table. Each record then is composed of multiple different fields. Each field is a specific data entry of a given data type. For instance, if a PHA study contains a table of safeguards, that table might contain individual fields for tag (text string), description (text string), probability of failure on demand (floating point number), Effectiveness Determination (Boolean) and ID (GUID). All of these structures combined together result in the overall database structure.

Better Options for Holding Relational Data During PHA Studies

The discussion in the preceding section used terminology that is consistent with relational database technology. While the discussion is applicable to relational databases, and relational databases are very commonly used to store PHA data, use of relational databases is becoming obsolete in applications like PHA studies. There are better ways to store and work with data that have been developed by the leaders of internet technology. All of industry, and all of society, is migrating to a paradigm where information is stored in the cloud, and all knowledge work will be performed by interacting with the cloud, generally through a web browser. Even when desktop applications are developed, they are generally a thin wrapper that essentially holds a web page. In this paradigm, relational databases work great when the user would like to interact with a single record of a single table in the database, but when a user desires to work with and view multiple different records of multiple different tables, the result is a dreadful and slow user experience, as the web page tries to kludge up a concatenation of all of the requested information in a single form. This form then updates every time the user shifts focus from one element to another – especially if anything was edited, because that is what is required to maintain contact with the database server.

The primary problem is the transactional nature of a relational database. When a user requests a piece of information, the database must know specifically what table, record, and field to get. It then grabs that piece of information transfers the data between server and client, and the client processes the information by presenting it on the screen. Well, as we described in the previous section, in order to present a PHA worksheet on a computer screen, data needs to be obtained from multiple fields, in multiple records, of multiple tables. Furthermore, the drawing of the information on the screen is complicated by the fact that the application need to change the view as a function of how many records in secondary tables are associated with a given record in a primary table. This requires thousands (or even tens of thousands) of individual database transactions to occur between the client and server to obtain the information for a single screen view. The problem is usually further exacerbated by the use of third-party "controls" during the programming process. Many software vendors do not have the talent to directly access the database, and instead rely on third party controls that they configure to access the data that they would like. Unfortunately, these controls – such as text boxes, and grids – have bloated and slow code because they are designed to be flexible, not fast.

The elite companies in Silicon Valley have developed better ways to handle this problem that are in frequent and widespread use in their software and web sites. Unfortunately, most PHA software has not caught up with the times. The state-of-the art approach is to eschew traditional relational database technology in favor of the flexible data object models that were born in cloud computing. Specifically, the data transaction speed problem was solved using eXtensible Markup Language (XML) and subsequently its even lighter cousin JavaScript Object Notation (JSON). In the new paradigm, when a web page wants to get data from a database server, it doesn't request a single field at a time, it requests that a large collection of data is “serialized” into a JSON object, and that single object is conveyed from server to client in a single transaction. As a result, a best-in-class cloud based PHA application will only communicate with the server twice per worksheet – once to load the data from the server, and then once again to return the edited data back to the server. In the interim, the web page keeps the entire data object in memory on the client. When the user interacts with the data, they are interacting with the data on the client – at lightning speed, not the data on the server.

New data structures like XML and JSON have all of the advantages of a relational database. They can easily store multiple tables, each with multiple records and multiple fields. They can also manage relationships between tables, in all formats, the same way that relational databases do. Figure 2 presents an example of some PHA data being stored as a JSON object.


```
{       “CauseID”: “1”,
        “Cause Description”: “Pump Failure”,
        “Safeguards”:[
                { “ID”: “1”, “Tag”: “LFL-01”, “PFD”: “0.1”},
                { “ID”: “2”, “Tag”: “FZC-02”, “PFD”: “0.01”}
        ]
}
```


Figure 2 – PHA Data in JavaScript Object Notation (JSON)

While some applications are built with a relational database server to server up the data to build a web page, and then store the final results after the page is edited. More and more, the relational database is not used at all, and the JSON objects are simply stored on the server as the end result.

