Visualization and Study Mode Architectures for Real-Time Power System Control

Santiago Grijalva, Member, IEEE
PowerWorld Corporation, IL, USA

Abstract—This paper describes the architecture of Retriever, an innovative system for power system control centers, which provides advanced visualization and study mode features. Visual monitoring has many advantages over traditional non-graphic supervision, allowing the operator to process large amounts of information and capture the aspects relevant to prevent insecure power system states in a very efficient manner. Real-time power flow and study mode simulations allow the discovery of potential problems in the system and points out towards prompt control actions by the operator. These visualization and analysis tools have been integrated to provide a powerful architecture for real-time power system monitoring, and are operative in large control centers with tens of thousands of nodes.

Keywords—Visualization, real-time control, power flow.

I. INTRODUCTION

Modern power system control center operations face numerous challenges originated in the forces of deregulation, technology development, and operation complexity. This has resulted in more stressed operation close to security limits in power systems around the World. The recent blackout events have increased the need for more portable and easy to use tools that can predict dangerous conditions and alert the operator to take corrective actions [1,2]. It also points out the need of better integration between real-time operations and system planning.

Historically, real-time operations and planning have been separated both from the enterprise process point of view and from the technologies therein utilized. There were few processes “in the middle”. This separation created numerous technological difficulties to power system and computer engineers. The real-time features needed by the data acquisition and processing functions of EMS systems forced EMS vendors to use real-time operating systems. The planning environment on the other hand adopted the less expensive personal computer platform. The proliferation of PC’s resulted in fast evolution of new algorithms, visualization and graphical user interfaces. Ironically, the more expensive real-time environment did not evolve as fast as the personal computer industry. As a consequence of this, some analysis and visualization features available in the planning environment have not been implemented in real-time systems. In most operative EMS systems, the core algorithms have not migrated to object-oriented programming technologies and maintain their Fortran and C implementations, considerable limiting technological development and application integration.

At the core of this process separation lay the power system EMS and planning models that were adopted by the industry. While real-time EMS applications model the complete topology of the power system at the breaker/node level, the planning model is a less detailed bus/branch model that was more portable and easy to use in personal computers. Core applications, such as the power flow, which in essence are meant to model the same system and determine the same quantities, require different implementations in the planning and in the real-time environment because of this model differences.

In addition, since most of the technology utilized in real-time systems was proprietary, for the longest time there was no mechanism to compare models that were meant to represent the same physical system and to determine their consistency. Even with great advances in networking, open operating systems, databases, and the internet, there are few control centers that work with fully compatible planning and real-time cases. This was pointed out as a strong limitation for post-mortem analysis developed by the task force that investigated the New York 2003 blackout [1].

Deregulation and electricity markets have created a tremendous need to overcome this planning and EMS separation and have created great integration possibilities. There are many examples of processes where EMS and planning compatibility would be of great service [3]:

- Day-ahead markets, where engineers and marketers require real-time information and fast flexible tools to analyze system conditions and economic opportunities.
- Short-term bilateral transaction analysis
- Reactive power planning
- Ancillary services operation
- Market settlement
- Post-mortem analysis
- Same day security analysis
- Strategic simulation by market agents
- Market monitoring

Seamless exchange of models between these two environment would result in significant benefits, including improved operating practices, increase in the ability to predict insecure conditions in the system, better tools to identify economic opportunities in the electricity markets as well as to monitor market efficiency, more secure operation of system through real-time visualization, unification of control center technologies, and savings in time and cost of personnel training.

We start this paper by presenting the state-of-the-art visualization architectures in Section II, and visualization strategies in Section III. In section IV we introduce the architecture needed for real-time power flow analysis, topology processing, and model compatibility. In section V we present how real-time power flow is integrated.

II. VISUALIZATION ARCHITECTURE

A. Aliases and Incomplete Power Flow Models

A basic design goal of power system visualization application is to make them compatible with different EMS data platforms ranging from flat text files and relational databases to historical databases. The application is designed to retrieve a snapshot of the
system, i.e., a complete description of the power system at a certain point in time, usually equivalent to the state estimator solution or a complete set of the process measurements, indications, and events. This goal is achieved with a generic object capable of interacting concurrently with different data source types. To the operator it is irrelevant whether the data comes from one or more data sources of different types. The data sets are unified through the diagrams managed by the application, facilitating in this manner discovery of relevant information.

