



MIMOSA's Open System Architecture for Enterprise Application Integration (OSA-EAI) Technical Architecture Summary

December 2006

Open Architecture Requirement

Operators, maintenance personnel, logistic managers, OEM's, parts suppliers, and engineers have always wanted to have information about the condition of equipment assets at their fingertips when they need it. Unfortunately, it typically is scattered among separate information systems, one for each platform and then separated by information type: manufacturer's nameplate data, as-installed data, as-maintained data, operational data, condition monitoring data (such as vibration readings, infrared thermography, oil analysis, control device monitoring, etc.), and asset diagnostic/health and reliability data.

It is difficult, if not impossible, to view the different information types on the same computer terminal, let alone compile and synchronize them into an integrated view or report on which to base intelligent asset management decisions. Even when the systems can be accessed from the same display, it usually requires customized efforts to integrate disparate databases which use proprietary programs with different data models.

Interconnectivity of the islands of engineering, maintenance, operations, and reliability information is embodied in MIMOSA's Open Systems Architecture for Enterprise Application Integration (OSA-EAI) specifications. Previously, these separate information islands were built using specialized proprietary systems that provided value because they were optimized for a specific task or tasks, and they provided best results and value for those purposes. However, their combined value can be multiplied significantly if they can be merged into an OpenO&M™ information "data network".

The "data network" can be building open OSA-EAI bridges to proprietary data stores to allow this information to be easily understood and utilized. The following discussion outlines some of the advantages and limitations of each approach.

Building custom bridges

The Collaborative Maintenance network can be developed by constructing custom software bridges between all the disparate data systems which may be required. This normally involves contracting to an integration company or utilizing in-house information technology resources. However, this route can be very-time consuming and have a very high initial cost and high on-going maintenance costs.

Advantages include:

- High level of customization for commercial and military needs
- High performance tuned to specific bandwidth requirements

Limitations include:

- High risk due to unforeseen incompatibility issues
- High cost due to lack of multiple users
- Difficulty in resolving problems among application suppliers (possible finger-pointing)
- Possible dependence on proprietary interfaces
- High annual software maintenance cost. These costs typically run around 20 percent of original cost which translates to \$100,000 annual maintenance costs for a \$500,000 software integration project.

Using an open EAI systems bridge

A number of the limitations inherent in custom bridge solutions can be overcome by using an open Enterprise Application Integration (EAI) specification. In mature sectors, this translates into plug-and-play capability that allows a company's information services department to hook up any compliant product to the network.

Advantages include:

- Engineered plug-and-play system capability designed up-front into system
- No burden of on-going integration efforts
- More freedom to choose best technology from information supplier (plug and play)
- Creation of the information backbone of e-maintenance

Limitations include:

- Necessity for suppliers to support industry standard
- Standard gateway may exhibit some performance degradation when custom interface is supplied

Definition of an open system

According to the Software Engineering Institute (SEI) at Carnegie Mellon University (www.sei.cmu.edu/opensystems/glossary.html), a specification is open if its interface is fully defined and available to the public and it is maintained by a group consensus process.

The SEI goes on to define an open system as a collection of interacting software, hardware, and human components:

- Designed to satisfy stated needs
- With interface specifications of its components that are fully defined, available to the public, and maintained according to group consensus
- In which the implementations of the components conform to the interface specifications.

It follows that open system architecture would be made up of components, both hardware and software, that are specified in an open manner.

A number of open EAI consensus-based specifications have been developed or are in development for various information sectors. Those information sectors, specifications, and consensus-building organizations include:

- Control systems and production schedulers: OPC; OPC Foundation (www.opcfoundation.org) and ISA—The Instrumentation, Systems, and Automation Society (www.isa.org)
- Engineering product data management systems: STEP; Standard for Exchange of Product Model Data, ISO 10303 STEP (http://cadd.cern.ch/cadd_step.html)
- Enterprise resource planning (ERP) systems: OAGI; Open Applications Group, Inc. (www.openapplications.org)
- Condition monitoring systems: MIMOSA (www.mimosa.org)
- Maintenance scheduling (CMMS/EAM) systems: MIMOSA (www.mimosa.org)

- Predictive reliability systems: MIMOSA (www.mimosa.org)

Operating in close association with MIMOSA is the International Standards Organization's (ISO) Technical Committee 108, Subcommittee 5—Condition Monitoring and Diagnostics of Machines (www.iso.ch). Specifications which emerge from this subcommittee are regularly reviewed and adopted into MIMOSA to ease adoption of the standards.

