DOI:10.1145/2461256.2461276

How to mitigate a cyber-physical attack that disables the transportation network and releases a cloud of chlorine gas.

BY NABIL ADAM, RANDY STILES, ANDREW ZIMDARS, RYAN TIMMONS, JACKIE LEUNG, GREG STACHNICK, JEFF MERRICK, ROBERT COOP, VADIM SLAVIN, TANYA KRUGLIKOV, JOHN GALMICHE, AND SHARAD MEHROTRA

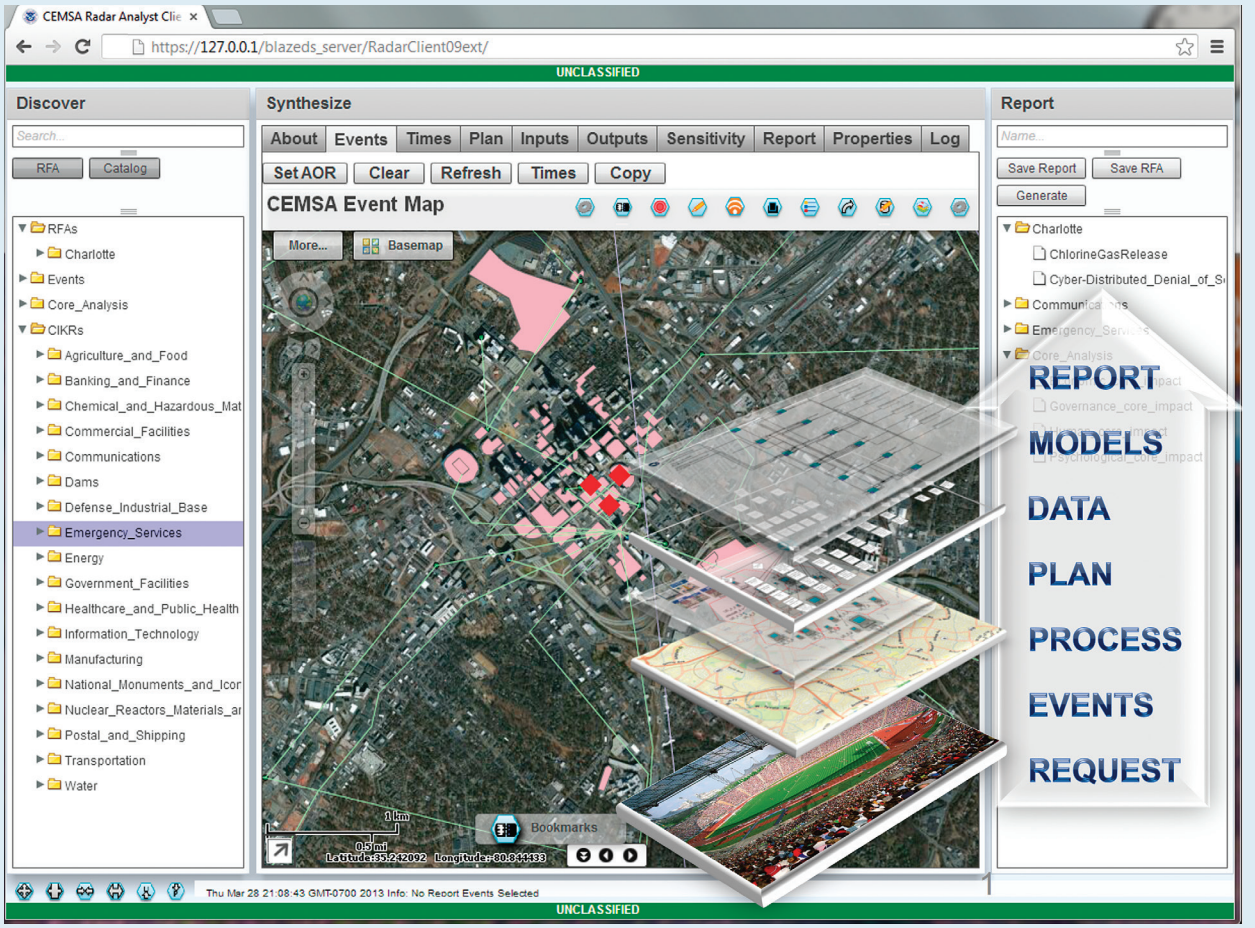
Consequence Analysis of Complex Events on Critical U.S. Infrastructure

U.S. DEPARTMENT OF Homeland Security analysts develop simulation models and tools to analyze the consequences of complex events on critical U.S. infrastructure and resources. An example of such an event is a coordinated cyber/physical attack that disables transportation and causes the release of a toxic chemical plume. The results can inform policymakers at the local, state, regional, and national levels. The Complex Event Modeling, Simulation, and Analysis, or CEMSA, program in the DHS Science and Technology Directorate is developing and deploying such a system to let analysts quickly integrate data, models, and expertise to arrive at credible consequence analysis of complex events. CEMSA aims to reduce turnaround time and costs, provide

» key insights

- **DHS analysts must perform their assessments quickly and therefore require tools that support quick response.**
- **CEMSA reduces the time analysts need for initial assessments while expanding the scope of simulations.**
- **As in any automated system, visibility is needed into where underlying data and models are available for inspection and modification.**

Figure 1. CEMSA enables rapid analysis of complex event consequences.



organic capabilities for risk analysis within DHS, enhance interoperability within DHS, and enable DHS to access and leverage the best available models within other government agencies, as well as within partner universities and industry.