A second goal of the visualization application is displaying information of particular types without the need to model the entire real-time snapshot. For visualization purposes, the data sets of the system snapshot are typically grouped by:

a) Voltage level, for instance, only 230kV and above.
b) Geographic region
c) Object type: node, substations, zone, or control area.
d) Level of detail, for instance substation object or substation topology diagrams at the node/breaker level.
e) Thematic diagrams such as voltage contouring, marginal price profiles, reactive power reserves, etc.

Often, control center operators require a specific diagram that requires only a small subset of the entire real-time snapshot. In order to achieve this, the visualization application uses the concept of aliases. An alias is data point name that relates the value in the data source to a graphical object in the diagram. As an example, suppose that a certain 138kV bus named ADAMS 138KV NODE 1 in the system has a real-time voltage of 136.3KV. We can define an alias called ADAMS 138KV NODE 1.VOLTKV where the first part of the name is typically the EMS name and the suffix is the quantity type. The datasource contains a table of this alias together with the real-time value. It also contains a table of alias subscriptions. An alias subscription is a mapping from a graphical object field to the alias. For example, the following subscription:

3465, BusKVVolt, ADAMS 138KV NODE 1.VOLTKV

states that the KV voltage field of bus 3465 drawn in the oneline diagram will display the kV value associated to the alias ADAMS 138KV NODE 1 which corresponds to the actual bus in the system.

The use of aliases and alias subscriptions completely decouples the EMS data from the visualization objects. In this manner, diagrams with as few as key devices can be built regardless of the EMS data model. A second advantage is that by creating fictitious devices, the objects in the diagram do not have to correspond to actual devices. Suppose that only the flow through a set of phase shifters need to be displayed in a diagram. Then phase shifters graphical objects can be drawn on the diagram with arbitrary from and to node numbers and names. The fields of the phase shifter device will subscribe to the corresponding phase shifter aliases to display the correct values of for instance MW flow and phase angle. If the voltages at the from and to ends of the phase shifters are to be displayed, the association between these buses graphical objects and the actual devices must be established.

The set of graphical objects whose fields would subscribe to aliases for real-time display are set in a power flow case called the Underlying Power Flow (UPF) model. The UPF is just a set of objects not necessarily related or connected among them, which form a base for alias subscriptions. When only few devices are modeled in the case, the UPF is said to be Incomplete.

One of the key challenges when using alias and alias subscriptions is the resolution of the alias name when reading real-time data. Special searching algorithms are utilized to make the search fast and comparable to a sorted list search. Common speeds of data reading are about ten thousand data points (measurements or indications) in 0.2 seconds.

![Diagram](image.png)

Fig. 1 Alias Subscription Architecture for Incomplete Models

### B. Full-Topology Models

We have seen that if only a subset of the snapshot is to be displayed, then it is sufficient to use aliases and an incomplete UPF. On the other hand, often times control centers require visualizing the entire system snapshot. Consider a substation diagram of an entire system. In a substation diagram, substations are represented by little rectangular objects from which transmission lines to other stations emanate. However, some information about the substation is often displayed as the result of calculations based on the values of the substation devices:

a) Maximum KV voltage at the highest voltage level
b) Minimum KV voltage at the lowest voltage level
c) Total generation MW, Mvar and reserves
d) Total shunt Mvar injection
e) Total MW and Mvar load.

The determination of these values requires knowledge of all the measurements of devices in the station, with the exception of transformer, zero impedance, and breaker flows. Clearly, retrieval of a significant portion of the snapshot data would be required for a system substation diagram.

It is also common to link the substation graphical objects in the substation diagram to detailed substation topology diagrams, in which all the nodes, breakers and individual station devices are displayed. This requires that all individual components of the station are modeled, and their fields (measurements and indications) retrieved. When this level of detail is required by the visualization application, it is common to use a Full-Topology UPF, which is a power flow model as described by the EMS database. When the application uses a full-topology UPF, real-time data retrieval can take place through aliases, or it can occur directly if the modeling convention with respect to node names and numbers is consistent with the EMS database.