Brief Background on XML

XML is often compared to hypertext markup language (HTML), the document language of the Internet's World Wide Web. HTML is a set of rules used to describe how a Web document should look on your computer screen. Tags enclosed by angle brackets in the text of the document file on the server computer indicate how various passages are to be interpreted by the browser on the client computer requesting the page. For example, the tags for the title of this section might be coded as `<p>Brief Background on XML</p>` for display on a Web page. The tags indicate that the text is a paragraph to be set in boldface type one size larger than the regular font size. This approach works well for documents, but not for structured data.

XML is a set of rules used to create a special markup language for exchanging structured information over the network. It is used to describe both structure and content of the document; how the document should look is handled separately. The rules of XML tagging (mark up) are tightly defined by the World Wide Web Consortium (W3C), but not the language's structure and vocabulary. Tag vocabulary is defined by a consortium or a corporation and varies according to document content. A MIMOSA asset-related section of an XML document looks like:

```
<asset
  asset_org_site="1367960"
  asset_id="123497423"
  user_tag_ident="E3923-271-02"
  name="Brand X Model Y Part #323-4 S/N: E32923-271-02"
/>
```

The definition of the XML documents are explained in XML Schema Definition files (XSDs). XML rules ensure that the specialized languages created by XML can be processed on compact computer programs called parsers. Humans can also interpret the XML code. Many of the elements in the previous example taken from the MIMOSA demonstration project are easily recognized without referring to a schema for the data. Further processing is needed to make the information easily read by people.

XML does not include rules for displaying information. Sorting and formatting of the information in the XML document is handled by stylesheet. The stylesheet approach allows a single set of data, such as a work order, to be specially rendered for different devices. The data could be formatted for a desktop computer screen or a palm computer display, or even audible speech.

Advantages of XML

XML provides a standard by which the content structure of a wide variety of information, from simple to complex, can be marked up for easy transfer over the network for use on a variety of computing platforms. XML can be used to mark an ordinary document, a structured record (work order or purchase order), a data record (query result), graphical presentation (user interface), standard schema elements, and more.

Legacy data encoded in XML can be delivered over a local area network or the Internet without having to be changed. Once on the client computer, it can be edited, manipulated, and presented in various views without return trips to the server computer. The widespread adoption of XML should speed up the Internet because more processing could be done locally, reducing bandwidth loads for traffic between the client and server.

MIMOSA's OSA-EAI Specification

The Open System Architecture for Enterprise Application Integration (OSA-EAI) specification has been developed by MIMOSA. MIMOSA is a trade association composed of industrial asset management system providers and industrial asset end-users which develops information integration specifications to enable open, integrated solutions for managing complex high-value assets.

MIMOSA's OSA-EAI system specifications offer advantages for maintenance and reliability users as well as technology developers and suppliers. For users, the adoption of MIMOSA OSA-EAI specifications facilitates the integration of asset management information, provides a freedom to choose from a broader selection of software applications, and saves money by reducing integration and software maintenance costs.

For technology suppliers, the adoption of MIMOSA OSA-EAI specifications stimulates and broadens the market, allows concentration of resources on core high-value activity rather than low value platform and custom interface requirements, and provides an overall reduction in development costs.

Taken as a whole, maintenance and reliability information is complex. A Collaborative Maintenance network must provide for the open exchange of equipment asset related information between condition assessment, logistic, and maintenance information systems. The condition assessment sector must include the specialized data required by vibration, oil analysis, infrared thermography, motor circuit evaluation, and many other technologies.

Prior to MIMOSA, developers defined data fields to fit their own hardware and software systems. MIMOSA's experts from across the globe have spent more than 10 years in developing the Open System Architecture for Enterprise Application Integration (OSA-EAI) specification (see Figure 1) so that all loosely-coupled systems can exchange information by using MIMOSA's open system architecture.