Unified Hazard Assessment Data Structure

PHA studies have been formatted in a multitude of different ways, generally to speed to performance of the study as opposed to optimizing the usefulness of the data that is generated. A single structure is needed to allow for the multitude of uses of the data that is developed in the PHA. The first, and most critical step, is to get agreement among end users on what data should be tracked in a PHA, and how that data will be organized. The best way to standardize the data format would be to create and publish a JSON schema. A scheme is simply a formal description of how data in a data object is defined and structured. The Center for Chemical Process Safety would be a perfect custodian for such a data structure. Other groups have already started work on creating a standardized data structure, specifically the Purdue Process Safety and Assurance Center (P2SAC) of Purdue University has already embarked on research and made publications related to development of a standardized data structure.

In the language of relational data, we all need to agree on what tables are required, what fields are required for each table, and how the records of all of the tables are related to each other.

Once agreed, a standardized structure can be published as a schema. It may sound easy but getting agreement on what a PHA should look like is virtually impossible. For instance, many organizations only calculate and present a risk ranking after safeguards have been considered and listed. Other organizations like to calculate and present a separate risk ranking before application of safeguards and then after. It would seem an impasse has been created, but there is a work-around. The use of super-sets.

A super-set is a definition of the complete set of fields that might be desired for a given record. The super-set would contain every possible item that anyone might want to use. So, in the case of risk ranking before and risk ranking after, using the super-set approach, both would be included in the table. Even though the super-set of fields is available for use, they are not required to be used. Applications that use the data structure can be configured to only display the portion of the super-set that any particulate user is interested in viewing. As a result, even though two different users create different views of their studies, the underlying data structure is identical. The effort that industry must undertake is to determine what that super-set of fields is. This is the subject of the P2SAC study and ongoing industrial and academic research.

While the use of super-sets can address the issue of what fields are required for each record of each table, the problem of which tables are required, and what are the relationships between the tables is an area where we will have to agree. A single data structure cannot be both cause-indexed, as is the most common approach for HAZOP, and also to be consequence-indexed, as is the most common approach for LOPA. Furthermore, neither of these data structures is suitable for Hazard Registers or Bowtie. For these two types of studies, the data is indexed by a "Hazard" or a "Hazard Scenario".

Unified Hazard Assessment is based on a data structure that is "Hazard Scenario" indexed. This realization came to the authors while reviewing the typical structure of a bowtie diagram. In a bowtie diagram, there are many causes, each cause results in a singular Hazard Scenario. A hazard scenario is generally a loss of containment event in the process industries. It is the point where control of the process is lost and material from the closed process escapes to atmosphere. In the terminology contains in the CCPS Guidelines to Chemical Process Quantitative Risk Analysis, this is the "Incident" that follows an initiating event and intermediate events, and is followed by multiple difference incident outcomes, each of which has a different consequence associated with it. In this way, we see that consequences are not children of causes, and causes are not children of consequences. They are, in fact, siblings to the parent of hazard scenario. This hazard scenario is also the key starting point for the hazard registers that are inherently indexed by hazard scenario. Thus, the unified hazard assessment data structure will be arranged with the core data contained in the following tables: Hazard Scenarios, Causes, Consequences, and Safeguards. There will be a one-to-many relationship between Hazard Scenarios and Causes and also a one-to-many relationship between Hazard Scenarios and Consequences. There will be a many-to-many relationship between Causes and Safeguards and also between Consequences and Safeguards. The safeguard data structure super-set of fields will need to ensure that this type of record can be used either as preventive, i.e., preventing a cause from becoming a loss of containment, or mitigative, i.e., mitigating the magnitude of the consequence.

Implementation and Benefits of Unified Hazard Assessment

The Unified Hazard Assessment data structure provides a platform that allows a single data set to be used for multiple purposes. Implementation in software and work practices will require extension of existing paradigms to take full advantage. HAZOP studies can generally be performed using the same workflow as always, but some additional care will need to be taken in where things are documented and at least one additional data field will need to be completed, or at least separated out of the cause or consequence description, where it usually resides.

