



PKDD/ECML 2007

Knowledge Discovery Standards in Ubiquitous Environments

Marko Grobelnik, Michael May, Dennis Wegener



Tutorial's Objectives



- Successful real-world data mining applications often require to embed the data-mining engine into an environment
- These environments become more and more distributed and/or ubiquitous
- To build ubiquitous knowledge discovery systems, we have to master a large and complex set of existing technologies
- To be able to build on the work of others and to build reusable systems, we have to be aware of existing standards

Tutorial's Objectives



- This tutorial provides:
 - What are standards for ubiquitous KD ?
 - Overview of existing KD standards
 - Motivation for using standards
 - How do these standards relate to each other?
 - Other relevant standards for ubiquitous KD

Tutorial Outline

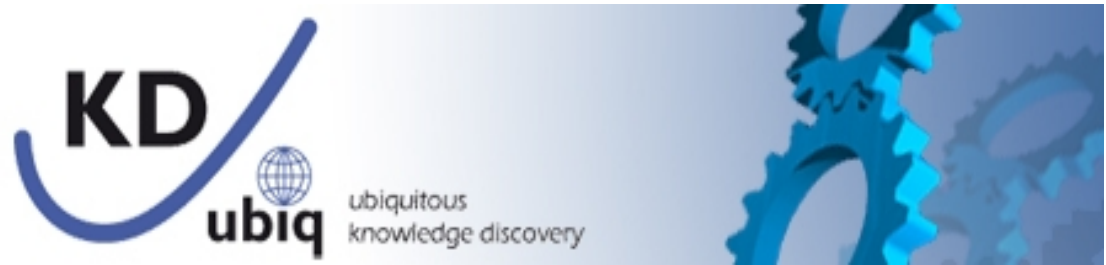


- Introduction
- CRISP-DM
- Microsoft OLE-DB fro DM
- XMLA – XML for Analysis
- SQL/MM Part 6: SQL interfaces for Data Mining
- Java Data Mining API
- PMML - Predictive Model Mark-up Language
- Grid standards relevant for KD
- Web Services relevant for KD