Here, we start with the complex event analysis environment, followed by an approach to addressing them. We briefly present technical detail of some CEMSA components and discuss an example of its possible use in an interesting homeland security application—an evaluation of the consequences of cyber events on the physical infrastructure (see Figure 1).

Complex Event Analysis

CEMSA is being developed for an operational environment to support DHS strategic- and operational-level planners. Products and services in crisis response require analysis to be conducted within given time constraints to meet DHS leadership decision cycles. CEMSA addresses the following requirements:

Analytical.

▸ Enables estimation of disruption consequences, guides use and development of more detailed models, and integrates models;

▸ Allows composition of current and future models in an operational environment to determine direct and cascading effects resulting from multiple sources of infrastructure disruption;

▸ Enables analysis and simplification of systemwide models, allowing estimation of disruption consequences, guides use and development of more detailed models, and assists development of high levels of confidence in model results;

▸ Ensures transparency for clarity and ease of communication, enabling explanation of possible system behaviors;

▸ Confirms modeling and analysis sufficient to engender a high level of confidence in modeling and analysis results; and

▸ Quantifies errors and uncertainty that can arise when combining real-time data streams with models based on historical data or analytic abstractions.

Real time. CMESA enables real-time decision making through the collection and management of real-time field information, identify effects of new information on an analysis already under way, and project effects of new information, with updates from the field on a simulation's outcomes.

Architectural.

▸ Links models across and within infrastructures at various resolutions and spatial and timescales while accommodating various modeling languages and approaches;

▸ Enables on-the-fly integration of multiple, disparate models incorporating consequence and other analyses to address specific questions and provide

analytical ability scaled to available schedule and budget; and

▸ Applies well-defined "semantics" for describing models and simulations, leveraging and expanding the existing suite of analytical tools and capabilities and developing methods to incorporate existing and potential new tools.

Standards. CEMSA adheres to relevant industry standards, including those from the World Wide Web Consortium and from the Open GeoSpatial Consortium.

Approach

The nonproprietary, net-centric, enterprisewide CEMSA[a] delivers innovative capabilities for addressing these requirements: It is designed and implemented as a modular system based on open service-oriented architecture standards. A number of modular core CEMSA Spring Framework services are implemented on the back-end server to support consequence analysis (see Figure 2). The back-end server includes an execution-engine service based on Kepler/Ptolemy software[5] that supports combined execution approaches, a catalog that provides knowledge reasoning about models and simulation domains, a central database service based on PostgreSQL for persistence, an approximation engine for delivering initial time-critical analysis results, a planning service that generates candidate simulation plans for analysts, and an explanation engine supporting assessment of the results.

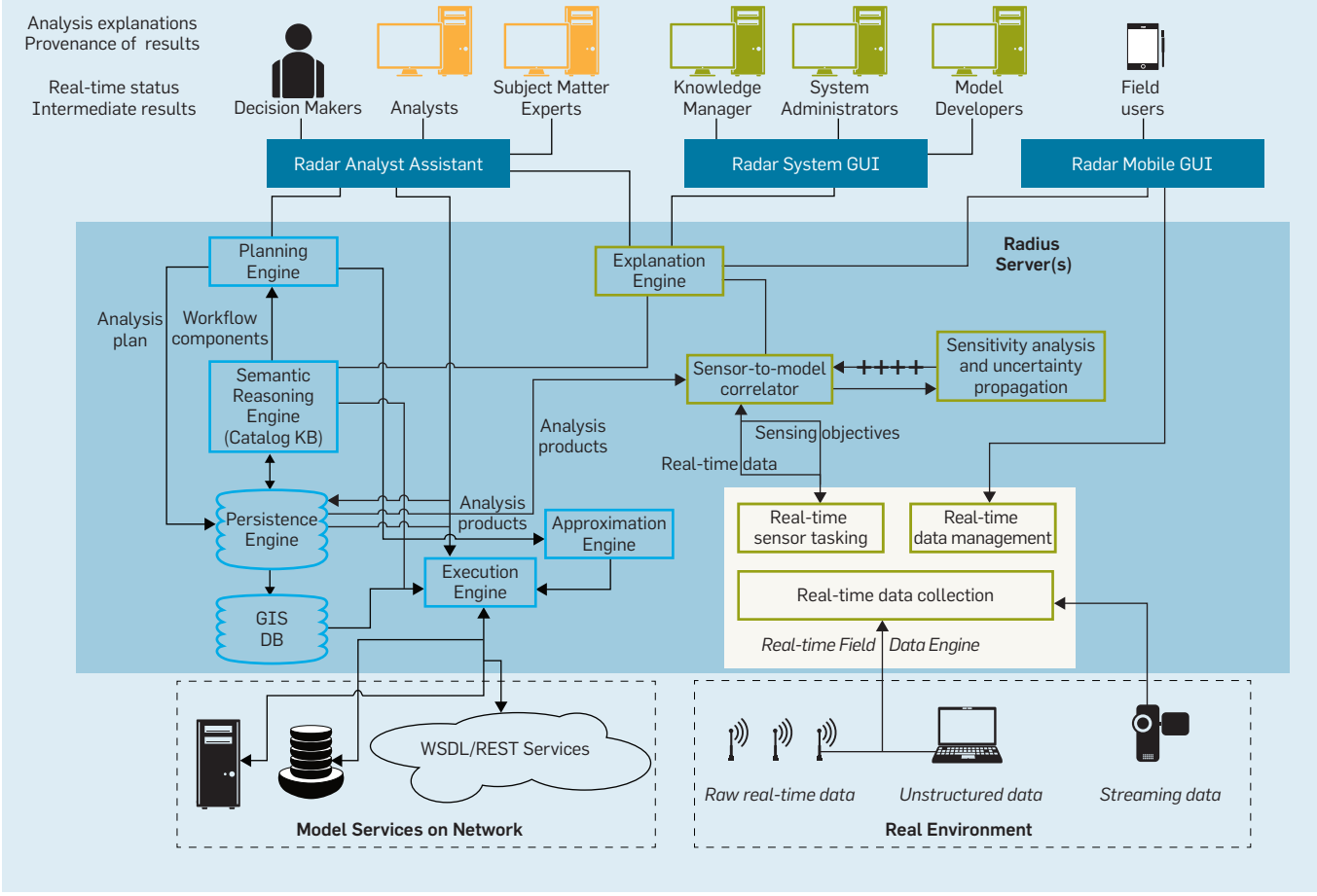
Net-centric analyst assistant. The Web-based CEMSA user interface helps analysts develop, manage, and assess the effects of multiple complex events, supporting their requests for analysis (RFA), delivering a consequence analysis report, and communicating with CEMSA back-end services for model composition.

