PENNSTATE ARL 1855

Applied Research Laboratory
The Pennsylvania State University

# OSA-CBM: An Open System Architecture for Machinery Monitoring

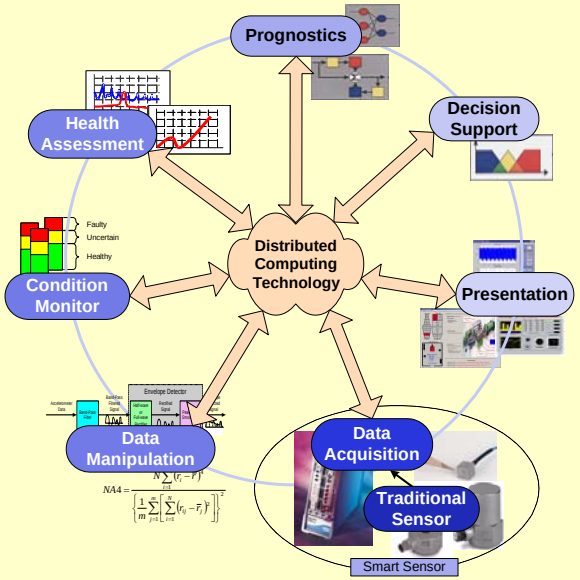
*Structural Health Monitoring COE Inaugural Meeting*
*– April 13th 2007 -*


***Contact at ARL***

***Mitchell Lebold***
*814.865.8958 (voice)*
*814.863.0673 (Fax)*
*Lebold@psu.edu*
*http://www.arl.psu.edu*

# Overview

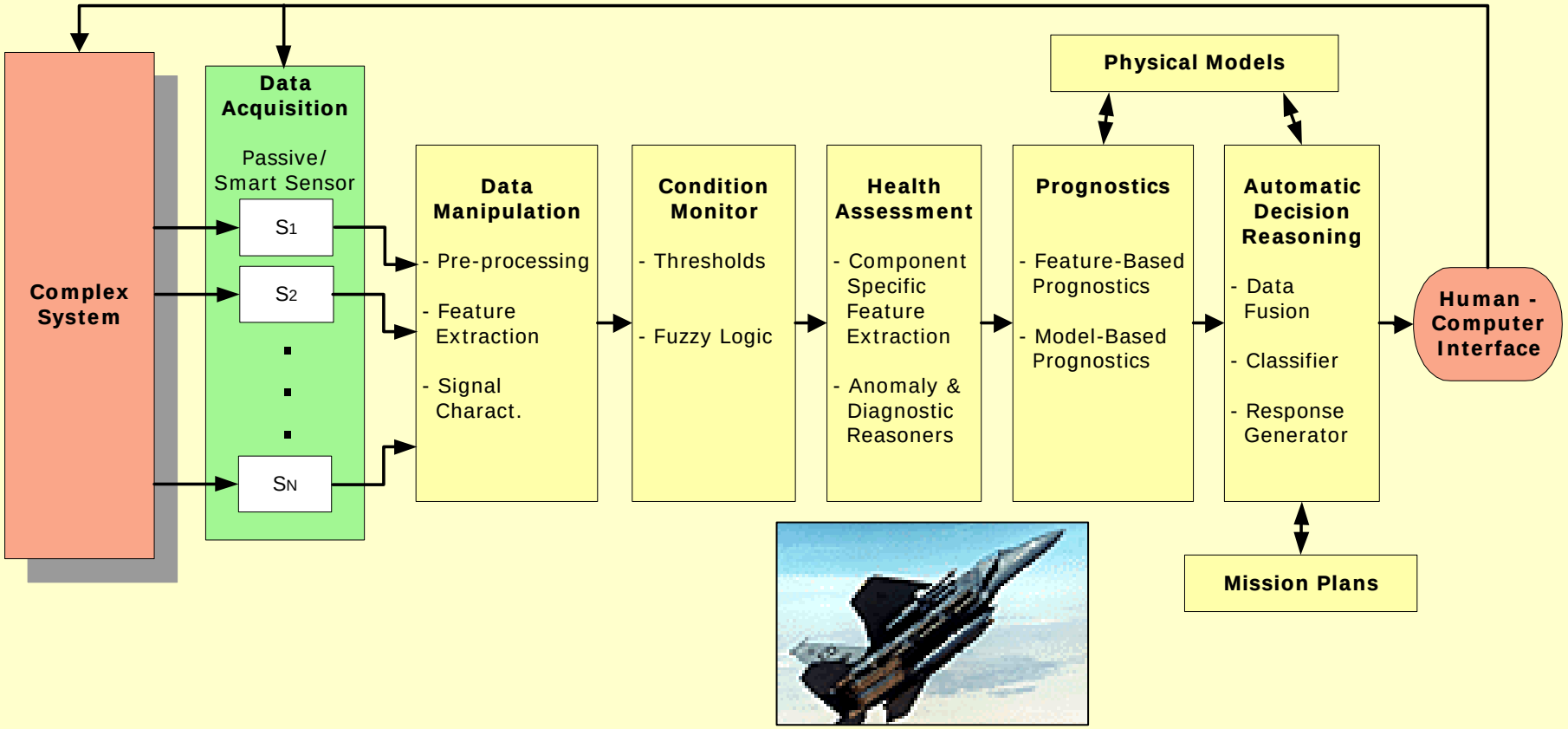
- Condition Based Maintenance (CBM) Systems
- OSA-CBM Development Team
- OSA-CBM Architecture
  - Data and Control
  - Data Models
    - UML
    - AIDL
  - Middleware Technologies
    - CORBA
    - COM/DCOM
    - XML over HTTP
  - Design Flexibility and Examples
- OSA-CBM Website




# Typical Condition Based Maintenance System Design

# CBM Systems

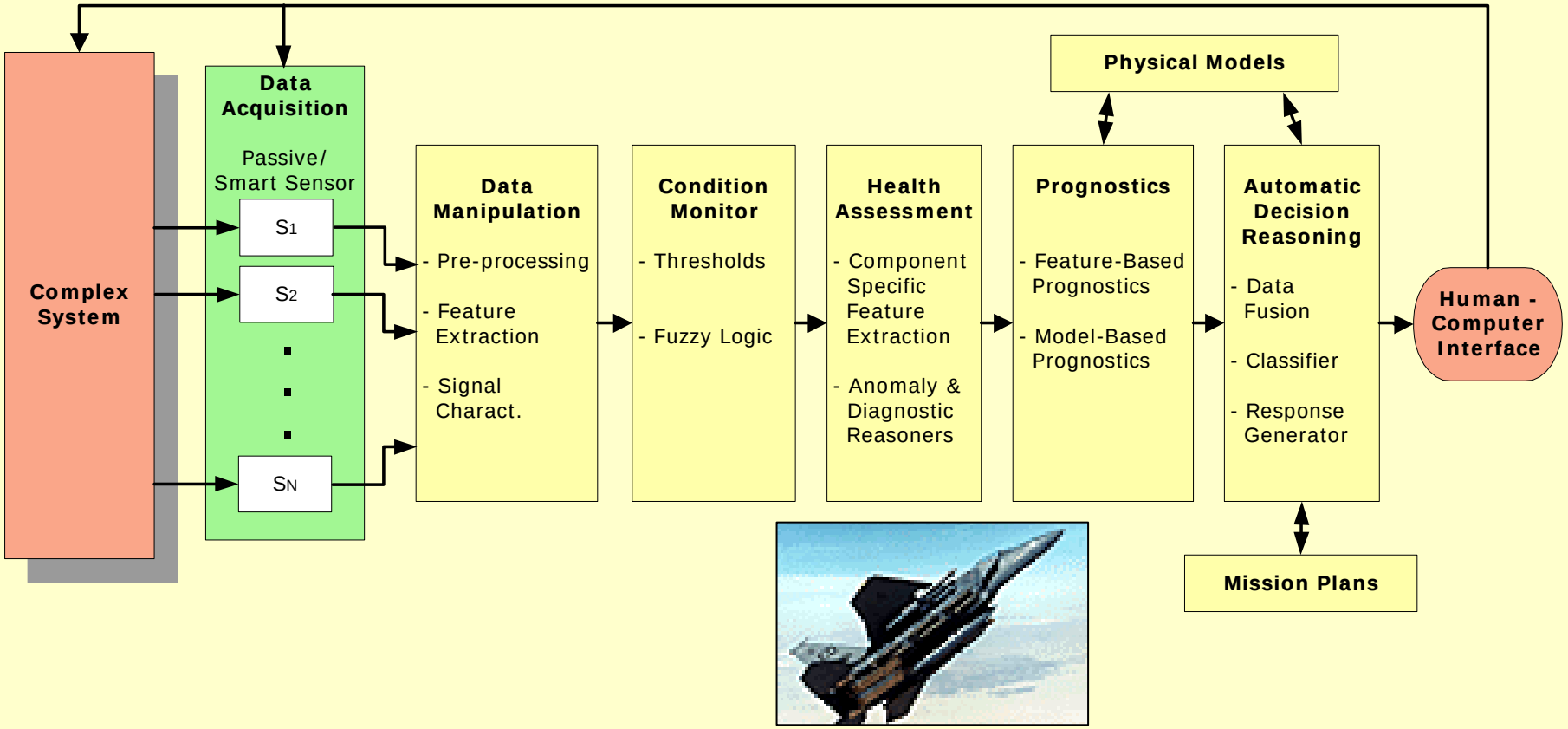
## Typical CBM System Design:



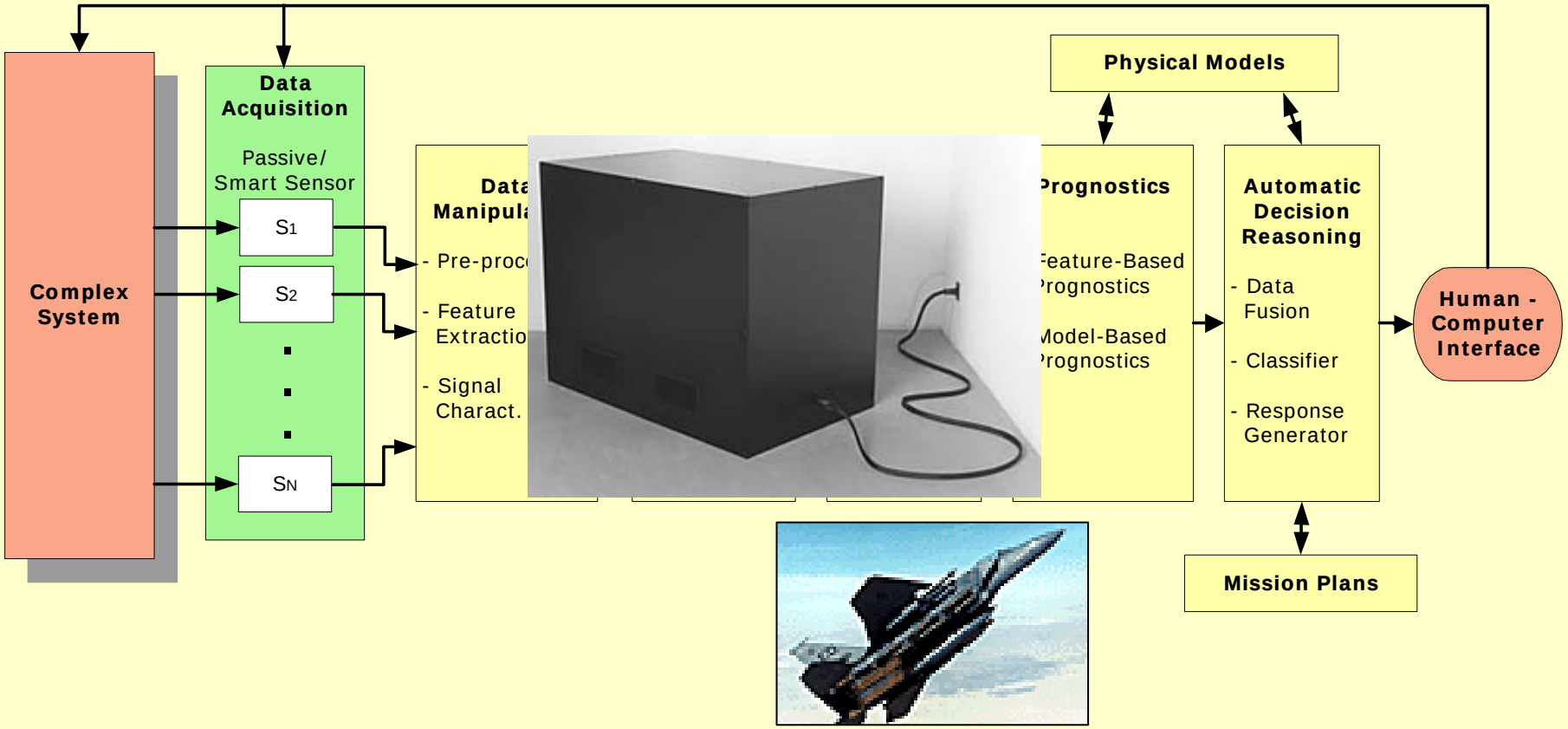
# CBM Systems

## General Concerns:

- Complex design
- Propriety algorithms
- Large amounts of data
- Inflexible designs



# Resulting in "*Black Box*" and "*Stove Pipe*" Solutions:



# OSA-CBM Development Team

# OSA-CBM Development Team





- 2001 - 2 year DUST program sponsored by ONR and PEO Carriers:
  - The Office of Naval Research under research grant number: N00014-00-1-0155 OSA-CBM Boeing DUST
- Numerous team members and contributors: Boeing, Caterpillar, Rockwell Automation, Penn State/ARL, Newport News, Oceana Sensor Technologies, Rockwell Scientific, MIMOSA, RLW, …







# OSA-CBM Focus

## Open System Architecture for Condition Based Maintenance

- Goals of the OSA-CBM program:
  - Define and Development open standards for distributed Condition Based Maintenance
  - Define an open architecture not exclusive to any specific hardware implementations, operating systems, or software technology
- Software Architecture Description
  - Define classical CBM functions and behavior
  - Define module interfaces
  - Developed multiple middleware implementations

# Why OSA-CBM?

***System Developer/Engineer***:


- Only one non-priority interface to learn
- Standardized information exchange methods

***Project Manager***:

- Reduce integration issues
- Increased security
- Remote module instantiation
- Broad choose of technologies

***OEM / Company***:

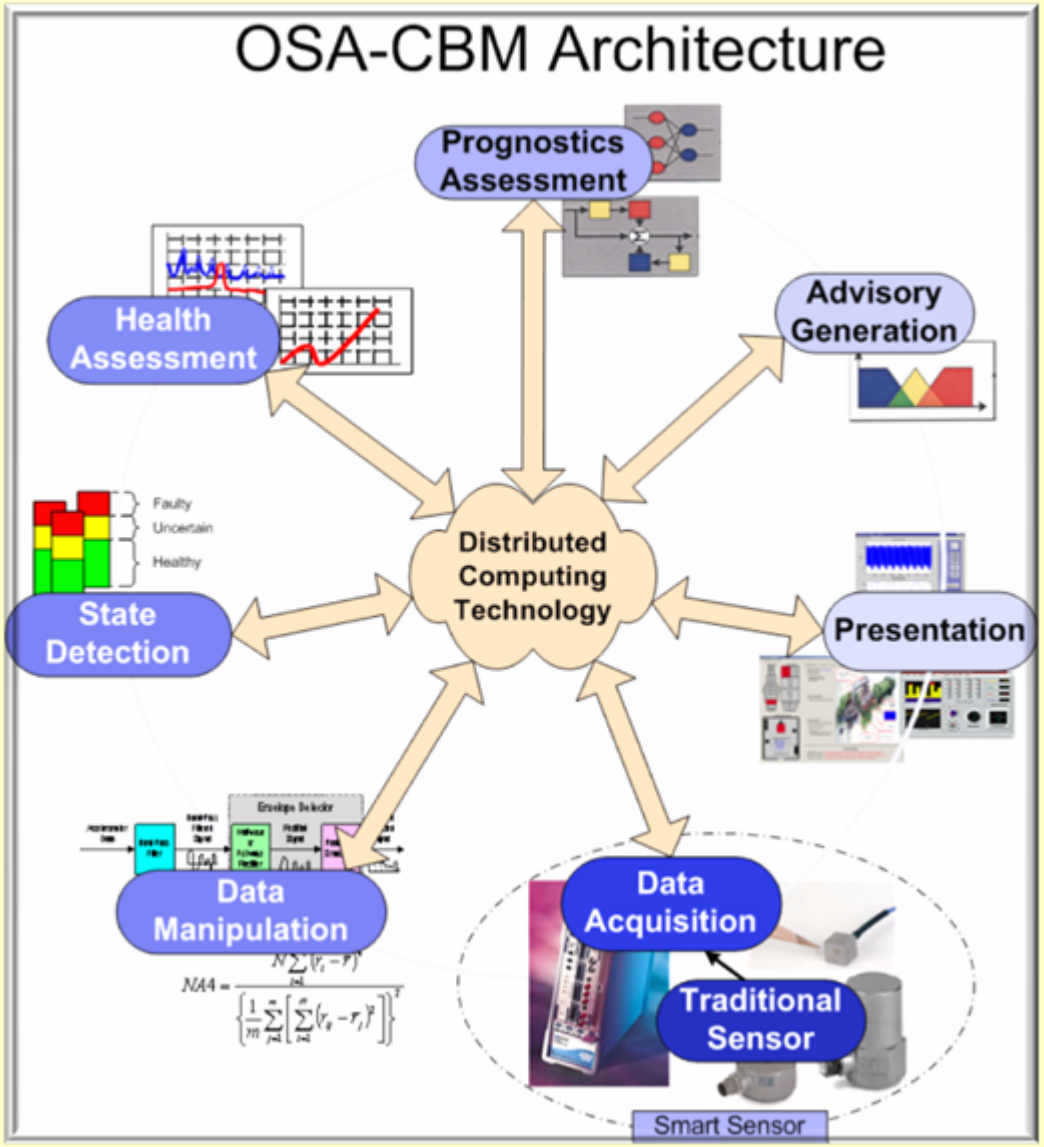

- Reduce time-to-market
- Increase profit margins

***Customer***:


- Scalability, upgradability, and interchangeability
- Broader supplier community
- Increased market competition resulting in decreased prices