MIMOSA OSA-EAI V3.1 Release – December 2006

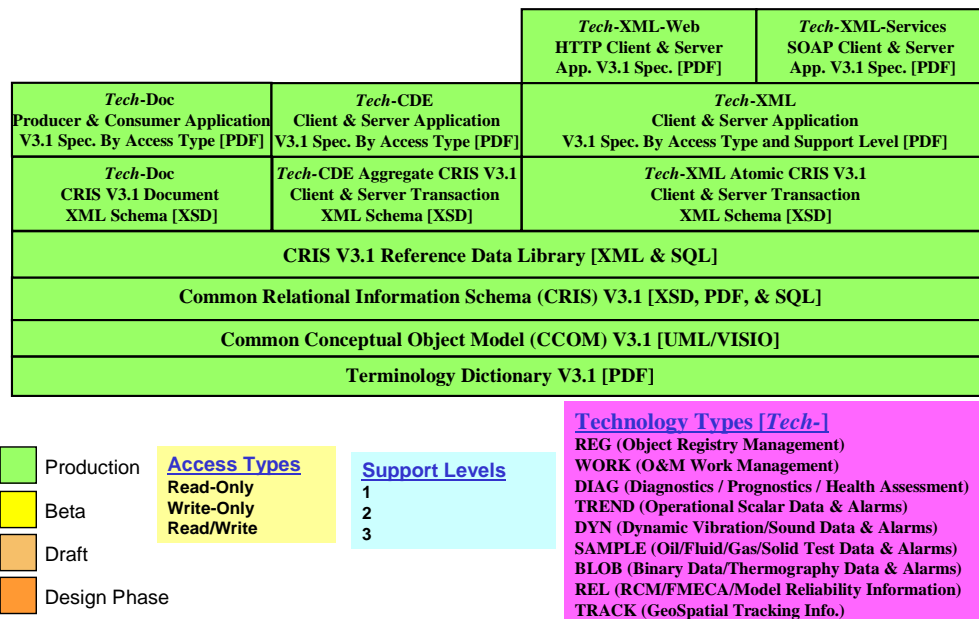


Figure 1. MIMOSA OSA-EAI Architecture Diagram

OSA-EAI Terminology Dictionary

To ease understanding among all parties using MIMOSA's OSA-EAI specification, MIMOSA provides a standard set of terminology in the OSA-EAI Terminology Dictionary. This provides the basic semantic descriptions used throughout the OSA-EAI specifications.

Common Conceptual Object Model (CCOM)

The Common Conceptual Object Model (CCOM) provides the basic conceptual model basis for OSA-EAI. Software designers familiar with class diagrams in Unified Modeling Language can utilize this model to understand the basic classes, major attributes, and relationships between classes in OSA-EAI. CCOM V3.1 is available in PDF document form and Visio (VSD) format.

Common Relational Information Schema (CRIS)

MIMOSA's Common Relational Information Schema (CRIS) provides a common implementation schema which allows information from many systems to be communicated and integrated. The schema is in a relational format, commonly used in most database systems. Each table in CRIS is assigned a unique table reference number. CRIS V3.1 is published as an XML Schema (XSD). Documentation is available in a Web Browser (HTTP) format and PDF format.

Data sources which house asset information are not required to be physically re-designed to match CRIS, but must be able to translate their information into CRIS tables and columns. In order to ease this translation and for MIMOSA implementers who desire to implement a physical CRIS database, MIMOSA provides ORACLE and Microsoft SQLServer table creation scripts.

OSA-EAI contains a mandatory unique identification methodology which allows the integration of enterprises, sites, reference databases, functional segments, assets, measurement locations, and agents identification nomenclature. An OSA-EAI **Enterprise** is the corporate level of an organization, or the top organizational structure of a non-profit or military body. An enterprise is composed of many Sites. The Enterprise uniquely identifies the **Sites** it manages. The OSA-EAI Enterprise Unique Integration Code (Enterprise-UIC) is a globally-unique identifier defined as a 4-byte, non-negative integer which is assigned by the MIMOSA organization. Normally, MIMOSA issues one Enterprise UIC per corporation/organization. For some organizations, multiple Enterprise UICs may be requested. In addition, MIMOSA will also assign the Enterprise with a globally-unique, alpha-numeric "user tag" identifier value. A representative from the registration authority for an organization should e-mail the MIMOSA Enterprise Registrar at info@mimosa.org with the name of the organization, requested enterprise alphanumeric User Tag Identifier, contact name, title, phone number, and e-mail address. The MIMOSA Enterprise UIC Registrar will then assign an Enterprise UIC with associated Enterprise User Tag Identifier and return the assigned non-negative integer to the point of contact.

