# Preventing Future Oil Spills with Software-Based Event Detection

**S.S. Iyengar, Supratik Mukhopadhyay, Christopher Steinmuller, and Xin Li,**
***Louisiana State University***

**Complex event processing systems detect problems in mission-critical, real-time applications and generate intelligent decisions to modulate the system environment.**

On 20 April 2010, the BP-owned, Transocean-operated Deepwater Horizon oil-drilling rig exploded, killing 11 workers and injuring 17, and sending massive quantities of crude oil riddled with lethal toxins from the sea floor into the Gulf of Mexico. Geophysicists, Earth-space scientists, and policymakers continue to debate the cause of the explosion, how much oil has been released, how best to contain it, and the long-term impact.

As BP made one futile attempt after another to cap the well, hundreds of millions of gallons poured into the gulf, eclipsing the 1989 *Exxon Valdez* spill in less than a week and causing incalculable environmental damage. More than 2,500 animals have already been killed, including hundreds of endangered sea turtles, and the marshes and wetlands that protect Louisiana's coast from erosion are seriously threatened. Scientists also fear that a hurricane could churn the oil-soaked waters, endangering many inland areas and landmarks. And should the oil reach the Gulf Stream loop current, it will wash up on Florida's beaches and may even reach the Atlantic seaboard.

The oil spill has also decimated the local economy, much of which relies on fishing, tourism, and deepwater drilling (banned by the federal government in the wake of the spill), and the ultimate cleanup, reclamation, and litigation costs will be in the billions of dollars.

In short, the Deepwater Horizon oil spill is one of the largest and costliest in history, with far-reaching effects on the Gulf Coast that will be felt for decades to come.

## COMPLEX EVENT PROCESSING

While BP's "milestone maneuver" on 19 July to stop the flow of oil may ultimately succeed, it begs the question: could this awful tragedy have possibly been prevented in the first place? Instead of the chaos occurring today, could the company simply have ordered a maintenance crew to the rig or perhaps shut it down?

BP has released documents indicating that it had concerns about the rig as far back as mid-2009 as well as numerous warnings that something was amiss prior to the blowout. These warnings were apparently ignored in BP's decision-making process since the low probability of a disaster masked the risk associated with such action.

Complex event processing (CEP) systems detect events in mission-critical, real-time applications and generate intelligent decisions to modulate the system environment (D. Luckham, *The Power of Events: An Introduction to Complex Event Processing in Distributed Enterprise Systems*, Addison-Wesley, 2003). In the case of deepwater oil drilling, advanced CEP technology could have helped to prevent the current crisis in the Gulf of Mexico.

## CIM SHELL

At LSU, we've developed the Cognitive Information Management Shell (CIM Shell), a CEP system that can "drill down" into complex events and activities and adapt rapidly to evolving situations in a wide variety of environments (www.2theadvocate.com/news/business/93725659.html). By archiving past events and cross-referencing them with current events, the system can discover deep patterns and then act upon them. Agent-based techniques continu-