Although more involved and memory consuming, the use of the full-topology UPF has several advantages with respect to the incomplete UPF as based for system visualization:

a) Data retrieval per data point is faster since resolution of the alias name is no longer needed.
b) Detailed visualization of every device, measurement and indication is possible.
c) With complete and consistent real-time snapshot data, it is possible to obtain a real-time power flow solution.
III. REAL-TIME VISUALIZATION STRATEGIES

Real-time visualization is a core application in modern control centers. Research on human factors and man-machine interaction has shown that visual features outperform other types of monitoring and facilitate information discovery and prompt reaction by the operators [4-6]. In this section we describe some of the application main visualization features.

A. Geographic One-line Diagrams

One of the major advances in large scale power system monitoring is the possibility of relating the one-line diagram of the system to an actual geo-referenced layout of the grid (compatible with GIS). Fig. 2 shows a large scale visualization of the impact of a transfer from Wisconsin to Tennessee using as background the one-line diagram of the United States Eastern Interconnection. The color intensity corresponds to the percentage of the flow, as expressed by the power transfer distribution factor (PTDF) that would flow through the transmission elements. While this particular diagram contains thousands of transmission lines, the visualization allows the operator to see almost immediately the effect of the transfer and the distribution of flows, and determine potential loop flow effects, overloaded elements, system constraints, and control areas involved. Zooming, panning and a variety of features make this type of interface significantly more effective than a mimic diagram or a tabulated list.

Fig. 2 Geographic One-line Diagram

B. Automatically Generated Diagrams

The visualization application provides two types of automatically generated displays: Bus View and Substation View diagrams. These diagrams provide a local-area view of small portion of the entire system. A number of options are available to automatically set the fields of each graphical object to be displayed in the diagram. As the operator clicks on new buses or stations, new diagrams are generated allowing her to navigate through the particular region of the system.

From the bus view, the operator can find out a bus’s voltage and angle, marginal price, load, shunt compensation, and generation connected to the bus, and the flows on all lines emanating from the bus, provided this information has been retrieved from the registered data sources. Each object that appears on a bus view has an associated local menu, which provides a number of options, including the ability to open an information dialog for the object that was right-clicked. Back and Ahead buttons allow stepping through the history of buses viewed thus far. The diagram allows specifying a bus name or a bus number. As with a one-line diagram, flows on a bus view can be animated. Other options of the bus view include the number of tiers displayed, showing object hints, showing serial buses, showing equivalent lines, and opening multiple bus views. Fig. 3 shows a bus view diagram.

Fig. 3: Bus View Diagram

A similar type of diagrams is provided for substations. This Substation View diagram shows a substation, a summary of its elements and principal metrics, and its interconnections, but does not show its internal topology.

C. Flow Animation and Dynamic Pie Charts

The application allows you to animate system flows based on values read from the real-time data source. There are many animation options such as type of flow object (normally arrows), density, size, and speed relation with megawatt value. Megavars can also be animated together with the megawatt values.

Pie charts are used with lines and interfaces to indicate the size of the flow on the component relative to its rating. Pie charts are very effective in highlighting those lines and interfaces whose heavy flows merit immediate concern. This capability is heightened in the application because it can be configured to change the size and color of a pie chart to reflect different levels of alarm. For example, the operator can set to show lines loaded between 80% and 100% of their ratings using an orange pie chart and lines loaded beyond their ratings as red pie charts. In both cases the size of the pie chart would be increased. Fig. 4 shows an example that demonstrates how pie charts make it easy to detect the system’s most heavily loaded lines in a system with hundreds of lines.

Fig. 4: Animated Flows and Dynamic Pie Charts
D. Contouring Diagrams

Contouring is a useful visualization technique that provides a spatial view of the variation of a specific quantity. Contouring communicates a great deal of information to the user very quickly because it frees the user from having to scan the contents of a diagram looking for exceptional values. Instead, exceptional values, such as low or high voltages or extreme bus marginal prices, distinguish themselves through color. The application allows the user to contour any data field for buses, lines, interfaces, and areas. The most common types of contouring displays include bus per unit voltage, bus marginal prices, bus angle, line flow, line transfer distribution factors, area average marginal prices, and interface transfer distribution factors.

The contouring tool has been designed to allow the user to customize the appearance of the diagram to communicate system conditions as clearly as possible. Color maps, ranges, fading, influence, resolution, and several other settings can be customized. Fig. 5 shows a contour plot of bus voltages that shows high voltages in red and low voltages in blue. Together, the voltage contour and the line flow pie charts highlight the voltages and branches that merit the most concern from a monitoring standpoint.