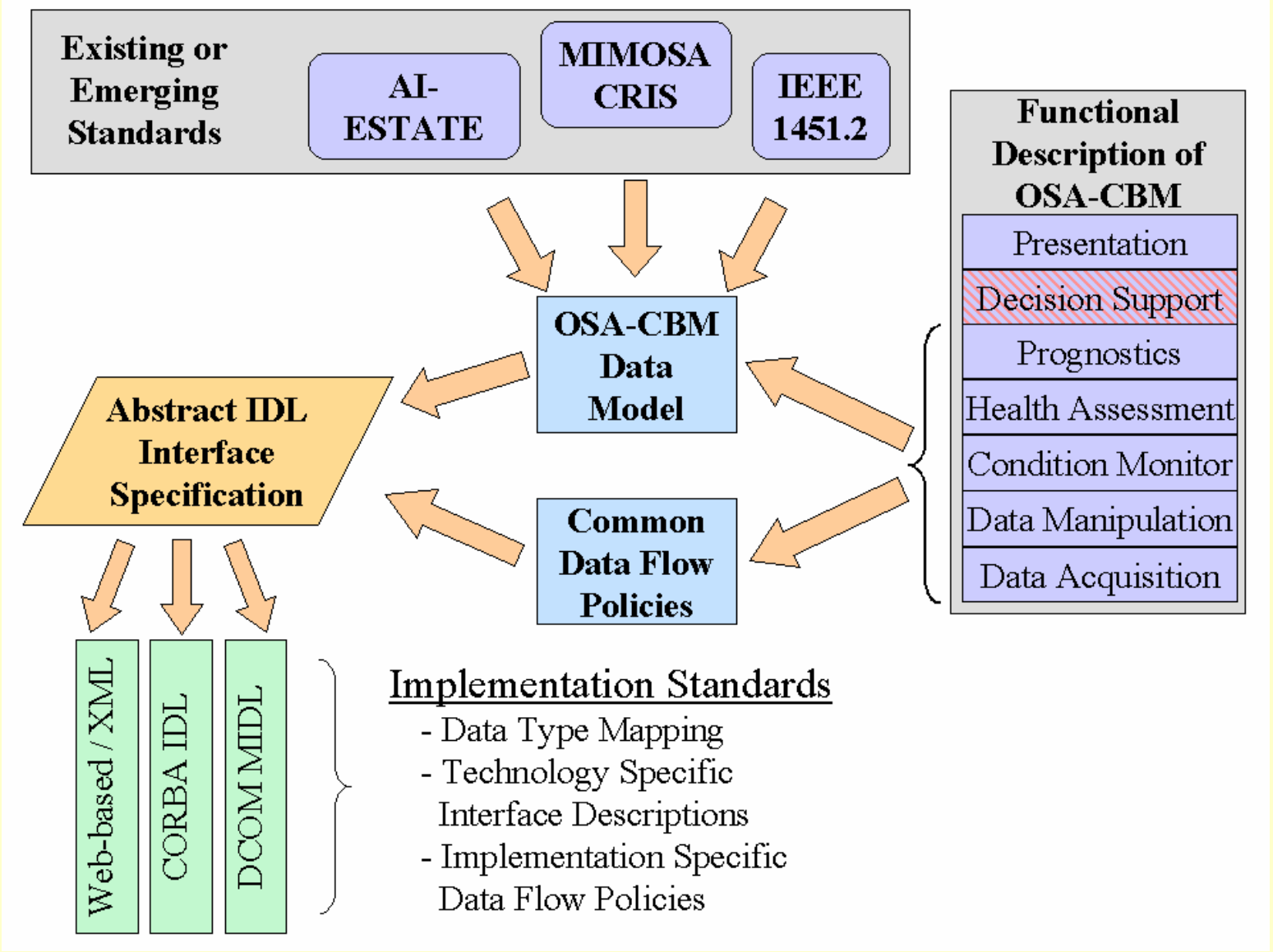
# OSA-CBM Architecture

# Open System Architecture for Condition Based Maintenance

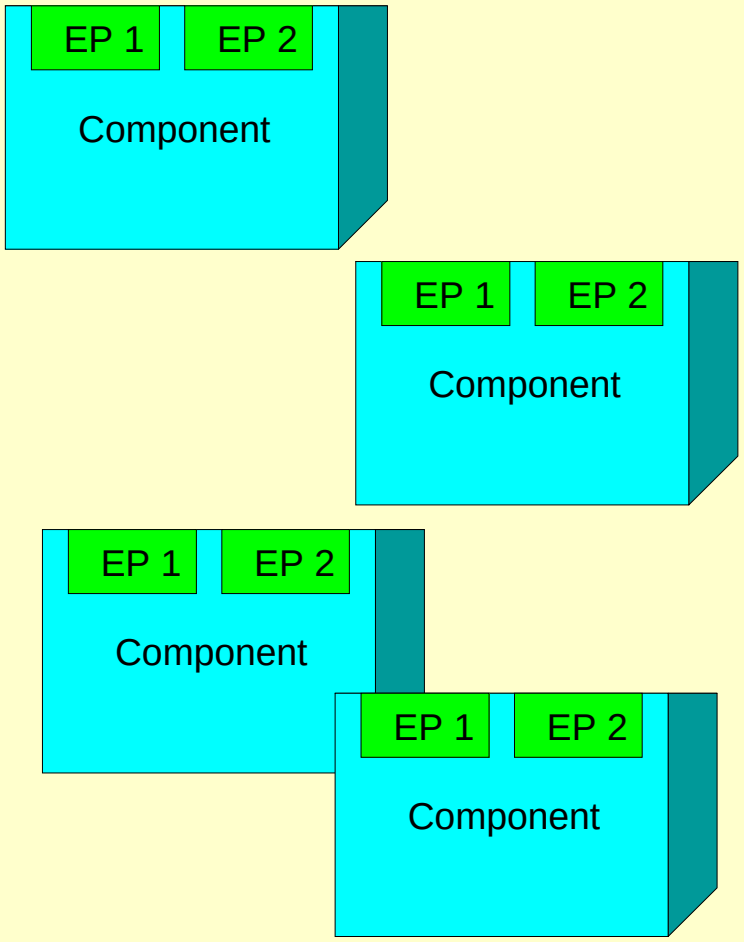



- Standardized architecture for health and condition monitoring systems
- Breaks monitoring system into functional layers
- OSA-CBM standard defines I/O for each processing layer
- Promotes scalability and upgradability
- Modules not confined to one locale
- Middleware technology independent
- www.osacbm.org

# Components of the OSA-CBM Framework



# Generic Component Description




## OSA-CBM Software Components

… implement the functionality of individual layers of the architecture

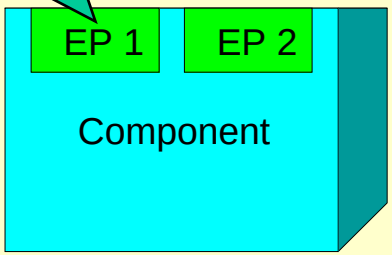
… communicate in a client/server relationship

… have EntryPoints that serve the information needs of specific clients

… provide access to synchronized data channel sets and to background information through their interfaces