All OSA-EAI **Sites** are generated by a "birth" **Enterprise**. A **Site**, in turn, "births" serialized **Assets**, functional **Segments** (where **Assets** can be installed over time), **Databases**, **Agents**, and **Measurement Locations**. A given Enterprise can contain many sites. A Site can contain many Segments. For facility applications, the Site normally represents a tangible as-built building. For industrial and manufacturing applications, this entity normally represents a physical plant. For fleet applications, this entity might represent a maintenance depot which has responsibility for assets. Some very large "self-managed" assets, such as an aircraft carrier, may be defined as a Site itself. Because of multiple methodologies for site/plant/platform identification in different industries and in various computer systems, a **Site** will often need to be associated with many identifiers relevant to the **Site** because they are the site's "unique key" for various databases. These identifiers may change as a **Site** is sold, transferred, or its role changes in relationship to multiple other **Sites** or **Enterprises**. Because of this

requirement, MIMOSA-compliant systems require that an Enterprise authority assign all **Sites** integrating their information be permanently assigned one and only one **Site Unique Integration Code (Site-UIC)** which is composed of the MIMOSA-assigned **Enterprise-UIC** concatenated with a site-unique, non-negative 4-byte integer (in CRIS, the "site_id") which should never change throughout the lifetime of the **Site**. This **Site-UIC (Enterprise UIC + "site_id")** does not have any external end-user meaning, but is the vital link for all MIMOSA-compliant systems to properly integrate and associate information to the correct **Site**. The identical **Site-UIC** should remain permanent for the lifetime of the **Site**, even if the **Site** is sold to a different **Enterprise**. To keep the number of primary key columns to a minimum, CRIS V3 compresses the **Site-UIC** into a fixed-length, 16-character string. This is called the "site code" and is composed of the two 4-byte non-negative integers converted into their hexadecimal format (resulting in 8 characters per integer) and then concatenated into a fixed-length 16-character string.

OSA-EAI also supports **Models** of serialized **Assets**, along with their functional design segmentation (**Segments**). OSA-EAI also provides for a method of standard measurement location identification across various condition monitoring technologies. Trendable, scalar data such as operational temperatures, pressures, and loads are modeled in OSA-EAI. OSA-EAI also supports dynamic data, such as time waveforms and Fast Fourier Transforms (FFTs), which are used in vibration analysis and sound monitoring. Binary data, known as Binary Large Objects or (BLOBs), are supported for communicating drawings, reports, diagrams, and photographs. OSA-EAI also manages sampling test data results, such as used oil analysis test data and air quality monitoring data. The specification also allows the communication of diagnostic, health, and prognostic information from smart systems and eases the generation of advisory recommendations. Special maintenance and reliability tables define fields for events (actual, hypothesized, proposed), health and estimated asset life assessment, and recommendations. In addition, OSA-EAI supports maintenance and production work request scheduling and the tracking of the completion (or non-completion) of a maintenance or production job as related to an asset. OSA-EAI also provides the information framework for storing reliability data for assets.

CRIS Reference Database Specification

Because OSA-EAI is designed for multi-site collaborative asset lifecycle management, nearly all tables in CRIS (except a few MIMOSA-managed reference tables) include a primary key reference to a Database-UIC. OSA-EAI contains many classification taxonomy meta-data "type" tables, which allow users to classify types of enterprises, sites, segments, assets, agents, measurement locations, engineering units, etc. with standard numeric codes, common throughout all their various systems. MIMOSA generates and maintains industry-standard taxonomies and codes for many of these tables, but the MIMOSA encourages both suppliers and end-users to collaborate to expand on

the MIMOSA entries with industry-specific entries and submit these to MIMOSA for multi-industry taxonomy harmonization. End-user specific classification entries are also supported. MIMOSA generates the CRIS Reference Database Specification in XML, ORACLE, and SQLServer format. Version 3.1 of this database specification contains many useful codes which allow standardization across many disparate systems. For example, the **Asset Type** table allows standard querying of common asset types such as "AC induction motor" which has an AssetType-UIC assignment of "0000000000000000-0-737" (Database-UIC "0000000000000000-0" + Asset Type integer identifier "737"). Other standard code tables include service segment, measurement location, engineering units, sampling test codes, diagnostic/prognostic event codes, health codes, failure codes, and root cause codes. This allows systems to search across various systems for common failures on certain equipment types.