E. Views, Layers, and Object Filtering

A View is a data structure that stores the x and y coordinates of the center of a display and the display’s zoom level. This allows the operator to quickly return to a particular view of the system. There can be as many views as desired.

Different objects can be assigned to different layers. The layers functionality enables the user to focus on subsets of data by hiding other subsets. For instance, objects of different nominal voltage can be assigned to different layers to obtain an overview of the high voltage grid or more detail of the lower voltage grid as needed. Another example is displaying or hiding economic information such as marginal prices.

Another alternative for hiding objects is to directly tell the application what type of objects to hide. The selection of objects to be hidden or shown is done using filters that can be combined such as zone, area, or owner; nominal voltage range; or device type. Furthermore, the filtering can be done at field level which depends on the device type. For instance one can hide all generator Mvar fields, but continue displaying the megawatt values on the diagram to de-clutter a specific view.

F. Visualization in 3D

Adding a third dimension to a oneline display with complex, interrelated data can often help clarify the relationships among system quantities. The plot in Fig. 6 contours power transfer distribution factors in the plane and emphasizes alarm-level line flows along the vertical axis by drawing line flow pie charts as cylinders whose heights represent the percentage flow of each line. Other common quantities that are usually represented in the vertical axis are generator MW and Mvar reserves, area interchange, and zonal LMPs.

G. Dynamic Formatting

Often, it is interesting to change the formatting of certain fields or objects depending on power system conditions. For instance, if the Mvar limit of a generator has been reached, we can change the value on the diagram to a red color, indicating that a limit has been hit. In addition, the generator graphical object itself can be modified regarding its size, color, etc. A dynamic format object specifies the type of object and object field that would change its format, how the format would be changed, and under what conditions the change will occur. There could be any number of dynamic formats, and the conditions can include advanced Boolean expressions. Furthermore, dynamic formatting can be used to show or hide the objects. For instance a transmission line pie chart showing the line loading can be displayed if the loading is over a warning level, e.g., 85%, or be hidden otherwise. This prevents the operator from seeing a large number of pie charts of lines that are within their loading limits and concentrate on those close to their limits.

H. Recording System Conditions as a Movie

Recording a movie of system conditions can be a useful way to capture and play back the evolution of a system over time. This technique is particularly effective for post-mortem analysis, as well as for training and monitoring. Among the options for saving a movie are the type (AVI or MPEG), the number of slides, the length of each frame, the display size, and the pixel depth, which controls the quality of the movie.

The slides in the movie can be generated by synchronizing them to the diagram rate of animation or to the rate of data retrieval: a slide is saved only after acquiring new data from the registered data sources.
IV. ARCHITECTURE FOR REAL-TIME POWER FLOW

A. EMS and Planning Models

There are important opportunities for technologies that would make power system analysis and visualization available to the control center operations planning and real-time control processes. Control centers are interested in flexible analysis and visualization tools that are not limited by the drawbacks described in Section I. The identification of the current data models utilized for real-time power system control allows integrating applications from the planning side into the real-time environment.

In the EMS model, system elements and devices are specified by names. Some EMS systems use a three-level naming convention to represent system elements: STATION, DEVICE ID, and then VOLTAGE LEVEL. The STATION is a unique identifier for the entire system. The VOLTAGE LEVEL corresponds to nominal voltages existing within each station. The DEVICE ID is a string that conveys both the type and reference to the particular device. For instance a name ADAMS 13.8KV GEN1 denotes a unique generator in the system, which is connected to a 13.8KV node in the Adams station. ADAMS 230KV BKR8 denotes a unique circuit breaker in the same station. For transmission lines, the convention utilized is FROM_STATION, then VOLTAGE and then TO_STATION. For instance, ADAMS 230KV ALDENE represents a 230KV line connecting stations Adams to Aldene. This naming convention allows operators to easily recognize different elements, in particular during alarm and event processing.

The planning case, on the other hand, consists of buses and branches. All devices are attached to buses and buses are identified by numbers. All the devices: lines, transformers, generators, shunts, etc are thus identified based on the number of the bus or buses they are connected to.

The power system EMS model is stored as a full-topology UPF model. This allows using a topology processing algorithm to dynamically generate a smaller planning representation of the system, keeping the references that allow mapping quantities between these two models.