The CEMSA front-end is based on a model-view-controller approach. Panels are laid out from left to right for overall analysis tasks of discovery, synthesis, and reporting. Discovery views include RFAs, data, models, experts, and ontology trees of Critical Infrastructure and Key Resources (CIKR), which are part of the United States National Response Framework. Synthesis views include events maps, inputs, outputs, sensitivity, plans, and system properties. Reporting views include report templates, generation options, and report editing.

When an analyst requests a plan, the constraints are passed to the planning engine, which returns a generated plan workflow for the analyst's review. After one or more iterations where the analyst identifies specific models, data, and experts, and receives refined generated plans, the analyst then executes the plan. Execution is coordinated by the execution engine, with calls to models on the network operating as services. The analyst can use the

a Developed by Lockheed Martin for the Department of Homeland Security, CEMSA is owned by DHS, with no Lockheed Martin copyright, and is free for DHS to install.

Figure 2. CEMSA architecture integrates models and data.



same view to monitor, pause, stop, or change plan execution. A report is generated when the consequence analysis simulation is completed.

Planning engine. The planning engine generates an analysis plan from a description of the business processes represented by the DHS functional area analysis, the DHS infrastructure data taxonomy, and catalog of models. This information is stored in the meta-plan, an XML representation of a directed acyclic graph of meta-models covering all required analysis activities.

The planning engine uses the meta-plan to generate a plan as a directed acyclic graph of meta-models that responds to the RFA. It generates a hierarchical task network, including a hierarchy of analysis activities, within which are active task networks that specify potential combinations of categories of models. The hierarchical task network approach is used in many industrial-strength planners to reduce the search space of potential plans to those corresponding to actual task activities in a given domain, in this case consequence analysis.[4]

The planning engine selects and composes resources that satisfy constraints (such as time, fidelity, and inclusion or exclusion of specific models, data, and experts used in generating a solution). Iteration through the CEMSA Radar user interface selects among alternative models to perform the actual model computations appropriate for that model category. The planning engine estimates the overall duration of the resulting plan, and if a time constraint cannot be met, alternate models are substituted into the plan to reduce the overall timeline. If the time constraint is still unmet, the planning engine uses model approximations. The user interface gives analysts the ability to modify the model composition and replace selected models with others as needed.

The planner translates the completed plan into an executor-appropriate format. For the Kepler executor, this is a workflow of "actors" representing Web Service Description Language (WSDL) calls and edges representing the data passing between models.

Approximation engine. The approximation engine enables timely consequence analysis through estimates of analysis plan outputs and of end-to-end run times as a function of desired granularity. It characterizes the uncertainty associated with the models and approximates individual models with a simpler surrogate model. The approximation engine works with the planning engine by applying statistical analysis to similar previous analysis plans, captured in a knowledgebase, to provide an approximate answer. This approach is similar to the one used in engineering optimization, where model approximations (surrogates) are used for repeated runs of the composite simulation.[1]

A disruption to supervisory control software might shut down an electrical generator, and the resulting loss of power might disable a water-pumping station.

