# APPLYING ENTERPRISE APPLICATION INTEGRATION TOOLS – STRATEGY AND TACTICS

*Craig J. Andrews, P.E., Global Enterprise Managers, Inc.*
*Aretha A. Calloway, Potomac Electric Power Company*
*Sean T. Kelly, Potomac Electric Power Company*

## Background

The Potomac Electric Power Company (Pepco) delivers electricity to 1.9 million people in the District of Columbia and Maryland. One of Pepco's strategic initiatives for information technology is to buy best-of-breed information systems – for distribution operations, asset management, geospatial modeling and distribution design, supply chain management, human resources, and other critical business areas – and integrate the systems to form a seamless information environment. Between 2000 and 2002, Pepco will have replaced its Outage Management System, Geospatial Information System, Supply Chain Management System, and Human Resources System, commissioned a new Mobile Data System for field force management, and made major enhancements or additions to its Customer Information System and Work Management Systems. A key enabler for Pepco's best-of-breed strategy is the use of Enterprise Application Integration tools.

## Enterprise Application Integration

Enterprise Application Integration (EAI) has become a well-defined software product space over the last several years. EAI is now generally considered to comprise the following four capabilities:

- Messaging – Reliable and consistent approach to communicating between diverse software on different computers
- Transformation – Use of a centrally managed set of rules or similar techniques for reconciling differences in data definitions used within and outside the company
- Adapters – Integration of software applications into the corporate environment through the use of pre-existing or customized "adapters" or connectors that match the application to the EAI infrastructure.
- Workflow Management – Computer-based tools for representing, analyzing, organizing, and executing end-to-end business processes.

Messaging is the key to decoupling interactions between applications and moving away from the point-to-point interfaces that have traditionally made enterprise-scale integration nearly unmanageable. The concept of publishing information from one application through the messaging system, from which other applications can subscribe, is now well accepted. Reliable data transport and flexible routing throughout the enterprise are essential characteristics of

publish/subscribe messaging. Transport approaches vary and may include multicasting for the highest performance, real-time applications, but in all cases the messaging system will require inherent queuing or a comparable persistency mechanism to guarantee reliable, once-only delivery to subscribers.

Equally important for an effective EAI messaging system is an organized approach to representing the myriad information structures used by various programs. Source and target applications may use fixed-format structures, delimited structures, self-describing structures such as XML, externalized object representations, or record sets conceptually similar to relational database tables. An EAI messaging system must be able to provide transparent transport for any of these information structures, and it must facilitate easy mapping from one to another.

This mapping of one information structure to another meets part, but far from all, of the practical requirements for data transformation. In its general form, data transformation involves receiving one or more incoming event messages, performing some set of operations on their data content, and repackaging the result into one or more outgoing event messages. A data transformation service should allow a systems integrator to create transformations of at least the following basic types:

- Copy
- Look up values
- Insert constants
- Reformat
- Perform string replacements, concatenations, and other manipulations
- Perform looping, logical, and arithmetic operations.

The transformations are visually defined and reside in a transformation repository to support reuse and allow centralized management of data transformations for the entire integrated enterprise.

The concept of an adapter is simple. On one side of the adapter, it interacts with a system, database, application, or other data source using whatever techniques are most appropriate for that source, such as direct API calls, CORBA messages, database transactions, HTTP exchange, file manipulation, custom socket protocols, or one of many other possibilities. On the other side of the adapter, the adapter interacts with the EAI infrastructure in a standardized way that is completely transparent to the external data source. The key architectural result of the adapter approach is that the application becomes a source and sink for event messages to and from the EAI environment – and hence to or from other systems and applications in the EAI-enabled environment. Interaction with those other systems and applications becomes independent of computing platform, language, and database.

EAI workflow management business models are generally created in the form of flow charts, and include activities, workflow relevant data, and decisions as their main components. Activities define the steps in the business, while decisions control the flow of the business process based on logic. Steps of the workflow may require user intervention or may only require that automated actions by integrated applications be performed.

There is a natural relationship between workflow management requirements and fundamental EAI capabilities that makes EAI an ideal platform for building active workflow processes. Workflow steps that involve external systems can readily use adapter instances (generally in a request/reply mode) to control the external applications via events, a process step corresponds to one or more events, and EAI event message contents provide the containers for business data that must be transported along the steps of the workflow.