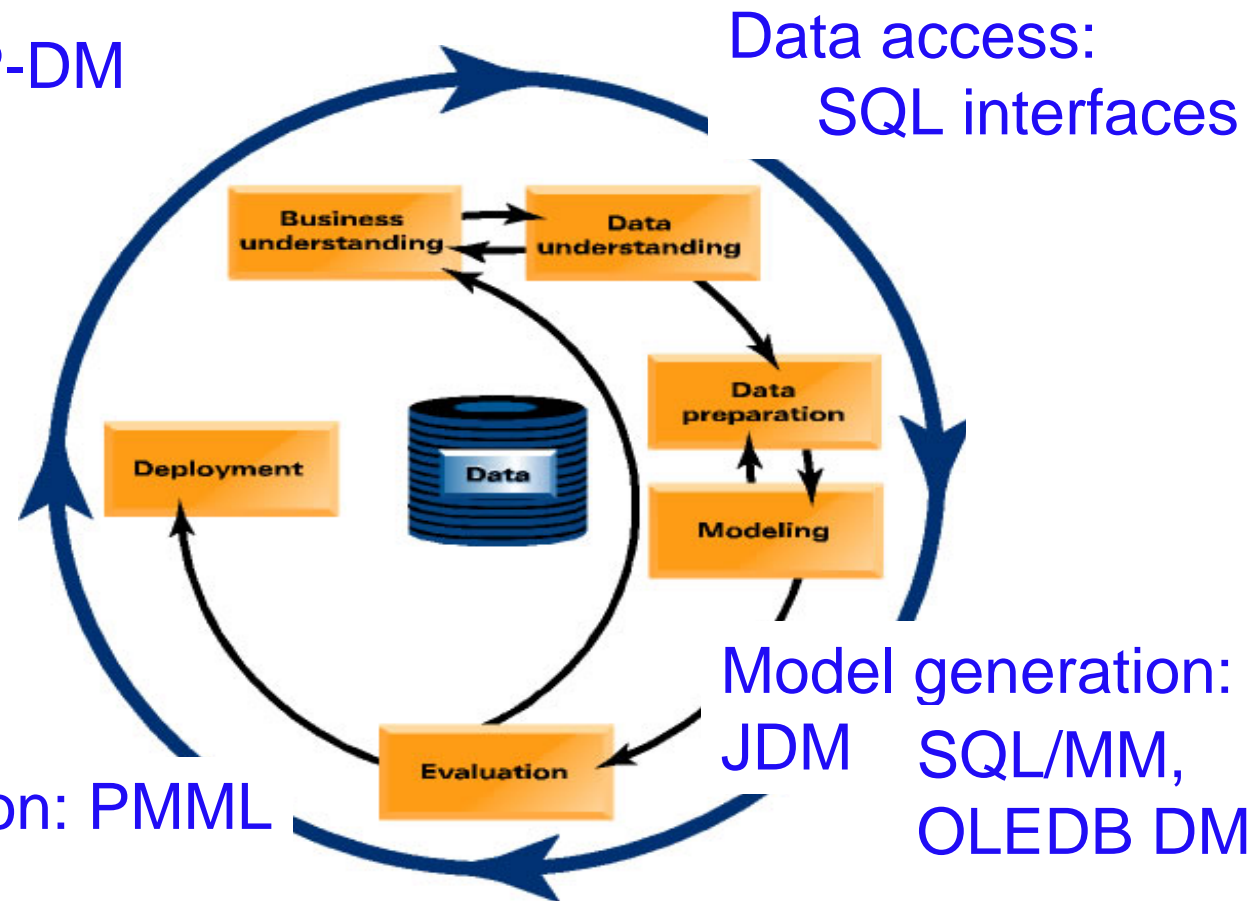
Introduction to Existing Knowledge Discovery Standards



The Knowledge Discovery Process with Classical Standards



Global view: CRISP-DM



Data access:
SQL interfaces

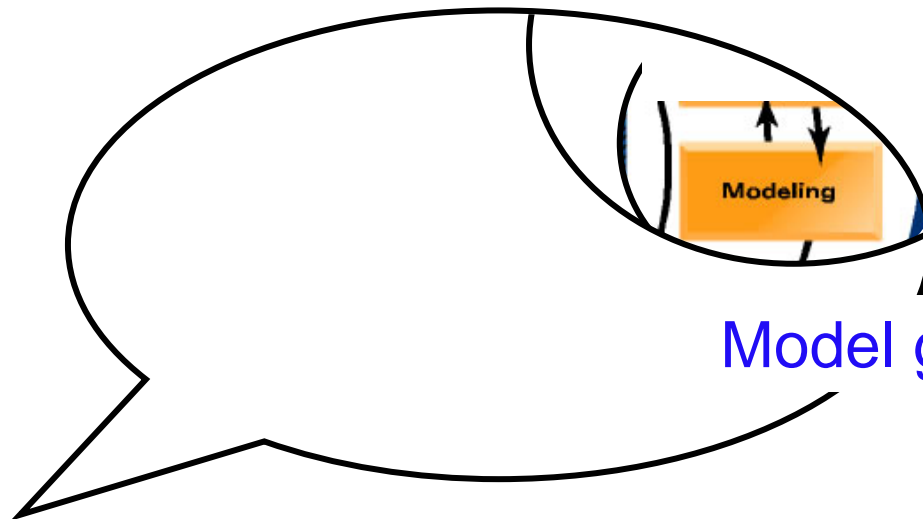
Model generation:
JDM SQL/MM,
OLEDB DM

Model representation: PMML

The Knowledge Discovery Process



Data access:
SQL interfaces



Model generation: JDM
SQL/MM,
OLEDB DM

Model representation: PMML





CRISP-DM

A Standard Process Model for Data Mining

<http://www.crisp-dm.org/>



What is CRISP-DM?