CEMSA's simulation computes probability distributions for desired parameters describing complex models consisting of existing models. Parallel and distributed simulations, in which multiple simulations consisting of individual models or simulations execute simultaneously on different processors/platforms, provide the best scalability as requested analyses become more complex and more users interact with the system.

Semantic reasoning engine. A key challenge for model composition is understanding when models can be coupled together and what models can be coupled as part of a larger simulation; this requires capturing and reasoning over model type information, as well as model input and output types. CEMSA's semantic reasoning engine addresses this semantic challenge, with the catalog component of the engine serving as a knowledgebase and reasoner storing previous studies, interdependencies, models, simulations, and datasets.

Ontologies in the catalog support model composition in the planner, categorizing models and their inputs. CEMSA's base ontologies are Semantic Annotations for WSDL Working Group and OWL-S ontology built on top of the Web Ontology Language (OWL) for services, Kepler for model composition, and an adaptation of the DHS infrastructure data taxonomy; the table here summarizes the standards governing CEMSA's catalog representation.

When the planning engine populates an abstract task in the hierarchical task network, it identifies context, including event type, CIKR sectors, and activity the analyst requested. The catalog iterates

through contexts to identify the models that are relevant to the event, sector, and activity, using an ontological reasoner to check for satisfaction over the class hierarchy in the ontologies.

System administrators integrating new models and simulations into the CEMSA framework modify the catalog through the knowledge-manager-perspective user interface that supports describing how a class of models fits into the analysis process; modelers provide details for model instances through a modeler-perspective user interface for describing the model itself. The planning engine in turn requests types of models matching constraints from the catalog, as do the approximator engine and the analyst assistant when presenting these constraints for selection by the analyst.

Model ingestion engine. The primary requirement for adding models is that they provide an external API from which developers can construct a Web service. CEMSA represents service interfaces through WSDL, invoking them through Simple Object Access Protocol (SOAP). Modelers construct their own Web service wrappers or use the facilities provided by the Kepler scientific computing framework, which also serves as the basis for CEMSA's execution engine.

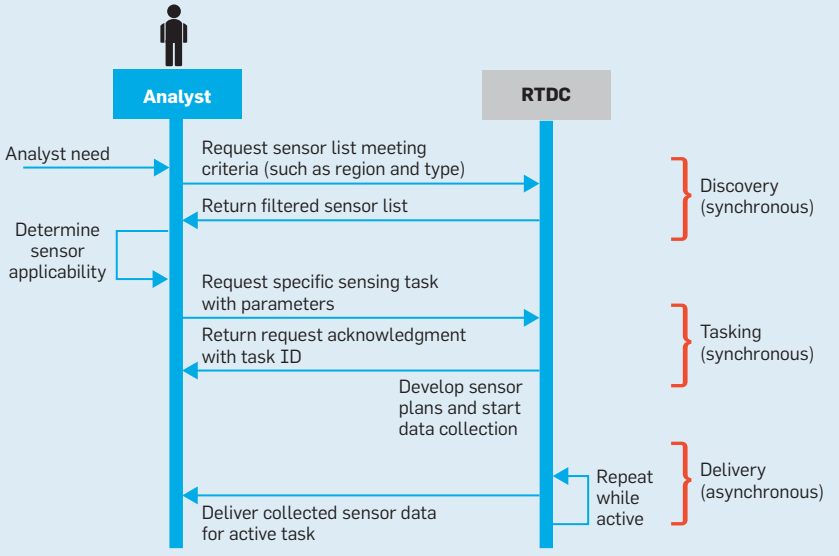
Once models are wrapped as a WSDL, the modeler can ingest them into the CEMSA catalog through the Radar user interface. Using the catalog, the model-ingest client prompts the modeler to provide details (such as execution time, units, and types used on inputs and outputs and valid domains for the model), as well as information on standards in the table. When the modeler supplies the metadata, the ingest client stores the Kepler wrapper and metadata in the catalog.

Knowledge manager assistant. A knowledge manager (KM) examines the CEMSA catalog of models, data services, and experts, as well as the end-user organization's analysis process, and provides the planning engine task hierarchies that fit the organization's business processes. The meta-plan states the sequencing of overall activities and potential couplings between categories of models. CEMSA provides a perspective for the KM to achieve these goals.

Modeling standards used by CEMSA.

Category	Standard	Description
Measures	Metric	Metric measures (meters, liters, grams) for model inputs and outputs and seconds for temporal measures
Network Protocol	SOAP 1.2 over HTTPS	Network transport uses SOAP xml over https for secure data transmission
Services	WSDL 1.1	Model provides callable input, output, and control interface through the WSDL 1.1 standard
Semantic Models	OWL-DL	Describes inputs and outputs for models that provide semantic results or use semantic inputs
Queries	SQL, SPARQL	SQL database queries from model to CEMSA database service; SPARQL queries from model to CEMSA ontology knowledgebase service
Information Exchange	NIEM	CEMSA emphasizes core, geospatial, CBRN, cybersecurity (emerging), emergency management, and infrastructure protection sectors of the National Information Exchange Model
Geospatial Data	OGC standards: KML, WFS 2.0, WMS 1.3, WCS 2.0, WPS 1.0	Support standards include KML for 3D overlays, annotations and interaction; Web Feature Service for map features; Web Mapping Service for images, Web Coverage Service for grid data of geospatial regions; and Web Processing Service for algorithms or processes operating on geospatial data, including process status
Geographical Coordinate System	WGS 84	Uniform coordinate system reduces processing time and potential translation errors
Control	CEMSA	Service interfaces to models include initialization, time advance, play, pause, resume, and stop
Test Cases	CEMSA	Provide example inputs, outputs, ranges, and datasets for testing and characterizing model operation

Figure 3. Real-time data can be combined with simulations.