**Pepco EAI Implementation**

Beginning in late 2000, Pepco defined its EAI requirements and began selection of a set of EAI product tools through a formal Request for Proposal and vendor evaluation process. Following vendor selection and completion of a pilot project, corporate-wide integration using the EAI tools began in mid-2001. Pepco formed an EAI Implementation Team to define implementation standards, processes, and shared services. This team also provides mentorship, training, and EAI implementation services to all Pepco IT projects.

**Tactics and Strategy**

The implementation of an EAI solution requires corporate commitment; the strategic shift from a point-to-point approach to an EAI solution is a major change for most corporations. The full benefits of the solution are not realized until data made available in the initial interfaces are needed by another system. The approach used at Pepco was to start with interfaces required for new systems. Retrofit of existing interfaces or interfaces that were well into their implementation phase using other techniques was not deemed cost-effective, although opportunities to use EAI on a tactical basis were pursued. For major new systems whose overall integration was undertaken following selection of the EAI approach, a full set of messages was defined and data was normalized to meet the needs of multiple subscribing systems, including anticipated future needs where possible, in order to gain the fullest long-term benefits.

**Tactical Application**

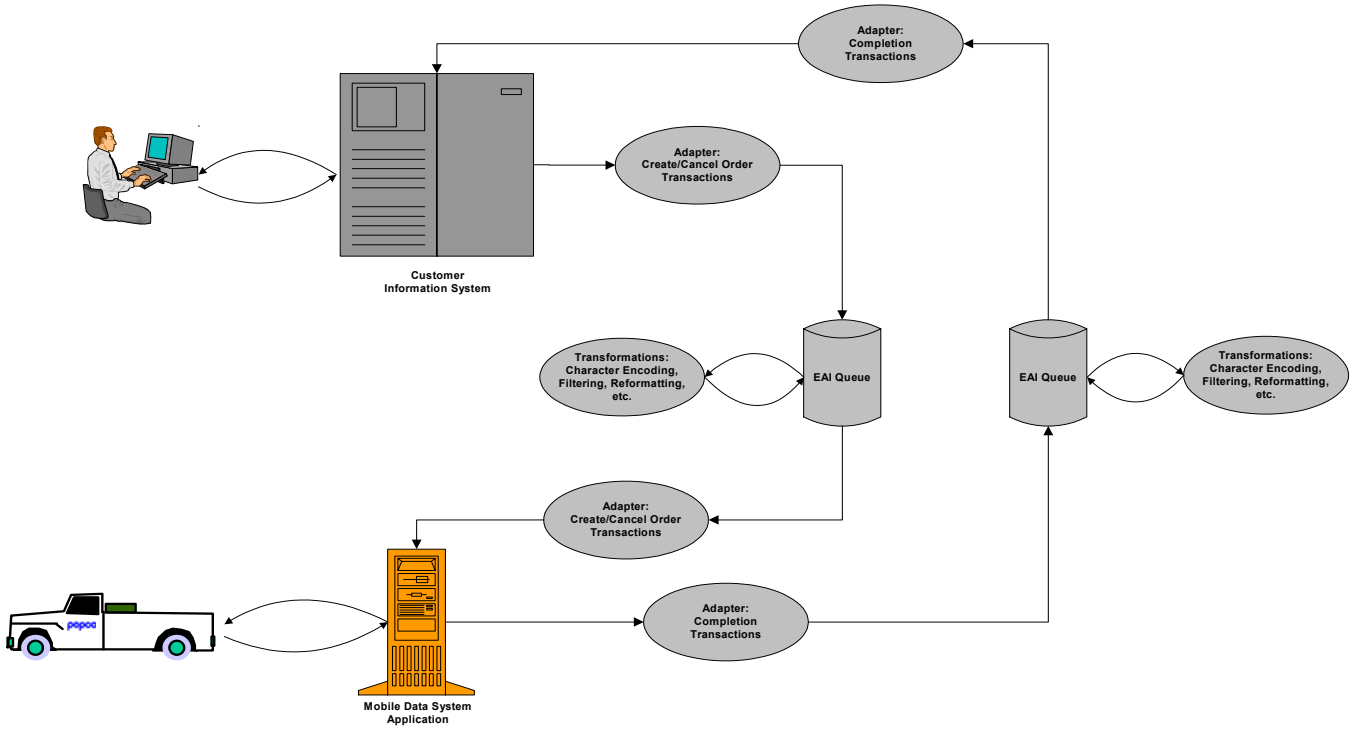
Several characteristics of the EAI tools have made them highly suited for individual, high-priority, tactical interfaces between distribution operations applications, including systems whose implementation and integration had begun prior to adoption of EAI. The most important of these characteristics are:

- Guaranteed message delivery
- Flexible data transformation
- Multi-platform support.