Fig. 7 shows a substation arrangement. Transmission lines are connected to the bus sections through circuit breakers. The bypass breakers at the ends of the transmission lines are used to carry the power while the other circuit breaker is opened for maintenance. For simplicity, the nodes are numbered.

It is assumed that all breakers have zero impedance. In the station configuration shown in Fig. 7, all the nodes connected by closed breakers correspond to the same electric point. This group of nodes forms an Electric Area. Each electric area identified in the EMS model corresponds to a bus in the planning model. If the breakers from nodes 2 to 6, 2 to 7, and 3 to 7 were open, there would be two electric areas in this station and correspondingly two buses in the planning model.

A consequence of this topological relation is that while it is possible to generate a planning model from the EMS model and real-time breakers statuses, the opposite is not feasible. The planning case does not contain information about the topology and status of the breakers that originated each bus. Furthermore, when a planning case is created from the EMS model, it is customary to number the buses sequentially. Because the number of electric areas changes dynamically as the breaker statuses change in real-time, the numbering of buses does not remain the same from a data retrieval to the next; they change all the time. While planning cases keep the same bus numbers, any bus-branch model generated from the EMS will have a different number scheme at different snapshots. This is one of the reasons why it is difficult to compare EMS-generated cases with planning cases.

B. Topology Processing

Topology processing is the algorithm utilized to generate the bus-branch case from the EMS model. It identifies the nodes that are connected through breakers and thus correspond to the same electric point. It merges these nodes and all the devices connected to them in a single node that becomes a bus [7-9].

There are two types of topology processing routines:

a) Full Topology Processing, in which the bus-branch model is created from scratch
b) Incremental Topology Processing, in which the breaker statuses that changed since the last data retrieval are identified and the electric areas and buses are split or merged correspondingly.

In full topology processing, the bus-branch model is recreated every time. The algorithm starts with a full-topology model. The identification of the electric areas of the system is implemented with a recursive function summarized as follows:

```plaintext
routine FullTopologyProcessing
set all nodes as not visited
for each node do
  if node was not visited then
    create a new ElectricArea with this node inside
    add ElectricArea to the list of electric areas
    SetElectricArea(node,ElectricArea)
  end;
end;

function SetElectricArea(node,ElectricArea)
set node as visited
while there is a connection from the node do
  if the connection is a closed breaker then
    other_node = node to which the connection connects to
    if the other_node was not visited then
      add the other_node to the list of nodes in ElectricArea
      SetElectricArea(other_node,ElectricArea)
    end
  end
end
```

![Figure 7: Group of EMS nodes that form a single Electric Area. The Electric Area becomes a bus in the planning model. The electric areas formed depend on the status of the breakers.](image)
This recursive function is fast and can identify all the areas in models with tenths of thousands of nodes in less than a second. At the end of the process there is a list of electric areas, each one of them with a list of nodes and breakers.

The second step in full topology processing is to merge the nodes that are inside each electric area. The following function summarizes how this is done:

```java
routine NodeMerging
for each ElectricArea do
    move all the generators, loads and shunts to the first node
    move all the node pointers to the first node
    remove the breakers inside the ElectricArea
    update the branch connection nodes
    update pointers of remotely regulated buses
end
```

At the end of this process, all except one node from each electric area has been removed. This node has attached to it all the devices that were previously attached to other nodes in the electric area. All external transmission lines, series devices, phase shifters, etc that had as terminal a node inside the electric area have been moved to the first node. All other pointers of remotely regulated buses, etc are updated to the first node. The remaining nodes become buses in the bus-branch model, and thus there are no nodes, circuit breakers, isolators, or bus sections in the model anymore. Typically the bus-branch model has about 10-15% of the number of nodes present in the EMS model. If the EMS system models isolators in addition to circuit breakers, the previous percentage drops even further to 4-5%. Creating the bus/branch model is needed for the following reasons:

- **Electrically**, circuit breakers correspond to zero impedance branches that would contribute to zeros in the diagonal of the bus impedance matrix and the Jacobian. This would prevent these matrices from being invertible. Therefore, the power flow algorithm can only take place on a bus-branch model.
- **Even if circuit breakers are replaced by low impedance branches**, the large amount of these elements in the system (about 85-90%) may create numerical problems during the factorization of the Jacobian and possibly divergence of the solution [10].
- **The gain matrix of the state estimator will also have zeros in the diagonal making it not invertible** [11]. Thus both the power flow and the state estimation algorithm require a bus-branch model.