This perspective can display all model information in the catalog. The KM examines the model metadata, aligns the model with the appropriate analysis phase or activity, and edits the meta-plan to assign possible model input/output connections. The KM can commit the meta-plan for future use from this perspective. The perspective also provides access to previous model results and performance data as another resource for updating the meta-plan.

Real-time field data engine. The real-time field data engine enables analysts

to use near-real-time information from third-party sensors that feed information into the CEMSA system; the real-time sensor information is a way to inform model compositions in CEMSA.

The real-time data collection (RTDC) module extends the CEMSA architecture with an integrated data-collection mechanism for real-time access to heterogeneous sensing and live data input mechanisms. The functionality of the RTDC is divided into three distinct steps (see Figure 3): The first is "sensor discovery," or the process of identifying relevant sensors to probe for the event at hand. The second is "sensor tasking," where sensors are activated with data-collection plans to match resource needs. Sensor-tasking requests come in a variety of types: periodic, where sensor data is scheduled to arrive periodically; event-based, where sensor data arrives upon occurrence of an event (as determined by an event condition); instantaneous one-off, where the `model_to_sensor_correlator` service generates a one-time sensor data demand that could arrive at any moment. Any of these requests may be deadline-based where the result of the sensing is required within a deadline based on when the request is made. The final step is sensor data delivery.

The RTDC design is based on the SATware middleware[3] for sensor-based systems that support high-level abstraction of sensors and sensor resources to abstract the heterogeneity and diversity of sensors and sensor platforms. Such heterogeneities make programming pervasive applications highly complex, especially when applications must explicitly deal with failures, disruptions, timeliness properties under diverse networking and system conditions, and missing or partial information. In RTDC, applications deal with higher-level semantic concepts (such as temperature, traffic level, and presence of entities/activities) at a given location and time. Such a semantic abstraction is provided through "virtual sensors" bridging application-level concepts and raw sensor data using "operators" that transform input sensor data streams (such as weather sensors) to higher-level semantic streams that capture application-level concepts and entities (such as region affected by hurricane). A phenomenon may be observed through multiple sensors by defining appropriate virtual sensors and/or combining inputs from multiple virtual sensors through the sensor composition language of SATware. Virtual sensors, when defined and registered with RTDC, hide the complexity of sensor programming from applications.

To address the challenge of real-time acquisition and processing of sensor data, RTDC models the sensor-data-processing problem as a constrained optimization problem.[7] Applications have associated with them a benefit function that models the utility of observing a phenomenon to the application. With CEMSA, that benefit may correspond to reduced uncertainty in the phenomenon of interest. The sensor-acquisition task is subject to constraints (such as artifacts of the sensor properties, as when a given camera can be focused on only one region at a given time), the acquisition process (such as a website where data is acquired and might impose restrictions on the number of queries an application can pose per unit time), and resource constraints (such as network bandwidth or limitations of available computational resources). Moreover, sensors involve associated actuation parameters (such as spatial/temporal resolution of data capture and other quality measures like errors/deviation from the actual value). RTDC chooses actuation parameters that maximize the expected benefits while ensuring the constraints imposed by the sensors, resources, and collection process are satisfied.[7]

Figure 4. Functional area analysis defines plans, including cyber effects.