With regards to workflow, the authors suggest one primary change, that would be to discuss the flow deviation last, instead of first. This will allow for the hazard scenarios to more naturally line up with the PHA deviations that initiated their discussion. The efficient workflow of cause local – consequence global can continue to be implemented, with one big exception. When a facilitator is in a node and identifies a cause of a deviation, let's say low flow. As always, the PHA team would be required to chase that cause all the way out to where the hazard scenario manifests itself, let's say as a high level in a different node. Currently, that scenario would be discussed and documented as a low flow in the current node. Optimally, after implementation of Unified Hazard Assessment, the scenario will be documented and discussed as a high level in the node where the high level occurs. This could necessitate the facilitator documenting the hazard scenario and cause in the appropriate node and noting that the team will need to complete the discussion when they arrive at that node. This is the big required change in behavior where a great degree of pushback is expected.

With regards to data structure, the HAZOP worksheet will need to show, for each deviation, one or more hazard scenarios. Each hazard scenario will allow multiple causes to be shown, and will also allow multiple consequences to be shown. Other than that, the PHA worksheet will look essentially the same but providing filtering on the fields of the super-set that are shown to be limited to those that are appropriate for HAZOP and also allow user customization of view.

Once the HAZOP has been documented using the Unified Hazard Assessment approach, development of the LOPA is dramatically simplified. The data structure for each hazard scenario record should include a Boolean variable that indicates whether or not the hazard scenario requires a LOPA. In this way, when the user starts by viewing the HAZOP study, they should be able to click on a single tab or button and have the software automatically redraw the user interface for LOPA. The first step in creating this view is to filter the HAZOP scenarios so that only those that were selected for LOPA are shown. Also, the sub-set of fields displayed from the various tables will be altered to be appropriate for the LOPA task. For instance, the fields shown for safeguards in the HAZOP might only have included a description, but in the LOPA, the list may also need to include tag number, SIL level, and probability of failure on demand. One area of complication is the safeguards that are used as consequence mitigation, or activate after the loss of containment occurs will need to be handled differently from preventive protection layers. In a LOPA these layers are commonly referred to as consequence modifiers. This is an area of research that will be discussed at the end of this paper.

In addition, and at any time after HAZOP or even after LOPA, the data should be viewable as a bowtie diagram. Bow-tie diagrams should be able to be automatically generated from the Unified Hazard Assessment data set, because it is specifically designed for that purpose. The Hazard Scenario that the Unified Hazard Assessment is indexed by forms the foundation of the bowtie diagram. It is the “knot” in the bowtie that connects all of the causes to all of the consequences. The bowtie diagram begins with selection of the hazard scenario. Then, for each hazard scenario, all of the causes are determined and drawn on the worksheet. For each cause, all of the preventive safeguards that are associated with the cause will be drawn on the diagram. Each cause/associated safeguards set will be connected to the hazard scenario, or bowtie knot. Then, all of the consequences will be developed and listed along with all of the mitigative safeguards that are associated with each consequence for that is associated with a given hazard scenario. In this way, the bowtie diagram can be completed generated from either the HAZOP data or LOPA data that was created in the traditional tabular format of the HAZOP and LOPA worksheets. There is an even more powerful aspect to bowtie diagrams using the Unified PHA format. Currently, most bowtie diagrams are simply a visual representation of a hazard scenario, but with an underpinning of a Unified Hazard Assessment data structure, the bowtie diagram can include all of the quantitative aspects, such as initiating event frequency, safeguard probability of failure on demand, and overall scenario frequency and risk ranking. This will enable performing a HAZOP or LOPA using the graphical format of the bowtie diagram, which provides the benefit of enhanced understanding from meeting participants.

The last information presentation format that needs to be addressed is the hazard register. This is the easiest of all of the problems to solve after the development of the Unified Hazard Assessment data structure. Basically, the data the is developed in a HAZOP worksheet is sufficient to meet this need, as long as it is hazard scenario indexed. In fact, one might want to limit the data presented to even less than what is shown in a HAZOP worksheet. The only additional consideration for the development of the hazard register would be another Boolean variable for each Hazard Scenario that indicates that the scenario is significant enough, usually as the results of its consequence category, to merit display on the hazard register.