# Generic Component Description

Interfaces
- ReqXXData
  - GetXXData
- GetXXExplanation
- GetXXConfig




## Interface Descriptions

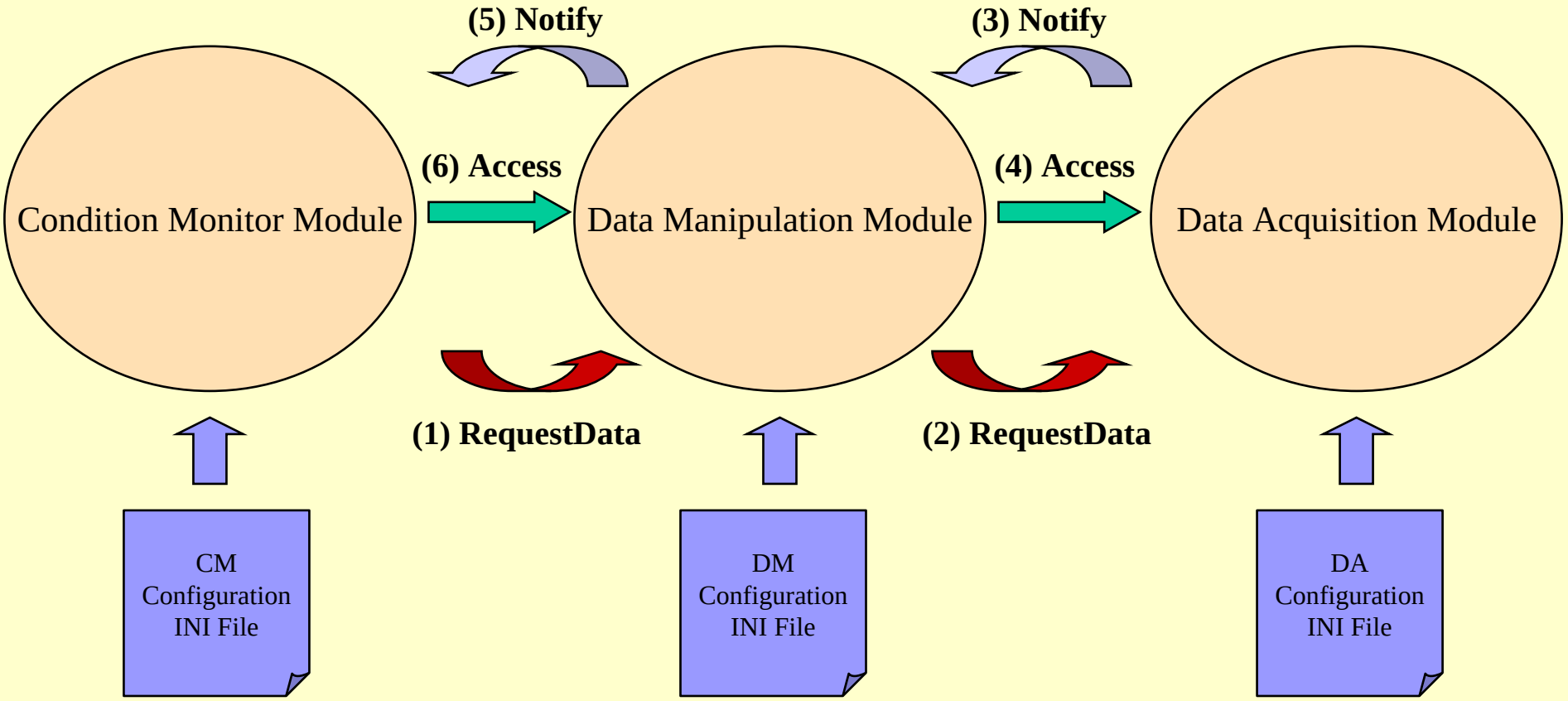
Request Data: prompts a measurement or calculation update

Get Data: returns dynamic measurement data or a calculated result to the client

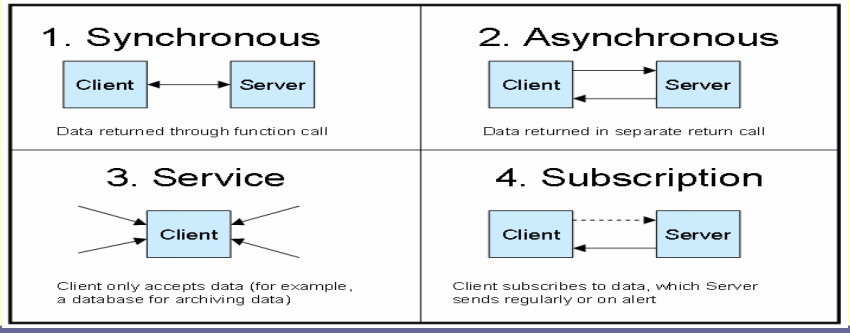
Get Explanation: returns a data structure which describes the input data and data transformation processes used in the calculation of the associated output data set

Get Config.: returns static information about the monitoring system and the monitored system configuration

# OSA-CBM Implementation



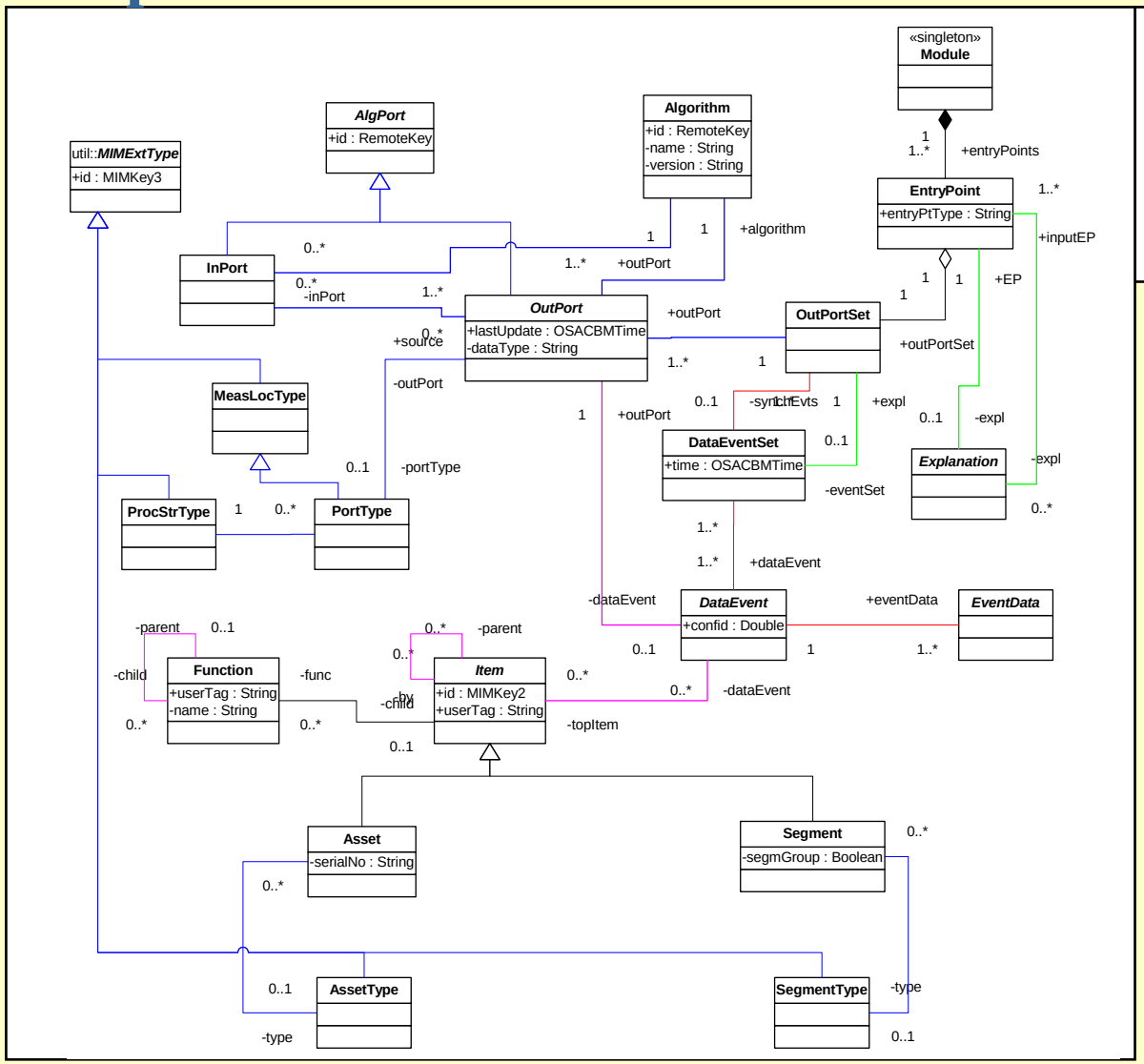

OSA-CBM data model supports: Synchronous, Asynchronous, Service, and Subscription transfers