Using these capabilities, Pepco has developed solutions to challenging integration problems related to event-driven and high-volume interfaces between its Customer Information System (CIS) and its new Mobile Data System (MDS) and Outage Management System (OMS).

The CIS-MDS interface, shown schematically in the diagram on this page, illustrates many of the basic characteristics of any EAI solution. Customer Service Representatives enter customer disconnect and reconnect orders using their existing CIS screens. The disconnect and reconnect order information, which previously was routed to a mainframe Trouble Processing System, is intercepted and submitted to the EAI environment as an event message via an EAI API call. The order information event message (still in its original format) is placed on a queue. A transformation engine subscribes to this message, transforms it to the format required by the MDS, and republishes it. A subscribing adapter receives this information and submits it to the MDS by placing it on the queuing system used by the MDS. The MDS queuing uses a different technology than the EAI queuing, but an off-the-shelf adapter was available from the EAI vendor to accommodate it. The identical approach is used to send order cancellations from the CIS to the MDS.

The MDS continues the business process within its own system environment by sending field orders to mobile data terminals over the Pepco data radio system. Upon completion of the orders by the field crews, the order resolution information in the MDS is placed on its outgoing queue (again, a different technology from the main Pepco EAI environment.) An adapter subscribed to that queue places the message on an EAI queue. A transformation engine subscribes to this message, transforms it to the format required by the CIS, and republishes it. A subscribing adapter receives this information and submits it as a completion transaction to the CIS.



A feature that should be noted about this interface is the non-intrusive nature of the integration – no changes were required to the MDS to accommodate the interface, and minimal changes were required to the CIS. The complete decoupling of the two systems through the EAI infrastructure will allow future upgrade or replacement of either system without impact on the other.

The CIS-OMS interface also requires bi-directional data exchange between two systems that are based on very different technologies. In this case, the integration requirement is near-real-time replication of changes in a set of mainframe DB2™ source database tables to a target Oracle™ database residing on a Unix™ platform, and similar replication from Oracle source tables to DB2 as the target. Tables are of similar, but not identical, structure, and some data transformations are required due to differences in data types. The aggregate throughput requirement under high activity conditions is replication of several hundred thousand rows of new, changed, or deleted records per hour with a latency of not more than several seconds.

The replication solution uses database adapters to detect changes in either source database, format the changes into messages, and publish them to a queue. A target system adapter subscribes to these messages. Its transformation rule set performs the necessary data type conversions and creates the appropriate data manipulation language actions, and the adapter then applies these transactions to the target database. At this time, the target system (either the OMS or the CIS) is the only subscriber to each type of replication message, but the replication scheme could easily be used to keep additional databases in synchronization or to provide notifications of some or all changes to other interested data sinks by simply adding other subscribers. The general-purpose nature of this replication scheme has also proven useful to solve other integration needs at Pepco, for example, maintaining synchronization between a meter inventory system and the CIS.

These particular integration examples illustrate that an adapter for a system or application may actually consist of several adapter components. The Pepco Customer Information System (CIS) has been adapted to the EAI environment via database adapters, direct message API calls to the EAI environment from Cobol/CICS, and execution of CICS transactions through an external API call into the mainframe. CIS is now simply a source and sink for event messages to and from other systems and applications in the EAI-enabled Pepco environment, so interaction with those other systems and applications is very simple to realize. In this way, the use of EAI approaches for CIS and other applications has not only met the immediate interface requirements, it has also positioned the systems for participation in the long-term enterprise architecture.