Published by the IEEE Computer Society

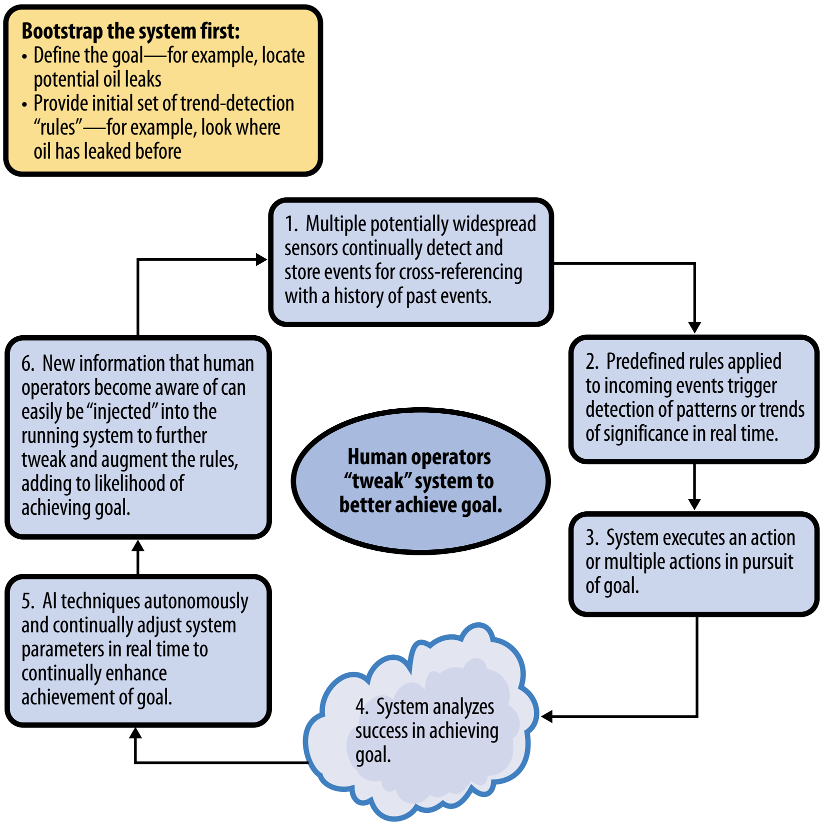


**Figure 1. High-level view of Cognitive Information Management Shell.**

ally adjust CIM Shell's parameters in near real time to adapt the system to changing environments, and human operators can easily add information to tweak the system, making goals easier to achieve.

Figure 1 shows a high-level view of CIM Shell. We've used the system in numerous scenarios including soil and water management and video control in the presence of frame losses. Deployed over a hardware cloud infrastructure with GPU accelerators, it can handle around 100,000 events and make a million inferences every second.

Using a distributed storage schema and middleware, CIM Shell provides persistent notification of events on top of reliable churn-tolerant group communication. The storage schema archives events as key-value pairs. An implementation of the Paxos algorithm (L. Lamport and M. Massa, "Cheap Paxos," *Proc. 2004 Int'l Conf. Dependable Systems and Networks*, IEEE CS Press, pp. 307-314) ensures eventual consistency over the network. Crawlers running 24/7 on the cloud infrastructure index video and other types of data offline. A variation of Friedermann Mattern's global virtual time (GVT) algorithm (F. Mattern, "Efficient Algorithms for Distributed Snapshots and Global Virtual Time Approximation," *J. Parallel and Distributed Computing*, Aug. 1993, pp. 423-434) allows for uniform interpretation of temporal events over a wide-area network.

Distributed intelligent agents (communicating state machines) can be created manually or automatically generated in a model-driven way and dynamically deployed over the network to meet mission requirements (R. Bharadwaj and S. Mukhopadhyay, "A Formal Approach to Developing Reliable Event-Driven Service-Oriented Systems," *Proc. 32nd Ann. IEEE Int'l Computer Software and Applications Conf.*, IEEE CS Press, 2008, pp. 227-230). The agents can fuse disparate streaming data—text, video, and so on—to provide a uniform basis for decision-making. An interactive/intelligent sensing, inspection, and visualization system provides real-time monitoring and analysis.

CEP systems can reason about causality, knowledge, belief, risk, and uncertainty (P. Singh et al., "A Risk Reduction Framework for Dynamic Workflows," *Proc. 2008 Int'l Conf. Services Computing*, vol. 1, IEEE CS Press, 2008, pp. 381-388) and are able to make decisions in the presence of incomplete information. CIM Shell can determine how effective the initial seed of information is and, along with a human operator, modify the conditions as warranted. Even rare events with low probability can trigger a response if the associated risk is high; the system will magnify the risk associated with an event related to a mission-critical scenario.

## OIL SPILL PREVENTION

Figure 2 shows how CIM Shell could have prevented an oil spill like that in the Gulf of Mexico. Had the system been integrated with the Deepwater Horizon infrastructure, it would have interpreted prior maintenance incidents and concerns voiced by BP's engineers as a complex event and magnified the associated risks. It would have advised remote console operators to shut down the rig for maintenance or could even have taken the action itself autonomously.