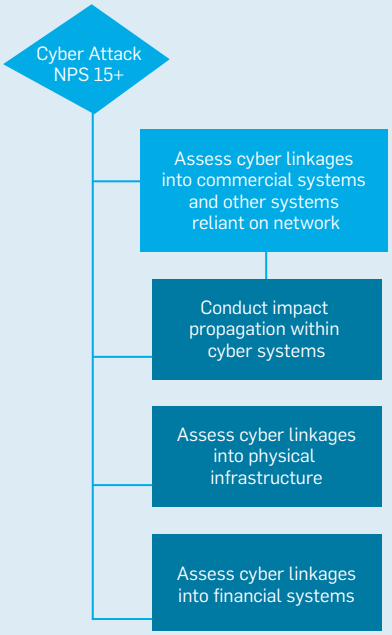
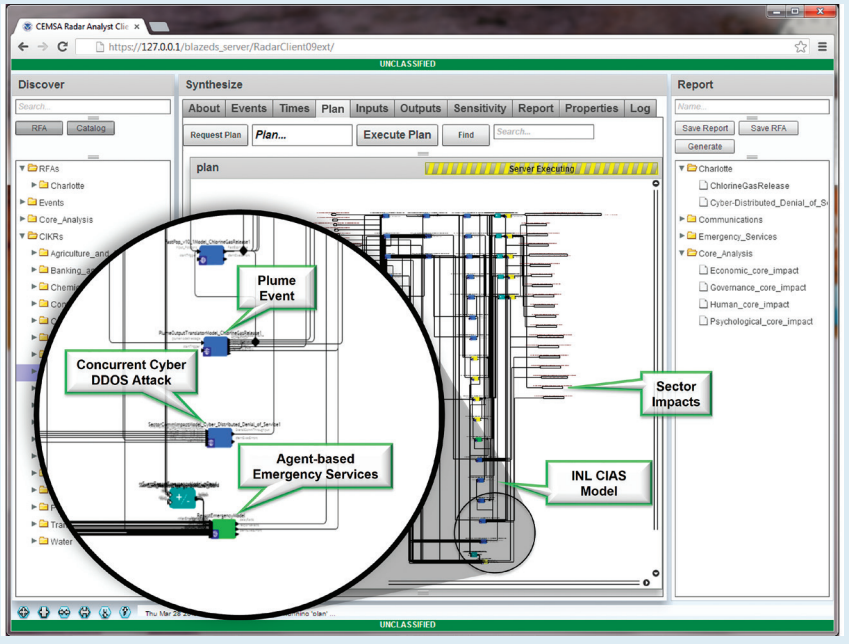


Figure 5. Generated plans model consequences of cyber events on the physical infrastructure.



Explanation engine. The explanation engine helps analysts understand the provenance and applicability of results by associating data and model results, collecting simulation data at the time of analysis, and presenting results. It supports real-time decision making by tracing where new data caused an analysis workflow to diverge from expected or previous results. The explanation engine stores data from throughout the composition and execution life cycle, including starting data, intermediate values, and final values. Analysts search across intermediate and final results for each run and use persistent data from prior runs to compare results.

The explanation engine lets analysts visually trace the running execution and the simulation results to an individual model and its associated inputs and outputs through an editable data-flow diagram in the plan view, providing error bounds for final results as a function of error bounds of each individual model. Sensitivity analysis is a key method for explaining data and model results. Starting with a plan from the planner, the explanation engine uses Monte Carlo methods and model approximations to provide input to a clone of a given plan, simulate its run, and calculate the distribution across each numeric output value.

Assessing Consequences of Cyber on the Physical

Now consider the following scenario. Preparations for a major public event held in a major metropolitan area (such as a national political party's convention) include analysis of the consequences of an attack on infrastructure. Analysts would use CEMSA to generate an analysis plan in response to an RFA. CEMSA generates the plan from a description of the underlying process represented in the DHS functional area analysis (see Figure 4).

One scenario analysts plan for is a distributed denial of service (DDoS) attack timed with highway explosions releasing toxic chlorine gas upwind of the political event, a scenario that represents a "complex event" with concurrent and cascading effects. The DDoS attack is under way while the explosion and release of the gas occurs (concurrent disruptions), with each of these initial events producing cascading effects on multiple infrastructure sectors.

Effects on communications network QoS degrade the capacity and reliability of communications between citizens and responders, causing dispatch delays and errors that lengthen response times and increase casualty rates.

The DDoS attack impairs communications by citizens, emergency responders, and industrial-control systems required for the safe operation of electrical-, water-, petroleum-distribution, and other networks. The explosion and gas release disrupt the regional transportation grid, and the ensuing plume requires citizens to shelter in place while causing illnesses that require health-care and emergency medical services. Both events interact with the load already imposed on local and regional assets by the political event in the form of concentrated demand for communications, information technology, and transportation resources, as well as additional responsibilities for police and emergency managers. The consequences of these interacting events ripple across the region.

An analyst selects a subset of interesting features from these possible interactions and affected CIKR sectors; for example, the CEMSA team has evaluated the performance of emergency responders—their ability to promptly answer 911 calls triggered by the interacting events, in addition to the base load of police, fire, and medical calls—as affected by changes in multiple network sectors (such as communication, water, and electrical power). In this case, the planning engine generates an analysis plan that, in addition to physical models, includes multiple cyber impact models spanning multiple infrastructure sectors and using different simulation techniques and models of computation (see Figure 5).

Infrastructure networks model. The Critical Infrastructure Analysis and Simulation (CIAS)[b] model performs flow-graph analysis and simulation of networked infrastructure, including water (such as pumping stations, pipelines, and consumers), power (such as generators, transmission, and distribution lines and loads), and communications (such as network exchanges, backhaul, and access points). Users specify interdependencies among infrastructure assets by setting attributes for individual as-

b Developed by the Idaho National Laboratory, Idaho Falls, ID.

sets (such as minimum demand and maximum capacity for commodities like water and electricity).

CIAS performs flow and reachability analysis to propagate effects among infrastructure networks stored in a geospatial information system's database. Reachability analysis determines the existence of source-to-sink paths and essential links for each infrastructure sector. Flow analysis computes the flow of resources (such as water volume and data throughput) available at each point, recording shutdowns and impairments when flow levels are below minimum requirements for assets.