### C. Incremental Topology Processing

This faster algorithm takes care of updating an already existing bus-branch model based on the updated breaker statuses. The circuit breaker objects store their last status, making it easy to determine status changes since the last data retrieval. From the full topology processing run that generated the case, breakers also know in which electric area they are. When there has been a change in the status of a single circuit breaker the following can occur:

**Breaker Closing:** If the two breaker nodes belong to the same electric area, no change occurs. If the two nodes belong to different electric areas, the two electric areas become a single one: closing the breaker results in merging two buses.

**Breaker Opening:** With the breaker open try to find a closed breaker path from one terminal of the breaker to the other terminal.

- **a)** If a path exists, then opening the breaker does not split the electric area. No change in the electric areas occurs.
- **b)** If the path does not exist, then opening the breaker splits the electric area in two. Find the terminal that is connected to the first node of the existing electric area. Select the other terminal as the first node of the new electric area. Determine the nodes that are now in the new electric area.

In most practical cases though there might be more than one breaker update from one data retrieval to the next (20-60 sec). Consider for instance a line bypass breaker being closed and the normal breaker being opened. This won’t change the electric area. In those cases, the analysis of what happens to the electric area using the previous logic becomes more involved.

A more efficient approach takes place as follows. Suppose hypothetically that all circuit breakers of the power system are closed. Then the electric areas would be as large as possible, but they will still be limited by the boundaries of the voltage level within a substation. Either a transformer or a transmission line will separate an electric area from the other. Let us call these special electric areas **breaker groups**. We note that running full topology processing assuming that all circuit breakers are closed would result in determining all the breaker groups in the system. We also note that these breaker groups are constant unless the EMS model is modified with new devices or changes in the physical topology of the system. These changes are not frequent and are implemented at most each week through an EMS database generation. Thus the breaker groups can be determined during database generation.

In the same manner as each breaker knows to which electric area it belongs, it knows to which breaker group it belongs. In order to update the bus-branch model due to changes in breaker statuses only those breaker groups with breakers that changed their statuses need to be re-processed. From one data retrieval to the next (30-60 seconds) few breakers would have changed status. Thus the number of breaker groups that need to be re-processed is just a tiny fraction of all the existing breaker groups in the system. Incremental topology processing can therefore be implemented by running full topology processing considering only those breaker groups. We recall that the first action in the **FullTopologyProcessing** routine described above was to set all the nodes as not visited. This step would be spiked in the incremental topology processing. Recalling that at the end of full topology processing all nodes are visited, incremental topology processing would be implemented as follows:

```java
routine IncrementalTopologyProcessing
for each breaker do
    if breaker status changed then
        set the breaker breaker group as changed
        set the nodes of the changed breaker group as not visited
    end
end
run FullTopologyProcessing // skip initializing nodes as not visited
```

Again, at the end of incremental topology processing, all the nodes are flagged as visited. The next incremental topology processing run will flag the nodes inside changed breaker groups as not visited and process their topology, etc.
Topology processing is complemented with estimation techniques to determine gross measurement errors. For instance, if a transmission line has zero MW flow, nonzero Mvar flow and both breakers are the line ends are open, then probably the Mvar measurement is erroneous and should be zero. However, if both MW and Mvar measurements are nonzero and one breaker is closed and other open, probably the open breaker indication is wrong. Gross Measurement Error Detection consists of a set of logical rules are applied to inconsistent quantities in the model before full or incremental topology processing.

V. REAL-TIME POWER FLOW INTEGRATION

A fundamental goal in real-time operations is making available to the operator a consistent model of the physical system he can interact with. This can be obtained by solving the power flow with retrieved real-time data. There are two modes of the real-time power flow solutions: Automatic real-time power flow and Study Mode power flow.

A. Automatic Real-Time Power Flow

In the automatic solution mode the power flow is solved on the bus/branch model after full or incremental topology processing. The solution is then mapped back to the full-topology model for visualization. All node voltages, generator outputs, line flows, etc are mapped back and displayed on the online diagram of the EMS model. Consider for instance the 52-node, EMS system shown in Fig. 8.