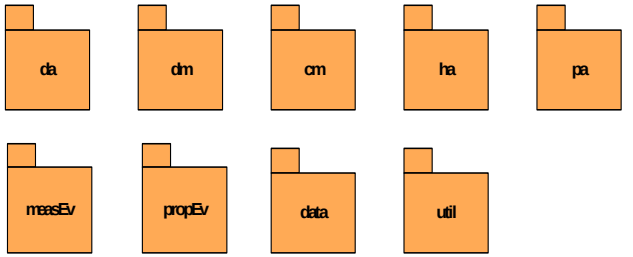


# OSA-CBM Models

# OSA-CBM UML Models

## Top Level UML Model




Organized in general, utility, and layer packages

RED - Associated classes are exposed by the data interface

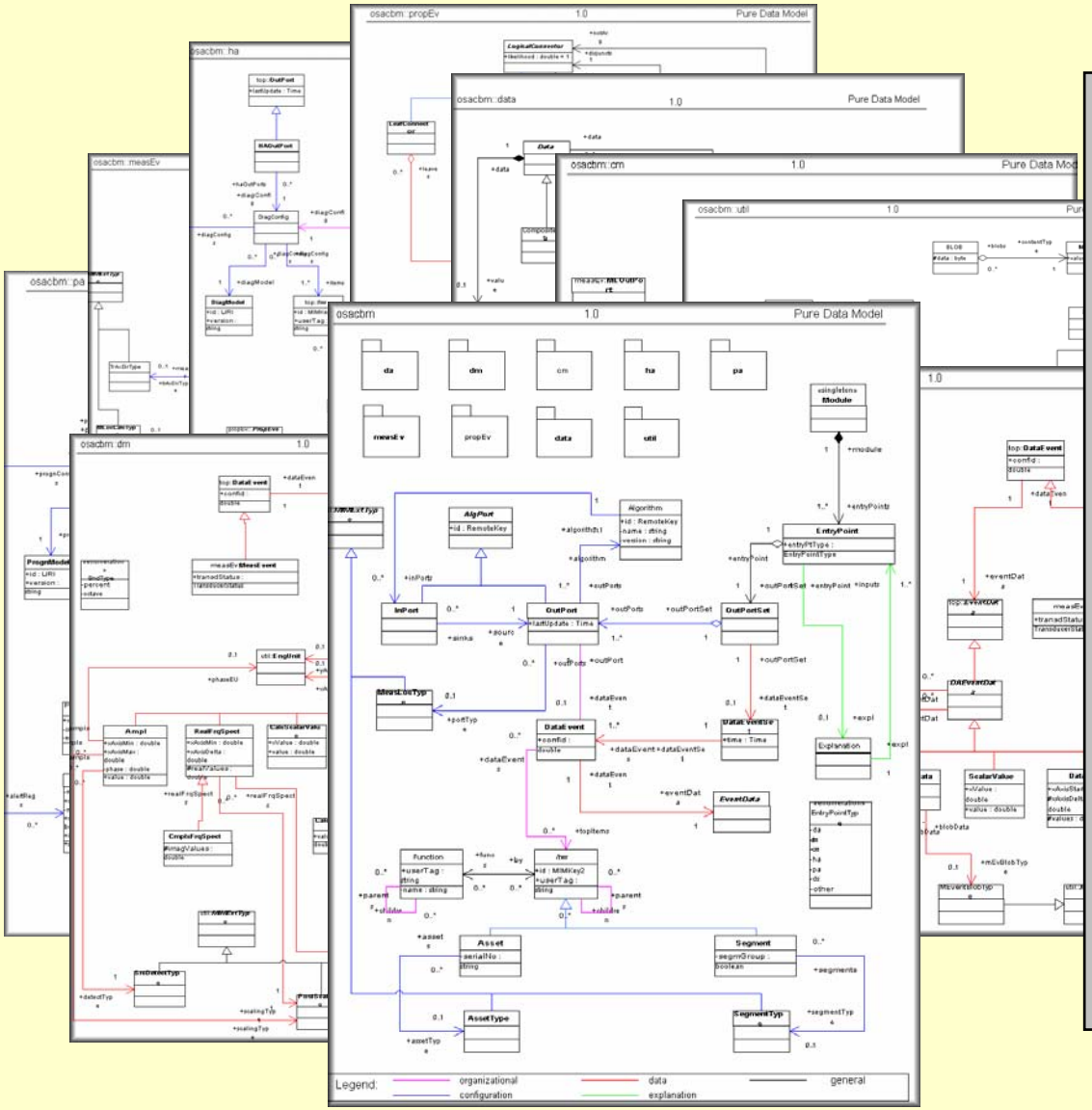
GREEN - Associated classes are exposed by the explanation interface

BLUE - Associated classes are exposed by the configuration interface

BLACK - Associations which are utilized by all interfaces (common concepts)

VIOLET - Represents a relationship between data elements which is not represented by the OSACBM interfaces.

# OSA-CBM UML Models

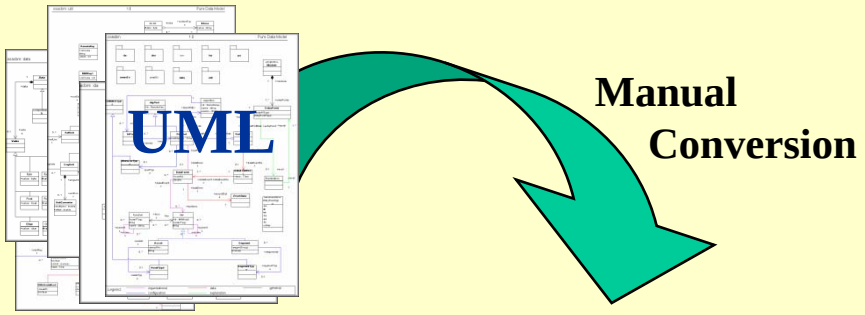



## UML Modules

*top* – Top Module, Common Classes

*da* – Data Acquisition Module
*dm* – Data Manipulation Module
*sd* – State Detection Module
*ha* – Health Assessment Module
*pa* – Prognostics Assessment Module
*ag* – Advisory Generation Module

*measEv* – Measurement Event Module
*propEv* – Proposed Event Module
*data* – Data Module
*util* – Utility Module



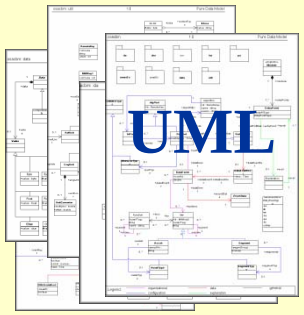

```
enum EntryPointType {
    da, dm, cm, ha, pa, ds, other
};

/* Entry point. */
interface EntryPoint {
    /* Identification of OSA/CBM layer. */
                EntryPointType    entryPtType;
    /* The set of ports served by this entry point. */
    [data] [config] [expl] [async]
                OutPortSet        outPortSet;
    /* Explanation of what sources were used by this module for its output. */
    [expl]
    [optional]    Explanation expl;
};
```

## Abstract Interface Document Language

Tags:

- Interface
  [data][config]
  [expl][org]
- Class Type
  [abstract]
- Multiplicity
  [sequence]
  [optional]

| UML syntax | AIDL Equivalent |
|---|---|
| 1 | no optional tag, no sequence |
| 0..1 | optional tag, no sequence |
| 1..* | no optional tag, sequence |
| 0..* | optional tag, sequence |




# AIDL provides the mechanism for porting to any Middleware Technologies

```
enum EntryPointType {
    da, dm, cm, ha, pa, ds, other
};

/* Entry point. */
interface EntryPoint {
    /* Identification of OSA/CBM layer. */
                EntryPointType    entryPtType;
    /* The set of ports served by this entry point. */
    [data] [config] [expl] [async]
                OutPortSet        outPortSet;
    /* Explanation of what sources were used by this module for its output. */
    [expl]
    [optional]    Explanation expl;
};
```

AIDL

## Middleware Technologies

COM/DCOM

CORBA

XSD using WebServices

# Middleware Technologies

# Middleware Technology

Applications

Data Acquisition | Data Manipulation | Condition Monitor | Health Assessment | Prognostic Assessment | Decision Support

Applications

Middleware

**Domain-Specific Services**

**Common Services**

**Distribution Middleware**

**Infrastructure Middleware**

Operating System

**COM/DCOM, CORBA, SOAP, JAVA-RMI,…**

**Benefits**

- Hide underlying complexity,
- Share common services,
- Synchronize program execution,
- Balance system load,
- Real-time solutions,
- Cross-language, and
- Cross-platform support

Middleware

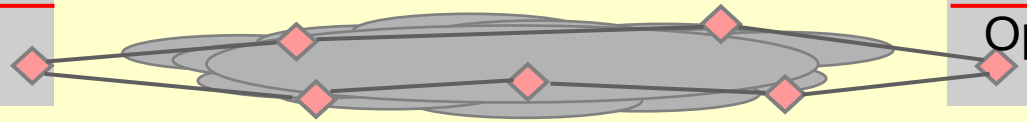
**Domain-Specific Services**

**Common Services**

**Distribution Middleware**

**Infrastructure Middleware**

Operating System



Endsystem | Wired/Wireless Networks | Endsystem

- ***Distributed Computing 'Glue'***
- ***Software layer between application and communication networking layer***

# CORBA

- *Common Object Request Broker Architecture (CORBA) is the Object Management Group's answer to the need for interoperability among the rapidly proliferating number of hardware and software products available today.*
- *Introduced CORBA 1.1 in 1991*
- *The OMG's CORBA is a specification, not a software implementation of a middleware.*
- *Real-time version available*

# COM/DCOM

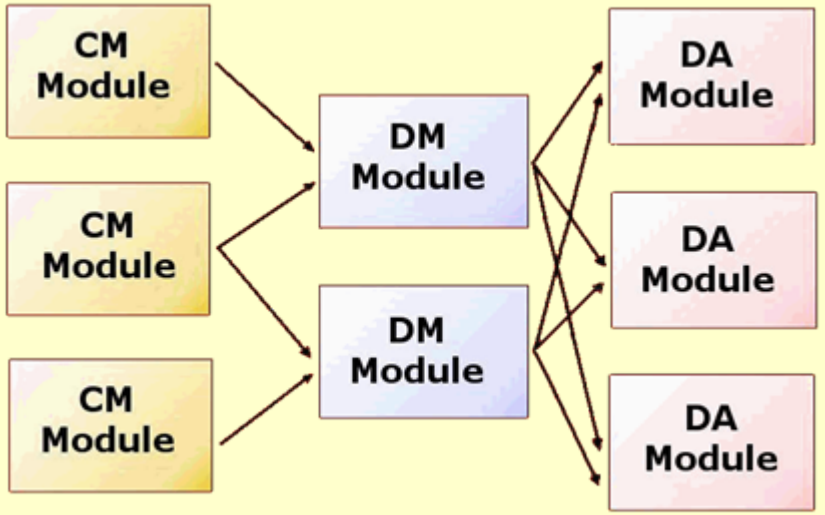
- *Component Object Model (COM)/ Distributed Component Object Model (DCOM) is a Microsoft Technology to replace Object Linking and Embedding (OLE) and Dynamic Data Exchange (DDE).*