Interdependencies among the supported sectors propagate failure modes across networks; for example, a disruption to supervisory control software might shut down an electrical generator, and the resulting loss of power might disable a water-pumping station. A nearby communications exchange might have backup power, but loss of cooling water will cause it to shut down. The resulting loss of network connectivity might require a nearby water-treatment plant to shut down. Effects propagate to other sectors that depend on networked commodities (such as emergency managers who require network connectivity to dispatch 911 calls and health-care providers who require water and electricity to run an emergency room).

Emergency services model (ESM). Inspired by Mysore et al.,[6] the ESM[c] is an "agent-based" simulation that computes "network infrastructure" effects (particularly impairment to transportation and communications networks) on the capacity and efficacy of police, fire, and emergency medical services. It represents individual health-care and emergency-services assets and citizens as agents, or computational entities responding to the simulated environment and other agents by following simplified rules that reflect essential behaviors.

The ESM is implemented in the Repast Simphony suite[d] and loads transportation network information from the OpenStreetMaps dataset, locations of emergency services, and health-care

The scenario envisions a DDoS attack on the carrier's 4G network, in which the communications model used by DEMSA simulates the cell towers closest to the political event and deploys multiple users in each cell to measure network impairment caused by the attack.

and communications assets from the Homeland Security Infrastructure Program (HSIP) Gold[e] 2012 dataset. Like CIAS, the ESM extracts "network" relationships from geospatial data. The datasets organize data into "layers" according to asset type (such as roads, law-enforcement facilities, and cell towers), describing each instance of an asset by a "shape"; for example, the road network consists of a set of road segments (represented in the dataset by polylines) that meet at intersections specified by (latitude, longitude) points. The ESM converts this spatial representation into an undirected graph with vertices corresponding to intersections and edges corresponding to road segments, using the graph representation to dispatch emergency responders along the shortest available path to the location of a casualty. The ESM uses heuristics to reconstruct connected networks from the sometimes-disconnected segments in geospatial datasets; for example, it "enlarges" the road segments' endpoints to connect them to adjacent segments, as when two or more multi-lane roads meet at an intersection that could be tens of meters wide.

The HSIP Gold dataset provides limited information about core network assets and backhaul links maintained by commercial voice and data carriers, so the ESM uses a spanning-tree approximation to connect "edge" assets like cell towers to core assets like switching centers. Each mobile responder is affiliated with a fixed site—police cars with police stations, fire engines with fire stations, and ambulances with hospitals.

Each asset type follows a simple state machine: citizens can be `healthy`, `ill`, `injured`, or `dead`; fixed sites can be `available`, `limited`, or `unavailable`; and mobile responders can be `available`, `dispatched`, `on_scene`, `limited`, or `unavailable`. Each asset type can

c Developed by the CEMSA team.

d Developed by the University of Chicago and Argonne National Laboratory.

e DHS HSIP Gold is a unified homeland infrastructure foundational geospatial data inventory assembled by the National Geospatial Intelligence Agency in partnership with the Department of Defense, DHS, and the U.S. Geological Survey for use by the homeland security/homeland defense community; it includes a compilation of best available federal government and commercial proprietary datasets.

enter the `limited` or `unavailable` state to model the effects of resource rationing and starvation as computed by CIAS, with communications between assets suffering quality-of-service (QoS) impairment computed by CIAS and the OpNet communications sector model (http://www.opnet.com/).

The ESM provides concrete examples of the consequences of impairment to networked infrastructure caused by cyber events. Effects on communications network QoS degrade the capacity and reliability of communications between citizens and responders, causing dispatch delays and errors that lengthen response times and increase casualty rates.

OpNet long-term evolution model. The OpNet modeler is a commercial modeling and simulation tool for voice and data networks. It is a discrete-event simulation in which "models" corresponding to software applications, networking protocols, and hardware devices interact by exchanging messages corresponding to networked data. OpNet and its partners have implemented several model libraries, including a wireless library with models of generic wireless devices (such as 802.11 Wi-Fi routers and terminals) and a long-term evolution (LTE) library that models the Third Generation Partnership Project (3GPP, http://www.3gpp.org/) LTE air interface. Subject-matter experts can use OpNet to simulate a range of use cases, including civilian, public-service, and military wireline and wireless networks carrying voice and data traffic.

Our scenario involving release of chlorine gas on a road, focuses on a region with a dominant commercial carrier that provides 3G (using code division multiple access, or CDMA, waveforms) and 4G (using the 3GPP LTE waveform) voice and data services to civilian and public-service users. It focuses on the 4G LTE network, which offers the highest end-user data rates and therefore drives the architecture of the carrier core network. LTE is an all-packet-switched, all-IP network that treats voice messages as a particular class of packet data. This converged approach, along with improvements in modulation and multiple access protocols, significantly increases network capacity. It also involves the possibility that abuse of the network by one class of application could impair the performance of others by exhausting shared resources. The scenario envisions such an event, in the form of a DDoS attack on the carrier's 4G network, in which the communications model used by CEMSA simulates the cell towers closest to the political event and deploys multiple users in each cell to measure network impairment caused by the attack. Each simulated user runs three simulated applications: voice (periodically initiating calls of varying duration), text messaging, and background IP data.