- **Cross-Industry Standard Process** for Data Mining
- **Aim:**
 - To develop an industry, tool and application neutral process for conducting Knowledge Discovery
 - Define tasks, outputs from these tasks, terminology and mining problem type characterization
- Founding **Consortium Members:** DaimlerChrysler, SPSS and NCR
- **CRISP-DM Special Interest Group** > 300 members
 - Management Consultants
 - Data Warehousing and Data Mining Practitioners
- **CRISP 1.0** released in 1999
- **CRISP 2.0 SIG** has been formed (Sept 2006, Chicago), SIG meeting in Jan. 2007

Four Levels of Abstraction



- **Phases**
 - Example: Data Preparation
- **Generic Tasks**
 - A stable, general and complete set of tasks
 - Example: Data Cleaning
- **Specialized Task**
 - How is the generic task carried out
 - Example: Missing Value Handling
- **Process Instance**
 - Example: The mean value for numeric attributes and the most frequent for categorical attributes was used

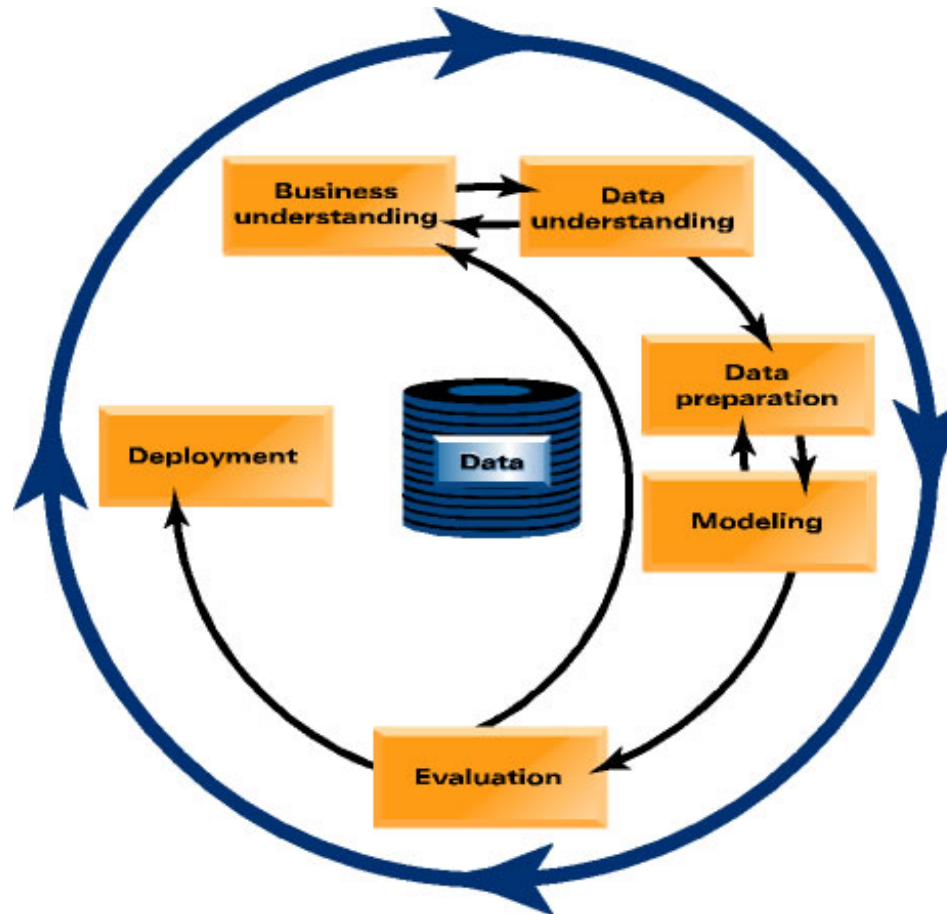
Data Mining Context



- In data mining the context is defined by four dimensions
 - Application domain: Medical Prognosis
 - Data Mining Problem Type: Regression
 - Technical Aspect: Censored Observations
 - Tools and Techniques: Cox's Regression, CIL's GENNA
- The context of the data mining task at hand is the starting point for mapping the generic tasks to specific tasks required in this instance