- *DCOM emerged to address COM's shortcomings in supporting remote components. DCOM is an extension to COM that allows networked interaction between two programs even if they are written in different programming languages.*

# COM/DCOM

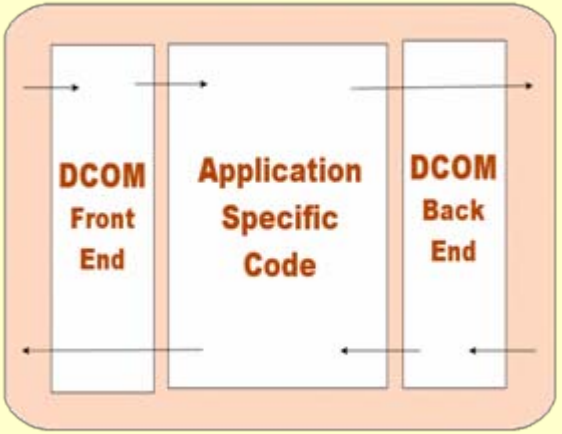
## Benefits of COM/DCOM

- Increased Security
- Remote Module Instantiation
- Standardized Information Exchange
- Client Control (lifetime, termination, Reference,...)
- Internal Pinging Mechanism




## COM/DCOM Interfaces:

- requestOutPortSet(id, sink)
- notifyOutPortSet(id, data)
- getTaskStatus(Seconds)
- getExpl(expl)




## Use of ID's:

- Allow flexible/tailored design
- Control how data is updated
- Control how data is processed

## Sink and Data Pointers:

- Allows asynchronous communication between modules
- Contains relevant information about the data: Time, Eng. Units, Confid., asset and organizational info…

# COM/DCOM

## COM/DCOM Modules Developed:

- DA
- DM
- CM
- HA
- PA
- PL

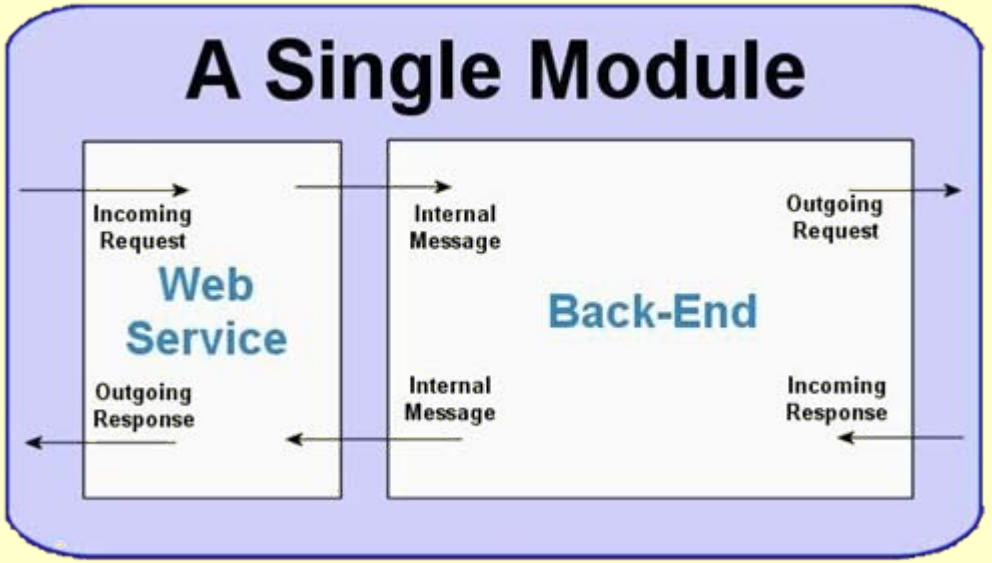


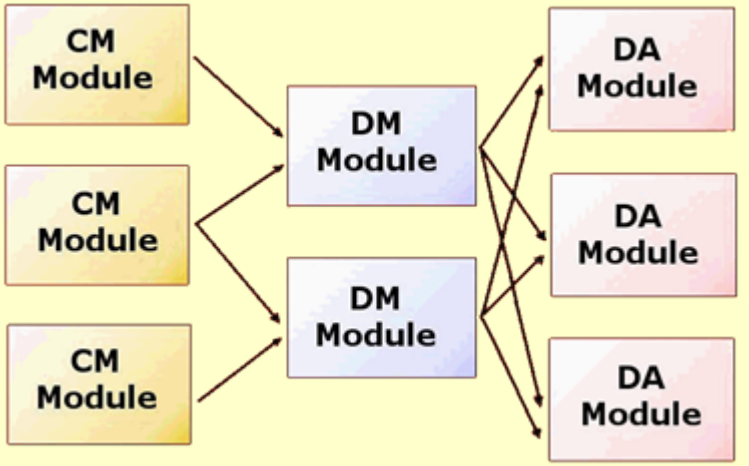

Gearbox Prognostic Application

# XML over HTTP

- *XML (eXtensible Markup Language) is not a middleware technology, it is simply a user-definable data format*
- *However, XML may be used with any networking technology for data transfer.*
- *Tag structure format similar to HTML*
- *XML started in 1996 and has been a World Wide Web Consortium (W3C) recommendation since February 1998*
- *Use SOAP or Web Services to transfer data*

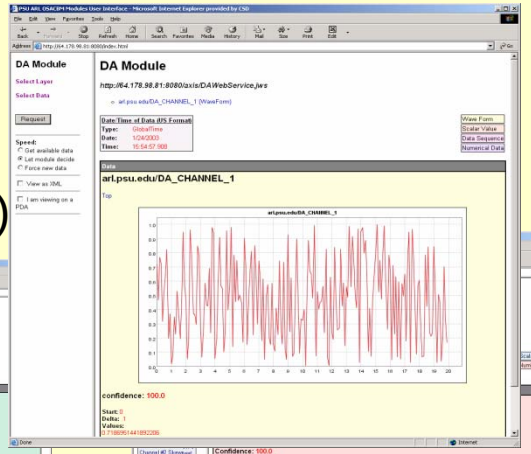
# XML over SOAP

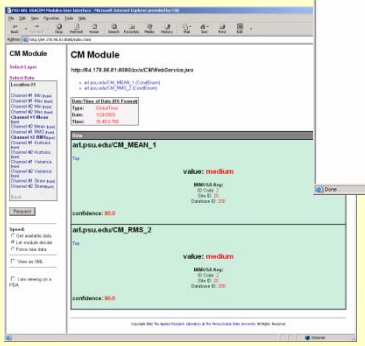



## SOAP Interfaces:

- requestData(request)
- forceNewData(request)
- getAvailableData(request)