Related Work

The High Level Architecture (HLA) is a specification developed by the U.S. Department of Defense Modeling and Simulation Office with several implementations, including the MaK RunTime Infrastructure. It emphasizes information exchange between simulations over an information bus at the syntactic level using the Federation Object Model. Model composition in HLA is highly dependent on the selected run-time infrastructure (RTI). Composing models based on RTIs usually means implementing a custom gateway between RTIs. HLA results in a static architecture for model federation that is difficult to change dynamically, as in the CEMSA planner.

Conclusion

We have described the CEMSA system and the algorithms being developed to initial operating capability. The CEMSA approach is based on semantic model composition via hierarchical task network planning. Applying constructs from the planning community and the engineering-optimization community to infrastructure impact analysis, CEMSA gives analysts the means to assemble, coordinate, and evaluate collections of models for complex events and quickly arrive at effective decisions.

Acknowledgment

This work was funded by the Infrastructure Protection and Disaster Management Division, Science and Technology Directorate, Department of Homeland Security, Washington, D.C.

References

1. Eldred, M.S., Adams, B.M. et al. *DAKOTA: A Multilevel Parallel Object-Oriented Framework for Design Optimization, Parameter Estimation, Uncertainty Quantification, and Sensitivity Analysis: Version 5 Reference Manual.* Sandia Technical Report SAND2010-202184 (Version 5.2), Nov. 2011; http://dakota.sandia.gov/docs/dakota/5.0/Users-5.0.pdf
2. Homeland Security Infrastructure Program Gold Dataset; http://www.dhs.gov/infrastructure-information-partnerships
3. Hore, B., Jafarpour, H., Jain, R., Ji, S., Massaguer, D., Mehrotra, S., Venkatasubramanian, N., and Westerman, U. Design and implementation of a middleware for sentient spaces. In *Proceedings of the IEEE International Conference on Intelligence and Security Informatics* (New Brunswick, NJ, May 23–24). IEEE, 2007, 137–144.
4. Kambhampati, S. A comparative analysis of partial order planning and task reduction planning. *SIGART Bulletin 6*, 1 (Jan. 1995), 16–25.
5. Ludascher, B. et al. Scientific workflow management and the Kepler system: Research articles. *Concurrent Computing: Practice and Experience 18*, 10 (Aug. 25, 2006), 1039–1065.
6. Mysore, V., Narzisi, G., Nelson L., Rekow, D., Triola, M., Shapiro, A., Coleman, C., Gill, O., Daruwala, R.-S., and Mishra, B. Agent modeling of a sarin attack in Manhattan. In *Proceedings of the First ACM International Workshop on Agent Technology for Disaster Management* (Hokkaido, Japan, May 8–12). ACM Press, New York, 2006, 108–115.
7. Vaisenberg, R. *Towards Adaptation in Sentient Spaces.* Ph.D. Dissertation, University of California, Irvine, 2012; http://www.ics.uci.edu/~ronen/resources/thesis.pdf

Nabil Adam (adam@adam.rutgers.edu) is a Distinguished Professor of computer and information systems and founding director of the Center for Information Management, Integration and Connectivity at Rutgers University; he was the CEMSA program manager while serving as a fellow and senior program manager in the Science and Technology Directorate of the U.S. Department of Homeland Security, Washington, D.C.

Randy Stiles (randy.stiles@lmco.com) is a senior staff research scientist at the LM Advanced Technology Center in Palo Alto, CA, and LM program manager for CEMSA.

Andrew Zimdars (andrew.zimdars@lmco.com) is a staff research scientist at the LM Advanced Technology Center, Palo Alto, CA, and led the CEMSA Real-time Analysis Communication Environment project.

Ryan Timmons (ryan.p.timmons@lmco.com) is a senior research scientist at the Lockheed Martin Advanced Technology Center, Palo Alto, CA.

Jackie (Man-Kit) Leung (jackie.leung@lmco.com) is a research scientist at the Lockheed Martin Space System Company, Palo Alto, CA.

Greg Stachnick (gstachni@comcast.net) is a principal software engineer at LM Integrated Systems & Global Solutions Advanced Technology Office, San Jose, CA, where he led development of the CEMSA planning service.

Jeff Merrick (jeff.r.merrick@gmail.com) was a firmware and software engineer for the CEMSA planner at Lockheed Martin Integrated Systems & Global Solutions, San Jose, CA.

Robert Coop (Robert.Coop@rtsync.com) is a software developer at the RTSync Corporation, Knoxville, TN.

Vadim Slavin (vadim.a.slavin@lmco.com) is a software engineer, research scientist, and entrepreneur in Palo Alto, CA.

Tanya Kruglikov (tanya.s.kruglikov@lmco.com) is a senior software engineer at IBM Research - Almaden, San Jose, CA.

John Galmiche (john.galmiche@gmail.com) is an associate at Booz Allen Hamilton, Arlington, VA.

Sharad Mehrotra (sharad@ics.uci.edu) is a professor in the Department of Computer Science and founding director of the Center for Emergency Response Technologies at the University of California, Irvine.