Phases of CRISP-DM



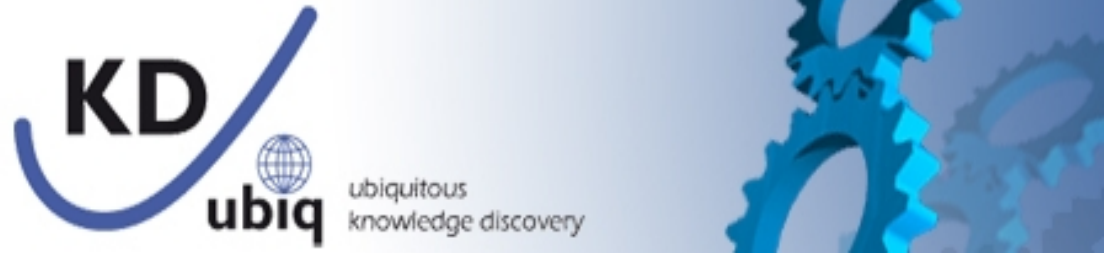
- Not linear, repeatedly backtracking

Business Understanding Phase : Part I



- Understand the **business objectives**
 - What is the status quo?
 - Understand business processes
 - Associated costs/pain
 - Define the success criteria
 - Develop a glossary of terms: speak the language
 - Cost/Benefit Analysis
- Current Systems Assessment
 - Identify the key actors
 - Minimum: The Sponsor and the Key User
 - What forms should the output take?
 - Integration of output with existing technology landscape
 - Understand market norms and standards

Business Understanding Phase : Part II



- **Task Decomposition**
 - Break down the objective into sub-tasks
 - Map sub-tasks to data mining problem definitions
- **Identify Constraints**
 - Resources
 - Law e.g. Data Protection
- **Build a project plan**
 - List assumptions and risk (technical/financial/business/ organisational) factors

Data Understanding Phase : Part I



- **Collect Data**
 - What are the **data sources**?
 - Internal and External Sources (e.g. Axiom, Experian)
 - Document reasons for inclusion/exclusions
 - Depend on a domain expert
 - Accessibility issues
 - Legal and technical
 - Are there issues regarding data distribution across different databases/legacy systems
 - Where are the disconnects?

Data Understanding Phase : Part II



- **Data Description**
 - Document data quality issues
 - requirements for data preparation
 - Compute basic statistics
- **Data Exploration**
 - Simple univariate data plots/distributions
 - Investigate attribute interactions
 - Data Quality Issues
 - Missing Values
 - Understand its source: Missing vs Null values
 - Strange Distributions

Data Preparation Phase : Part I



- **Integrate Data**
 - Joining multiple data tables
 - Summarisation/aggregation of data
- **Select Data**
 - Attribute subset selection
 - Rationale for Inclusion/Exclusion
 - Data sampling
 - Training/Validation and Test sets

Data Preparation Phase : Part II



- Data Transformation
 - Using functions such as log
 - Factor/Principal Components analysis
 - Normalization/Discretisation/Binarisation
- Clean Data
 - Handling missing values/Outliers
- Data Construction
 - Derived Attributes

The Modelling Phase

Part I



- **Select of the appropriate modelling technique**
 - Data pre-processing implications
 - Attribute independence
 - Data types/Normalisation/Distributions
 - Dependent on
 - Data mining problem type
 - Output requirements
- **Develop a testing regime**
 - Sampling
 - Verify samples have similar characteristics and are representative of the population

The Modelling Phase

Part II



- **Build Model**
 - Choose initial parameter settings
 - Study model behaviour
 - Sensitivity analysis
- **Assess the model**
 - Beware of over-fitting
 - Investigate the error distribution
 - Identify segments of the state space where the model is less effective
 - Iteratively adjust parameter settings
 - Document reasons of these changes