![Figure 8: 52-node EMS system and its Electric Areas](image)

The online diagram of this system includes the detail at the circuit breaker as in conventional EMS systems. After topology processing the electric areas are determined as shown in the Figure resulting in a bus-branch model. Each electric area becomes a bus in this model and the transmission lines are now connected between the buses. We note that the 52 node system has been converted to an 11-bus system. Three of these 11 buses are non-energized, and are not considered in the power flow solution.

In automatic mode the real-time power flow solution sequence is as follows:

1. **Retrieve real-time data**
2. **Populate** the full-topology model with real-time data
3. **Initialize** the full-topology model updating all quantities
4. **Run** gross measurement error detection
5. **Run** incremental topology processing
6. **Solve** the power flow
7. **Map** results back to full topology model
8. **Animate** the solution

When looking at the on-line diagram, the difference with regular visualization is that instead of presenting individual quantities on the display, the quantities are now interrelated by the power flow equations, i.e., a solved power flow (and possibly solutions of other applications). This is similar to the visualization of state estimation data being superior to visualization of SCADA data.

B. Study Mode Power Flow

Besides retrieving real-time data and solving the power flow, operators can simulate actions and look at how the system will respond. This is called the Study Mode environment. Typical actions that operators need to evaluate are: opening or closing transmission lines, re-dispatching generation, connecting voltage support devices, modifying generation voltage set-points, controlling phase shifters and DC lines, modifying area interchange, and implementing area to area transactions.

![Fig. 10: Topology Processing Main Dialog in Automatic Power Flow Mode.](image)

Fig. 10 shows the real-time power flow dialog and topology processing control. When the mode is automatic, the automatic power flow sequence takes places. Otherwise the user can operate in study mode using the corresponding controls.

When the operator works in study mode, he makes a copy of an instance of the current bus-branch model state and performs actions on this model. The application would detect the modifications the user wants to implement and update the model. If the operator study includes opening or closing circuit breakers, incremental topology processing is triggered. The actions are therefore implemented in the following sequence:
User initializes a new study system
User implements actions on the system
User solves power flow
Application detects operator actions and updates the model
Application triggers incremental topology processing if needed
Application obtains a power flow solution
Application maps solution to full-topology model for visualization
User analyzes the operation impact using advanced tools:
Automatic limit monitoring and warning processing
On-line diagram and contouring visualization
Case information displays filtering

Incremental topology processing every 20-30 seconds can be combined with full topology processing every 5-10 minutes, balancing in this manner the speed of updates of the planning case with its complete creation. In study mode, all the operator actions over the EMS model are reflected in the planning model, and the results of the planning model simulation are reflected in the EMS model for visualization.

Once a bus-branch power flow solution has been obtained, the user can take advantage of powerful system analysis and optimization tools beyond the power flow:

- Contingency analysis, including remedial actions
- Voltage stability analysis including PV and QV tools
- Sensitivity analysis and generation re-dispatch
- Available transfer capability
- Economic dispatch
- Optimal power flow (OPF) Security-constrained OPF
- Hourly simulation of future system conditions

Furthermore, since the planning case is fully compatible with the visualization strategies mentioned in Section III, the operator can take advantage of them to develop visual analysis either real-time, post-mortem, or for future system conditions. By integrating this with real-time advanced study mode applications, a comprehensive platform for efficient large control center operations has been consolidated.

The platform and applications described in this paper are operational in several large control centers in the US. Besides a geo-referenced main system display, most of the control centers include thematic diagrams such as voltage contouring, marginal prices, phase shifter control, security analysis data, station topology diagrams, substation mimic diagrams, and generator reserve diagrams. Often the main geo-referenced diagram is overlaid with GIS weather information.

VI. CONCLUSIONS

The visualization technologies described in this paper facilitate efficient data processing by the operator and enhance the overall quality of man-machine communication process for real-time power system control. Visualization techniques are significantly superior to other monitoring methods.

The architectures that combine EMS and planning models are now technologically available and should be implemented in existing control centers. Making these models compatible enables using advanced algorithms and software for power system analysis, optimization, and visualization which was not previously available for real-time control. They also help validate existing models in the planning and real-time environment.

The utilization of state-of-the-art computer simulation technologies is the most promising step in avoiding system events and paving the road for secure system operation under deregulation.