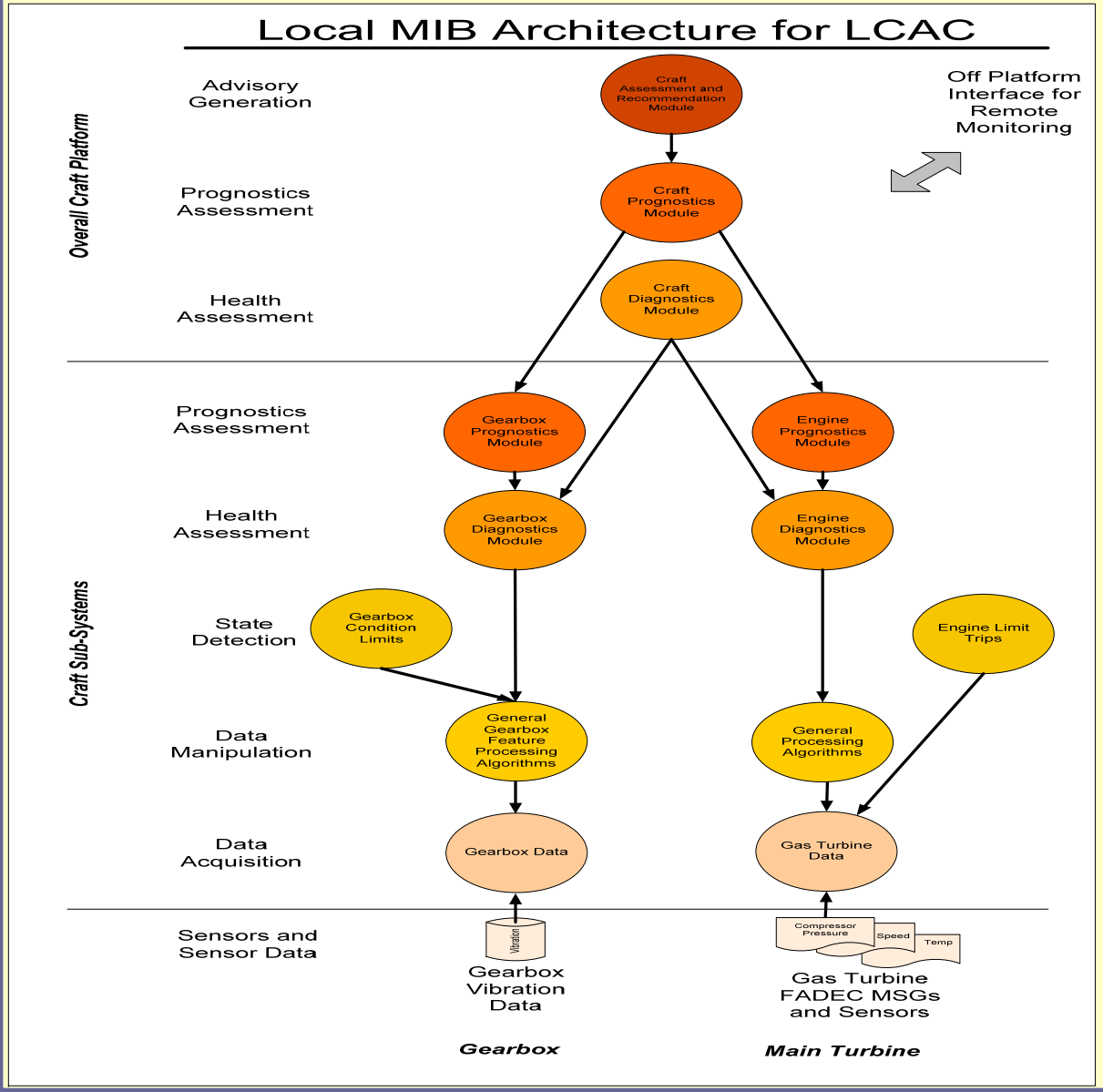


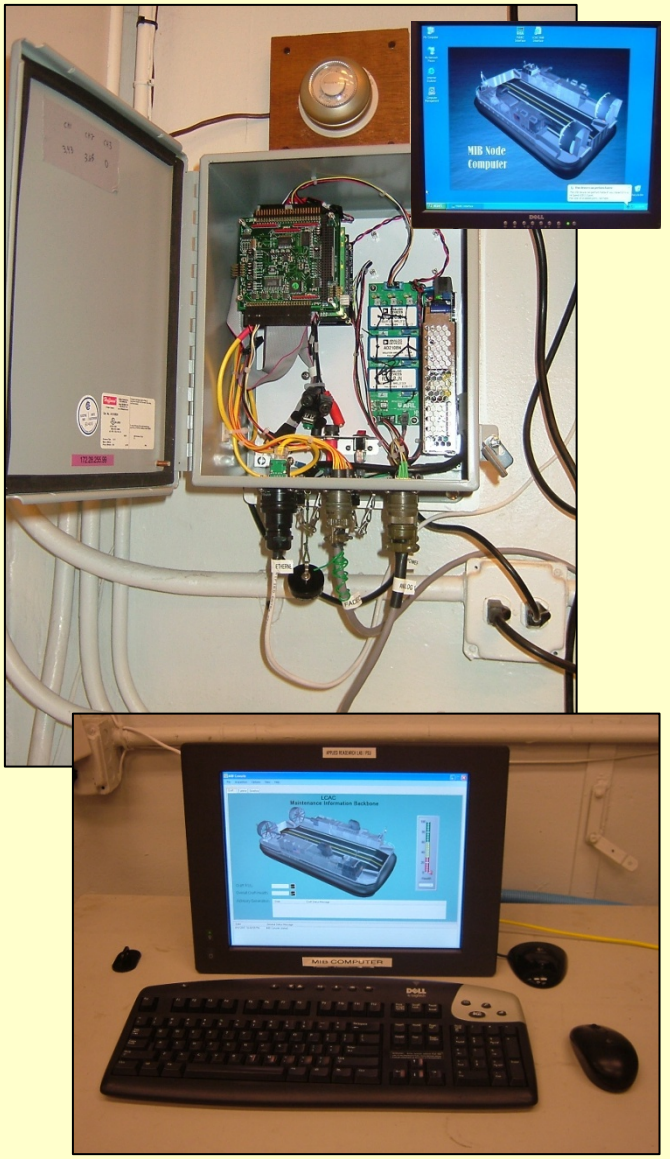
## Benefits of using SOAP

- Platform Independence
- Language Independence
- Reduced Network Restrictions (uses port 80)
- Standardized Information Exchange