The Evaluation Phase



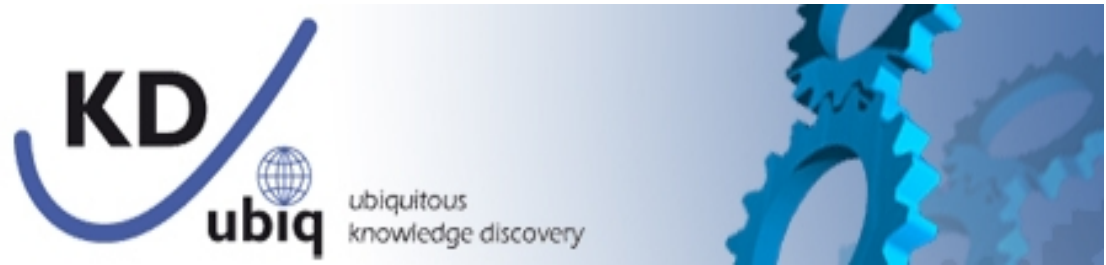
- **Validate Model**
 - Human evaluation of results by domain experts
 - Evaluate usefulness of results from business perspective
 - Define control groups
 - Calculate lift curves
 - Expected Return on Investment
- **Review Process**
- **Determine next steps**
 - Potential for deployment
 - Deployment architecture
 - Metrics for success of deployment

The Deployment Phase



- Knowledge Deployment is specific to objectives
 - Knowledge Presentation
 - Deployment within Scoring Engines and Integration with the current IT infrastructure
 - Automated pre-processing of live data feeds
 - XML interfaces to 3rd party tools
 - Generation of a report
 - Online/Offline
 - Monitoring and evaluation of effectiveness
- Process deployment/production
- Produce final project report
 - Document everything along the way

CRISP-DM and Ubiq. KD



- For ubiq. KD Systems, the deployment is much more complex, since those systems are often deployed to non-standard IT-infrastructures (sensor networks, embedded devices, Grids...) and for real-time usage scenarios
- CRISP-DM still highly useful for describing the model building process
- CRISP 2.0 SIG announces to address some of these more complex scenarios