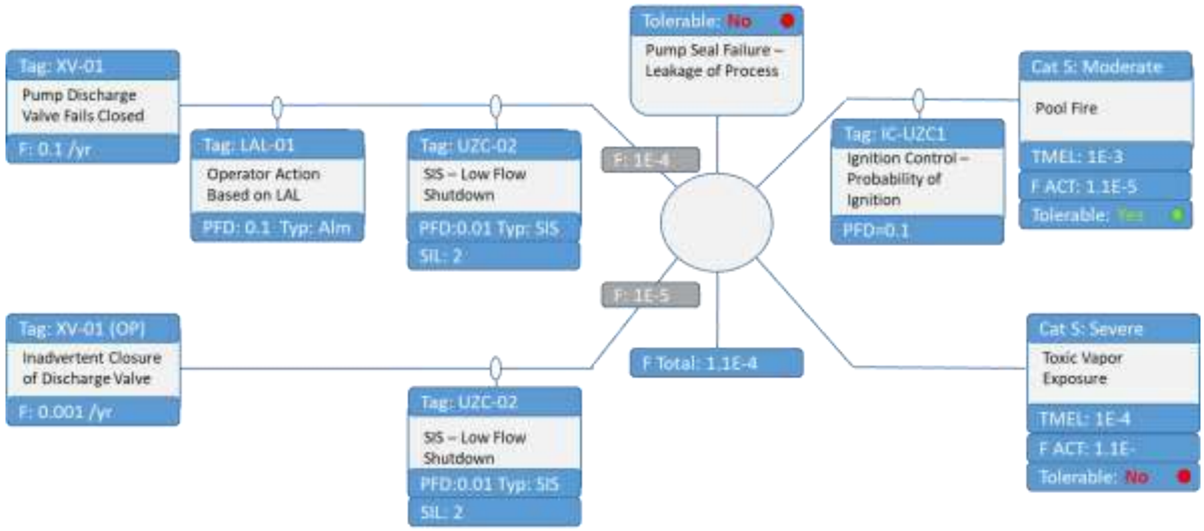

Figure 3 – Bowtie Diagram Including Quantitative LOPA Data

Future Development

While the concept of the Unified Hazard Assessment has been significantly developed, there are areas where more research and development need to be applied to further enhance the usefulness of its data structures and applications. The primary area where this is true is the handling of mitigative safeguards. While current methods for bowtie diagram development and HAZOP can easily show mitigative safeguards, LOPA cannot effectively address them. In LOPA safeguards are only "credited" in terms of the amount of frequency reduction that they can provide in terms of their probability of failure on demand. Safeguards the reduce the magnitude of consequence cannot currently be addressed in the LOPA method. As such, only safeguards that activate after the loss of containment that reduce the consequence by effectively 100% can be credited. Others are ignored. What is required is the ability to document and calculate risk for safeguards that not only have a probability of failure that they will activate, but also concede that they will not reduce the consequence to zero, but something that is lower than the consequence if they did not operate. The research of the authors is related to including an addition factor to mitigative safeguards that allows input of not only the probability that the safeguards will fail to operate, but also the consequence category that will be achieved if the safeguard does activate.

Conclusions

The PHA documentation methods that are in use today are still a relic of optimization of the thoroughness and speed at which PHA studies could be completed that was necessary during the early "compliance" phase of implementation of PHA. Since the compliance phase of PHA adoptions, many process industry management teams found value in the data developed during the PHA for other purposes that include overall hazard management, reliability centered maintenance of safeguards, and visualization of hazard scenarios by various stakeholders in these facilities. The traditional methods for HAZOP documentation and study performance resulted in data that was not usable for these other purposes. In order to allow PHA/HAZOP data to be used for other purposes changes need to be made not only in how the studies are performed, but also in how the data in structured. Unified Hazard Assessment is a method to restructure and optimize PHA data so that it can be used for other purposes, beyond just process safety management compliance. A standardized Unified Hazard Assessment data structure and complaint methods for documenting PHA data will allow for a single data set and a single software tool to be able to seamlessly present data as PHA (HAZOP), LOPA, Hazard Register, and Bowtie Diagrams.