# OSA-CBM Design Flexibility

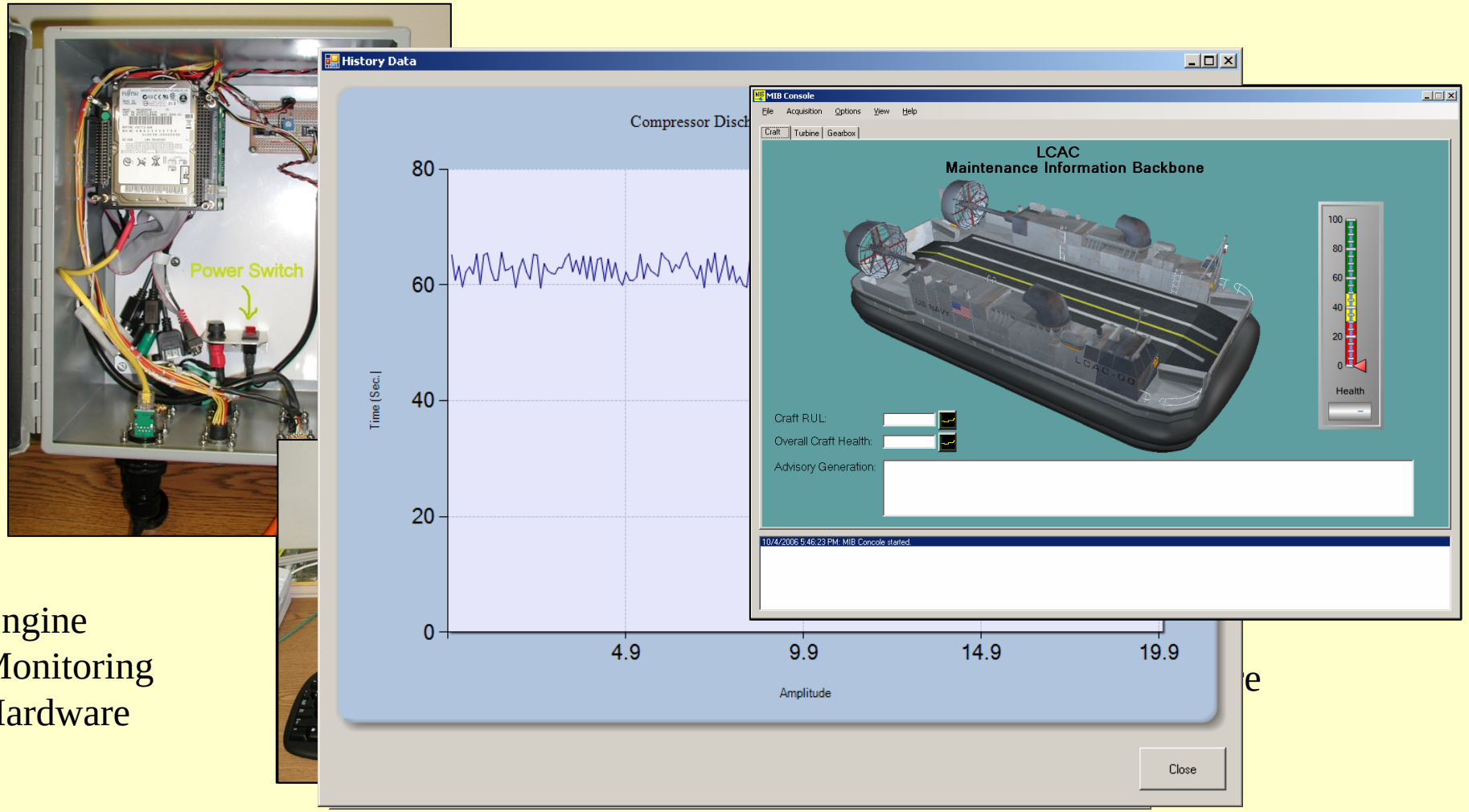
# OSA-CBM Design Examples



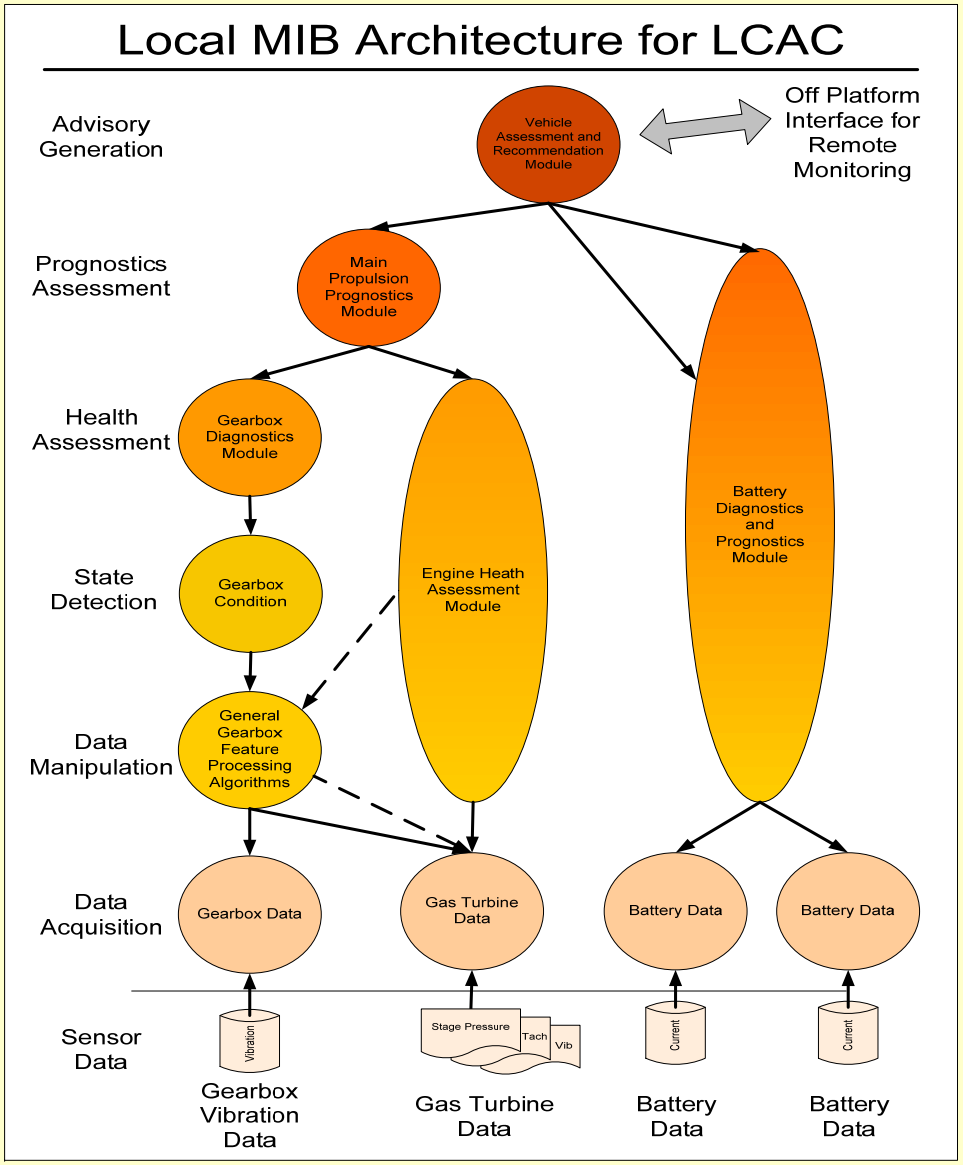
# Maintenance Information Backbone (MIB) Example

## MIB Demonstration System




Engine Monitoring Hardware

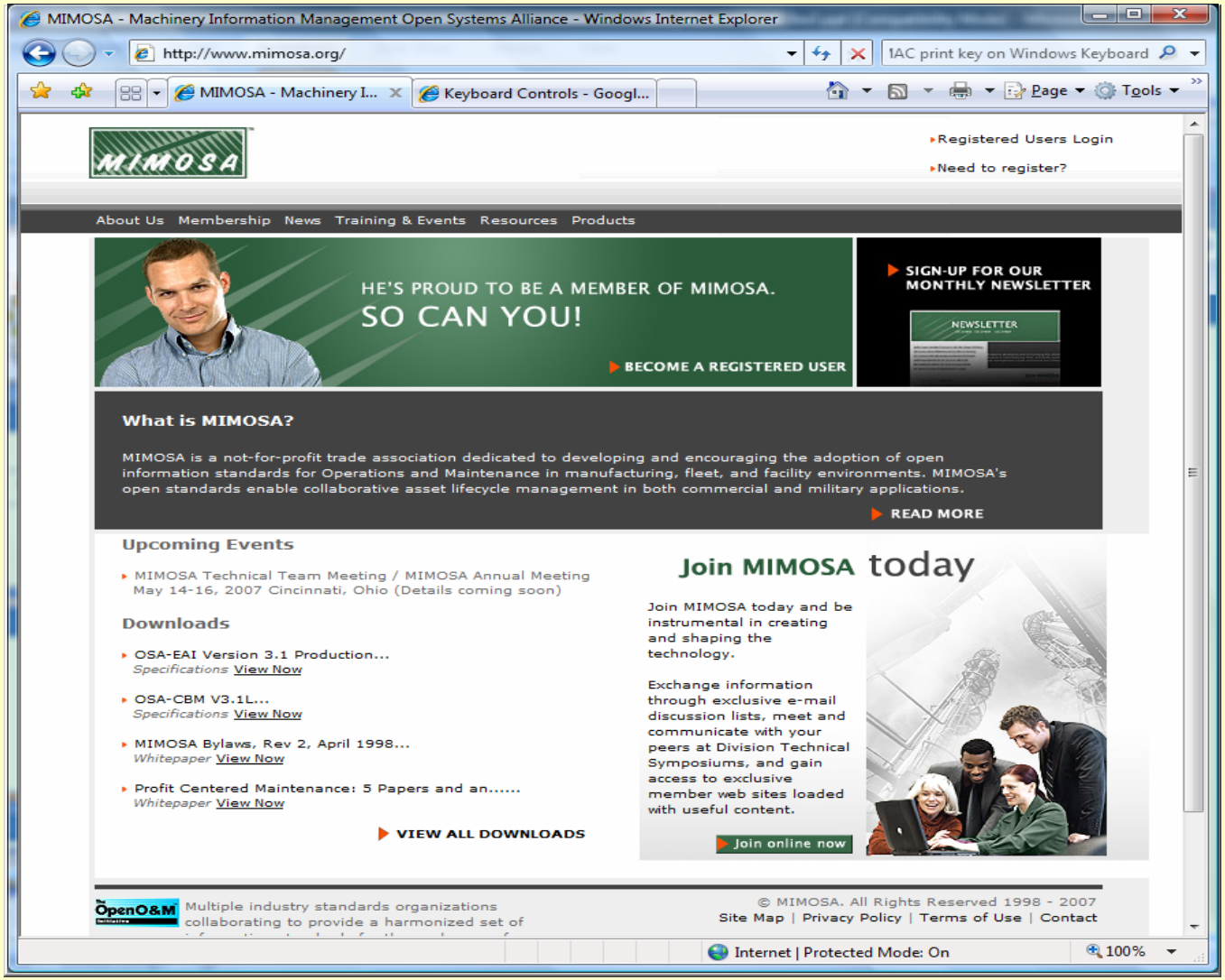
# OSA-CBM Design Examples



# MIMOSA (OSA-CBM) Website

# OSA-CBM Website

**WWW.MIMOSA.ORG**




- Documents and Presentations
- OSA-CBM Models
  - UML
  - AIDL
  - COM/DCOM IDL
  - CORBA IDL
  - XML Schema
- Training Material
- Developer's Tools and Example Code
- Real-time Demo
- Glossary
- Useful References

# Acknowledgements


- This work was supported by:
  - The Office of Naval Research under research grant number: N00014-00-1-0155 OSA-CBM Boeing DUST


- OSA/CBM team members and contributors include: Boeing, Caterpillar, Rockwell Automation, Penn State, Newport News, Oceana Sensor Technologies, Rockwell Scientific, MIMOSA…

PENNSTATE ARL 1855

Applied Research Laboratory
The Pennsylvania State University

# OSA-CBM: An Open System Architecture for Machinery Monitoring

*Structural Health Monitoring COE Inaugural Meeting*
*– April 13th 2007 -*


***Contact at ARL***

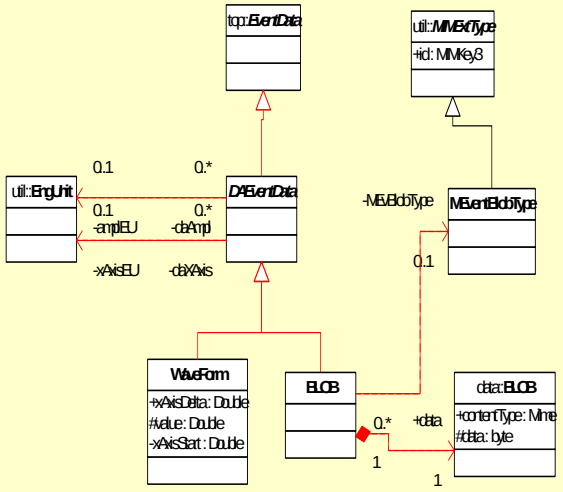
***Mitchell Lebold***
*814.865.8958 (voice)*
*814.863.0673 (Fax)*
*Lebold@psu.edu*
*http://www.arl.psu.edu*

# Backup Slides

# MIMOSA and OSA-CBM

**MIMOSA** is a standard for data exchange between Asset Management systems

**OSA-CBM** is a specification for transactions between components within a Condition-Based Maintenance system




- The core of the OSA-CBM standard is the Object Oriented data model, defined using UML (Unified Modeling Language) syntax
- The OSA-CBM UML data model is a mapping of key concepts from the MIMOSA CRIS with extensions for diagnostics, prognostics and data transactions