Tech-Doc XML Schema

The *Tech-Doc* XML Schema specifies the format of an XML document which can contain an unlimited volume of CRIS-compliant data from a data source. Binary data is BASE64-encoded.

Tech-Doc Producer/Consumer Application Specification

A *Tech-Doc* Producer system produces a stream or file of *Tech-Doc* XML data (using any transport method) which is compliant with the *Tec-Doc* XML schema. A system which consumes a set of this XML data is called as a *Tech-Doc* Consumer. *Tech-Doc* Consumer Read-only applications will utilize a *Tech-Doc* XML document, but not change permanent data storage based on this information. *Tech-Doc* Consumer Write-only applications change their permanent data storage after successful file import.

Tech-CDE [Compound Document Exchange] Client/Server XML Schema

The *Tech-CDE* (Compound Document Exchange) Client & Server XML Schema was developed as a means of transferring aggregate, related sets of CRIS data in XML format using one client-server query schema and one client-server write schema. Because the client has access to a broad query range, normally a server system can natively access CRIS content. The *Tech-CDE* Specification consists of two main schemas, a query and a write schema. There is also a supporting schema, V3-1TechCDE_CrisVocabAddition.xsd, which contains the CRIS vocabulary elements used by both schemas. Binary data is BASE64-encoded.

Tech-CDE [Compound Document Exchange] Client/Server Application Specification

A *Tech-CDE* Client system pulls or pushes a related set (aggregated compound document) of CRIS data (using any transport method) to a Server in a transaction which is compliant with the *Tech-CDE* XML schema. A Server system which responds to a Client's "pulls or pushes" of CDE requests for OSA-EAI XML data (using any transport method) is known as a *Tech-CDE* Server. The *Tech-CDE*

Client/Server Application specifies the contents of a client/server data exchange, but does not specify the physical transport method.

Tech-XML Client/Server XML Schema

The *Tech-XML Client & Server XML Schema* was developed as a means of transferring atomic, small packets of CRIS data in XML format with a specific XML schema defining each transaction. Because a client can only specify certain parameters in a query (read) or create (write) transaction, server system can decide which interfaces to support and map their proprietary data into CRIS-formatted XML. Binary data is BASE64-encoded. Three support levels (1, 2, and 3) of interfaces are defined, where support level 1 is the most basic support for a technology.

Tech-XML Client/Server Application Specification

A *Tech-XML Client* system pulls or pushes a pre-defined atomic set of CRIS XML data (using any transport method) to a Server in a transaction which is compliant with one or more predefined *Tech-XML* schemas. The *Tech-XML Client/Server Application* specifies the contents of a client/server data exchange, but does not specify the physical transport method.

Tech-XML-Web Client/Server Specification

The OSA-EAI *Tech-XML-Web* client/server specification provides the necessary transport details required for a *Tech-XML* server or a client to be implemented using HTTP or HTTPS (secure) protocols (non-SOAP).

Tech-XML-Services Client/Server Specification

The OSA-EAI *Tech-XML-Services* Client/Server specification provides a Web Service Definition Language (WSDL) and a definition of the SOAP Web Service transport required for *Tech-XML* client or server systems.

Technology Types

Tech-Doc, *Tech-CDE* & *Tech-XML* in Version 3.1 are further sub-divided into nine (9) Technology Types (see Figure 1 above). This allows developers to focus on supporting only what is relevant to their particular application area, such as vibration analysis or maintenance management. The technology types are listed in Table 1 below:

Technology Types	Description
Registry Management (REG)	Allows retrieval of "as-designed" segment hierarchical breakdown of facility, process, and machine systems, along with the "as-installed" asset information. Also allows access to name plate and image data on individual assets and models, including component part breakdowns. Used by: - OEM Model Information Systems - Asset Registry Information Systems - Maintenance Management Systems - Piping & Instrumentation Design Systems
Reliability (REL)	Allows retrieval of reliability study information related to a service segment, a model of an asset, or a serialized asset. Allows the retrieval of information associated with hypothetical causal networks, reliability-centered maintenance information, as well as actual failure events. Used by: - Reliability Information Systems - FMECA Analysis Systems - RCM Analysis Systems
Work Management (WORK)	Allows the creation and audit tracking of a new work request in a work management system for a service segment or a serialized asset. Allows the retrieval of work orders and work order steps, and actual work completed information.. Also allows the retrieval of pre-planned work packages ("solution packages"). Used by: - Maintenance Management Systems
Diagnostics / Prognostics / Health Assessment (DIAG)	Enables retrieval of human or "smart-agent" generated current and/or future proposed asset health states, current and/or future proposed diagnostic failure modes and casual trees, remaining useful life predictions, and recommendations. Also allows access to measurement evidence supporting the diagnoses/prognoses. Used by: - Diagnostic Systems - Prognostic Systems
Dynamic Vibration/SoundCondition Monitoring (DYN)	Enables the creation and retrieval of historical dynamic measurements (used with vibration and sound monitoring and including frequency spectra measurements and time waveforms), abnormal data alarms, and operational event logs. Used by:

	<ul style="list-style-type: none"> - Vibration Condition Monitoring Systems - Sound Condition Monitoring Systems
Static "Trendable" Condition Monitoring (TREND)	Enables the creation and retrieval of historical scalar measurements, abnormal data alarms, and operational event logs. Used by: <ul style="list-style-type: none"> - Process Data Historians - Process Condition Monitoring Systems - Used by Operational Data Systems
Oil/Fluid/Gas/ Solid Tests Condition Monitoring (SAMPLE)	Enables the creation and retrieval of historical fluid, air, and solid sampling data, abnormal data alarms, and operational event logs. Used by: <ul style="list-style-type: none"> - Oil Sampling Condition Monitoring Systems - Air Sampling Condition Monitoring Systems - Solid Sampling Condition Monitoring Systems
Binary Data / Thermography Condition Monitoring (BLOB)	Enables the creation and retrieval of historical binary large objects (BLOB) measurements (used with thermography and imaging monitoring), abnormal data alarms, and operational event logs Used by: <ul style="list-style-type: none"> - Thermographic Condition Monitoring Systems - Image Monitoring Systems
Physical GeoSpatial Tracking (TRACK)	Enables the storage of physical geo-spatial location information associated with assets or segments Used by: <ul style="list-style-type: none"> - Logistic Management Systems - Geographic Information Systems

Table 1. OSA-EAI Application Technology Packages

To refer to the entire set of 9 Application Technology Packages, the italicized prefix "*Tech-*" is used. Each vertical application technology only has certain CRIS tables which are relevant. This allows implementers to pair down the entire CRIS specification to only the application-specific tables.

OSA-EAI Interface Catalog and OSA-EAI Compliance Process

All OSA-EAI interfaces are assigned a 9-digit number, published in the **OSA-EAI V3.1 Interface Catalog**. Applications may support one or more OSA-EAI interfaces. System designers need to decide which interfaces are applicable for a given system and then write these OSA-EAI interfaces. To ensure acceptance by the user community at large, the use of the MIMOSA logo and the trademark MIMOSA Compliant[®] must be limited to those products that have met MIMOSA's requirements for compliance. Product buyers must have confidence that the logo and trademark on a product means that the product has the "stamp of

approval” from MIMOSA. MIMOSA has chosen to support compliance and certification through a process of self-testing using toolkits and/or a certification procedure supplied by MIMOSA. Both servers and clients can be certified. Since the needs of client applications vary greatly and no other application is dependent on them, there is no testing tool provided for testing client applications.

MIMOSA sponsors have access to compliant self-certification tools and may submit each major release of their products to MIMOSA as MIMOSA-compliant to a specific set of OSA-EAI interfaces. After MIMOSA reviews the submission and validates the compliance, sponsors are granted permission to mark their products with the MIMOSA-owned registered trademark "MIMOSA-compliant®" brand. Companies which are not MIMOSA sponsors or companies who produce products which have not complied with MIMOSA's compliancy requirements are not granted permission to use "MIMOSA-compliant" on their products. MIMOSA will post all currently certified *MIMOSA Compliant*® products on its website, listing the set of OSA-EAI interfaces supported.