**Strategic Application**

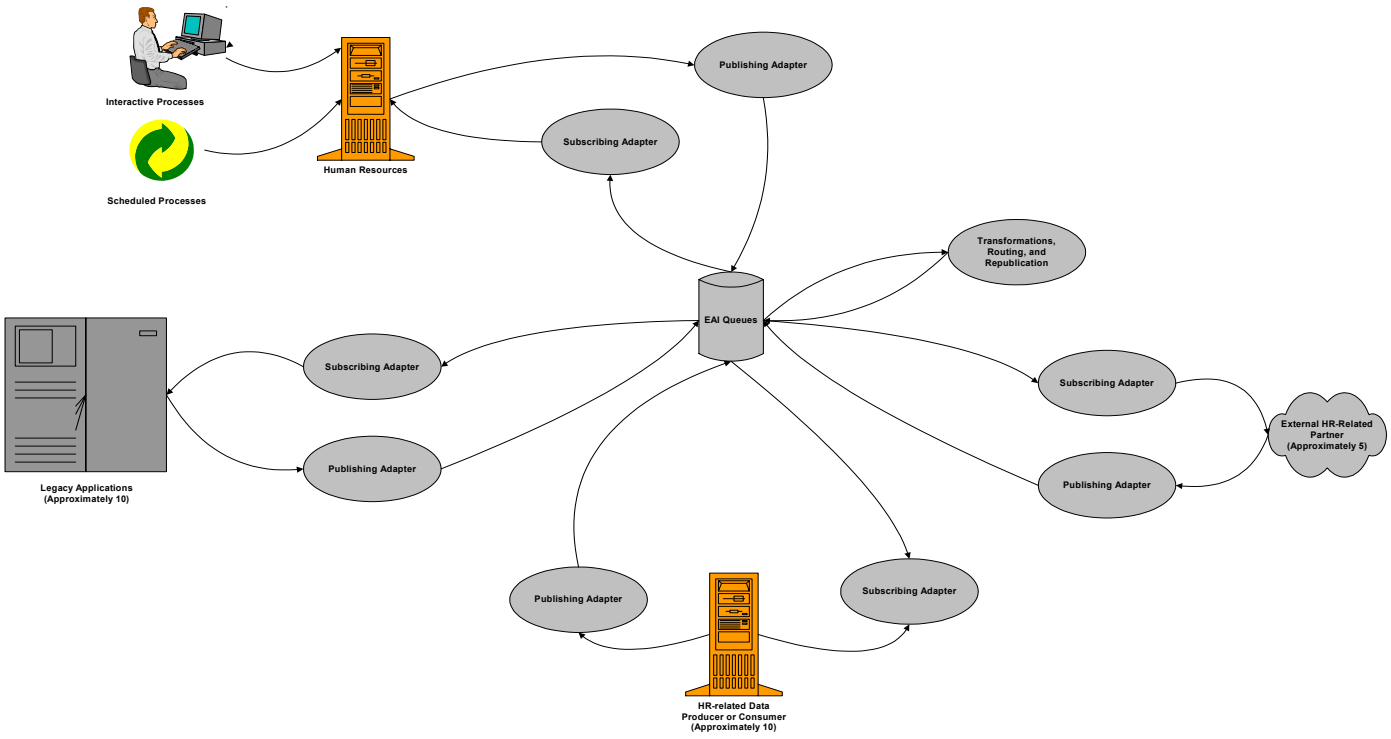
Business events are the principal mechanism for integration when using the full EAI paradigm. Rather than performing customized data transfers from one application to another on a rigid schedule, each application in the fully EAI-enabled enterprise publishes messages when significant business events occur. Other applications that require the information can subscribe to these messages through off-the-shelf or customized adapters and use the events and their contents to control the flow of an overall enterprise business process. Pepco is currently defining and creating the events and message contents required to integrate major business processes across the human resources, supply chain management, and metering areas.

Fundamental principles for the creation of the large-scale integration have included:

- Determination of the common message contents that are capable of supporting multiple interfaces and potential future subscribers

- Creation of appropriate publication channels, where needed, if the same message structure will be used in multiple contexts. For example, an Employee message may be published by the Human Resources system through its adapter following an interactive change to the employee's record. An identically structured message may be published on a periodic basis for all employees to support legacy applications that still require a periodic batch update of all employees' information.

The diagram on this page shows the typical strategic application of EAI at Pepco for large-scale integration and multiple subscriptions. Although publishing and subscribing adapters for a given application are shown as individual adapters in all cases on the diagram, they may be the same (or, indeed, may not both be needed), depending on the needs and capabilities of the application.




This enterprise-wide set of queues, publication channels, event messages, and subscriptions is the concrete realization of the abstract information bus concept that has been widely used to illustrate integration concepts.