Microsoft OLE DB for DM

Extension of Microsoft Analysis
Services for Data Mining

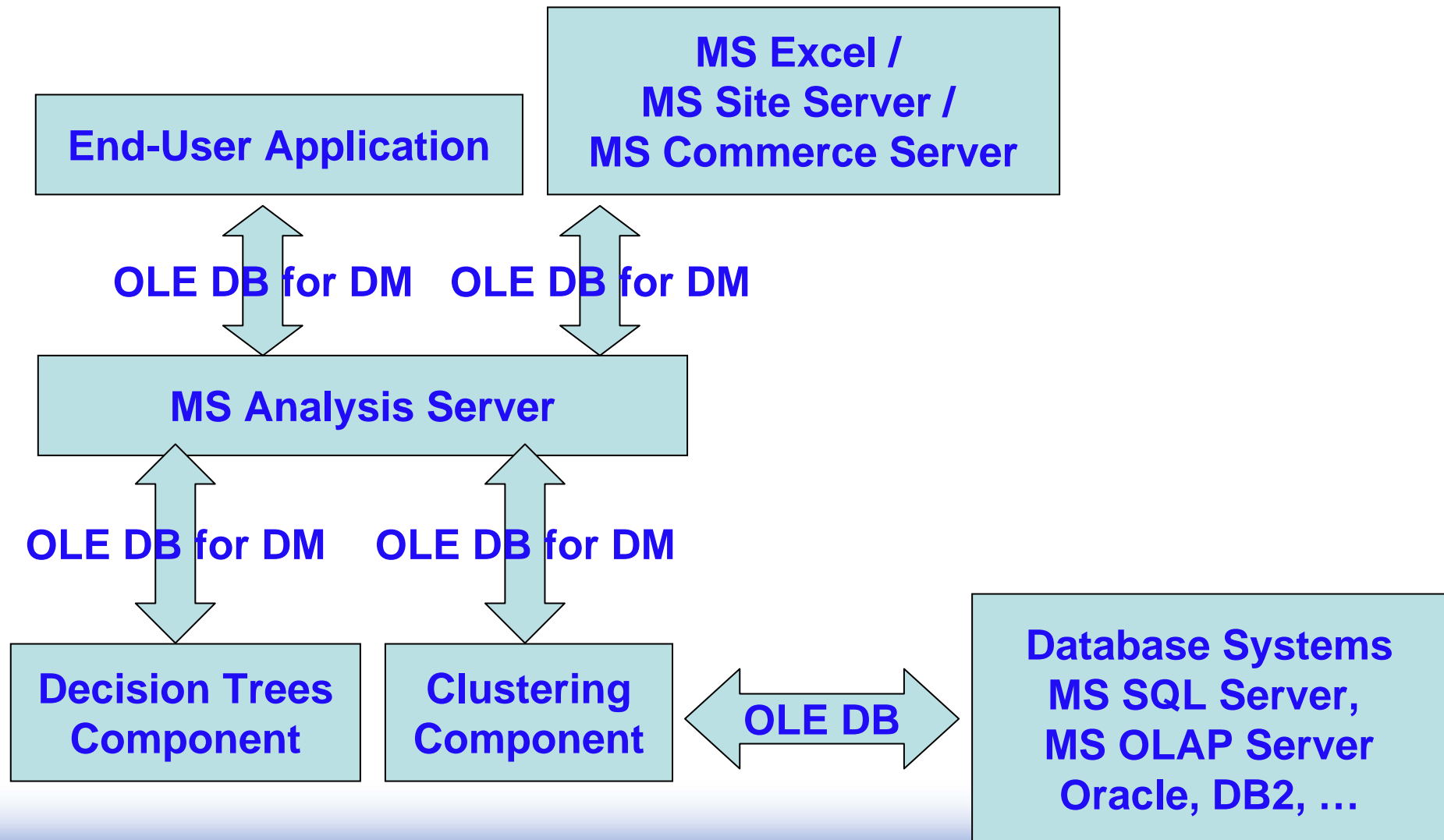


What is OLE DB for Data-Mining?



- “OLE DB for DM” is Microsoft’s Extension of Analysis Server product for covering DM functionality
 - It is closely connected to MS OLAP Server
 - Works within SQL Server database suite (SQL Server 2005)
- It defines DM at several levels:
 - Extensions of SQL language for describing DM tasks
 - API in the form of COM interface for:
 - (1) Programming DM clients within applications
 - (2) Programming DM providers (server side components) for including new DM algorithms
 - Uses PMML for model description

Architecture of a Solution Using OLE DB for DM Technology



What are key DM tasks?



- Key DM tasks covered by OLD DB for DM are:
 - Predictive Modeling (Classification)
 - Segmentation (Clustering)
 - Association (Data Summarization)
 - Sequence and Deviation Analysis
 - Dependency Modeling

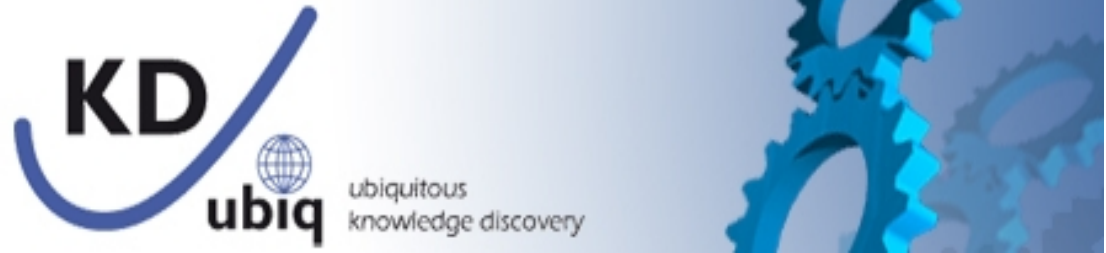
Defining a domain – Creating Mining Model Object