In a non-real-time situation, we tend to reason well about risks associated with a decision—for example, even though there's a very small chance of getting sick while traveling abroad, most of us would still buy travel insurance because the cost associated with such an illness could be enormous. The Deepwater Horizon disaster underscores the fact that sometimes events happen too fast for humans to cope with. It also shows that human decision making based on sensed data often isn't rational and tends to

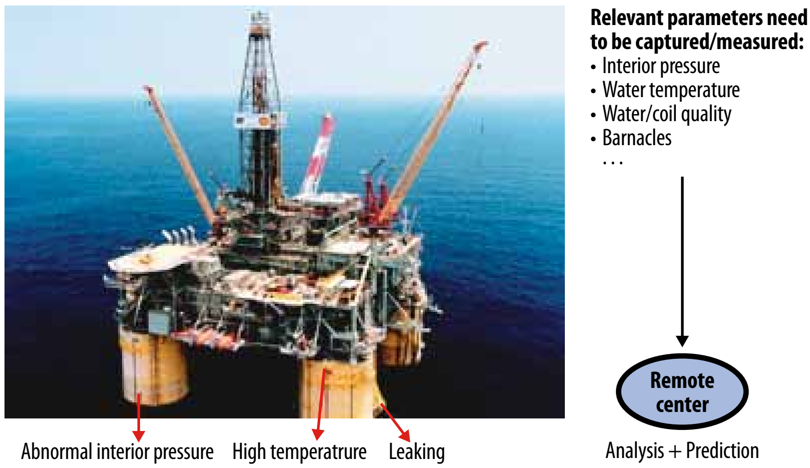


**Figure 2.** Had CIM Shell been integrated with the Deepwater Horizon infrastructure, it would have interpreted prior maintenance incidents and concerns voiced by BP's engineers as a complex event and magnified the associated risks.

underestimate risks, especially if rare events are involved. In such scenarios, normative decision making may not be an appropriate model.

Oil rig console operators are responsible for an asset worth $100 million to $1 billion, not taking into account the potential environmental and health risks of system failure; in comparison, a fighter pilot controls an asset worth between $15 million and $150 million. Each console operator receives information from about 1,500 sensors and can't possibly think clearly about the enormous risks associated with responding or not responding to events in such a data stream.

CEP engines like CIM Shell that sense events and respond based on provable decision-making theories can mitigate the problems arising from such limitations in human reasoning. They can help ensure safe operation in the petrochemical industry much like autopilot technology significantly contributes to safety in the aviation industry.

## OTHER APPLICATIONS

The practical uses of distributed event detection and monitoring and systems like CIM Shell are enormous, ranging from enterprise management to golf-course sprinklers to a hospital patient's intravenous pump. Some examples:

- Drone ships, robotic freighters with a small human crew that are capable of docking themselves and avoiding ocean hazards, could revolutionize the shipping industry.
- Advanced credit-monitoring systems that better detect anomalies in purchase activity could help prevent identity theft.
- Power-conservation systems could remotely shut down machines after long periods of inactivity or if no one is there to operate them.
- A security system could order an inventory after too many seemingly unrelated disturbances in a warehouse.
- Factories that detect too many defective parts could order the parts inspected and switch to a different stockpile.
- A fast-food restaurant manager could check consumption of each food item on the menu and stock the freezer accordingly.

CIM Shell can analyze and then act upon nearly any conceivable complex activity. Using distributed databases and capable of operating over a wireless network, the system is so intuitive and fault-tolerant that any type of user could create new rules or goals with the simple push of a button. One person could run an entire operation alone, or the system could operate autonomously from human input save for occasional inspection.

With CEP technology, BP likely could have prevented the Deepwater Horizon oil spill. Indeed, CIM Shell and similar systems being developed like Apama (http://web.progress.com/en/apama/index.html), may make such disasters mere memories in the future.

***S.S. Iyengar*** *is Roy Paul Daniels Professor and chairman of the Department of Computer Science, as well as director of the Robotics Research Laboratory, at Louisiana State University. Contact him at iyengar@csc.lsu.edu.*

***Supratik Mukhopadhyay*** *is an assistant professor in the Department of Computer Science at Louisiana State University. Contact him at supratik@csc.lsu.edu.*

***Christopher Steinmuller*** *is an undergraduate student in the Department of Computer Science and in the Department of Physics and Astronomy at Louisiana State University. Contact him at cstein1@lsu.edu.*

***Xin Li*** *is an assistant professor in the Department of Electrical and Computer Engineering and at the Center for Computation & Technology at Louisiana State University. Contact him at xinli@csu.edu.*

*The authors thank Daniel Walker of Praeses Corp. for helping to clarify the conceptual view of the CIM Shell from the application side and Ramesh Bharadwaj of the US Naval Research Laboratory for actively collaborating with them on this project.*

**Editor: Naren Ramakrishnan, Dept. of Computer Science, Virginia Tech, Blacksburg, VA; naren@cs.vt.edu**