Using an OLE DB command object, the client executes a CREATE statement that is similar to a CREATE TABLE statement:

```
CREATE MINING MODEL [Age Prediction](  
    [Customer ID] LONG KEY,  
    [Gender] TEXT DISCRETE,  
    [Age] DOUBLE DISCRETIZED() PREDICT,  
    [Product Purchases] TABLE (  
        [Product Name] TEXT KEY,  
        [Quantity] DOUBLE NORMAL CONTINUOUS,  
        [Product Type] TEXT DISCRETE RELATED TO [Product  
    Name]  
    )  
)  
USING [Decision Trees]
```

Inserting Training Data into Model



In a manner similar to populating an ordinary table, the client uses a form of the INSERT INTO statement.

Note the use of the SHAPE statement to create the nested table.

```
INSERT INTO [Age Prediction](
    [Customer ID], [Gender], [Age],
    [Product Purchases](SKIP, [Product Name],
    [Quantity], [Product Type])
)
SHAPE {
    SELECT [Customer ID], [Gender], [Age] FROM
    Customers ORDER BY [Customer ID]
}
APPEND (
    {SELECT [CustID], [Product Name], [Quantity],
    [Product Type] FROM Sales ORDER BY [CustID]}
    RELATE [Customer ID] To [CustID])
AS [Product Purchases]
```

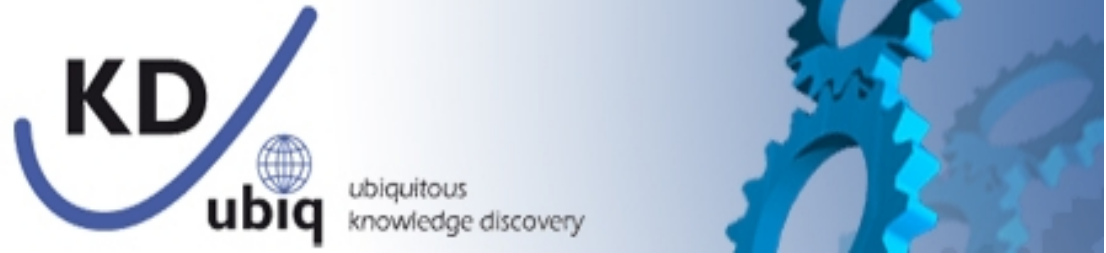
Using Models to make Predictions



Predictions are made with a SELECT statement that joins the model's set of all possible cases with another set of actual cases.

```
SELECT t.[Customer ID], [Age Prediction].[Age]
FROM [Age Prediction]
PREDICTION JOIN (
    SHAPE {
        SELECT [Customer ID], [Gender], FROM Customers ORDER BY
        [Customer ID]}
APPEND (
    {SELECT [CustID], [Product Name], [Quantity] FROM Sales
    ORDER BY [CustID]}
    RELATE [Customer ID] To [CustID]
)
    AS [Product Purchases]
) as t
ON [Age Prediction] .Gender = t.Gender and
    [Age Prediction] .[Product Purchases].[Product Name] =
    t.[Product Purchases].[Product Name] and
    [Age Prediction] .[Product Purchases].[Quantity] =
    t.[Product Purchases].[Quantity]
```

Association Rules



- The following statement creates a data mining model to find out those products which sell together based on an association algorithm. The model is interested only in rules with at least five items:

```
Create Mining Model MyAssociationModel (  
    Transaction_id long key,  
    [Product purchases] table predict (  
        [Product Name] text key          ) )  
Using [My Association Algorithm] (Minimum_size = 5)
```

- Training an association model is exactly the same as training a tree model or a clustering model.
- To get all the association rules discovered by the algorithm, run the following statement:

```
Select * from MyAssociationModel.content
```

Regression Analysis



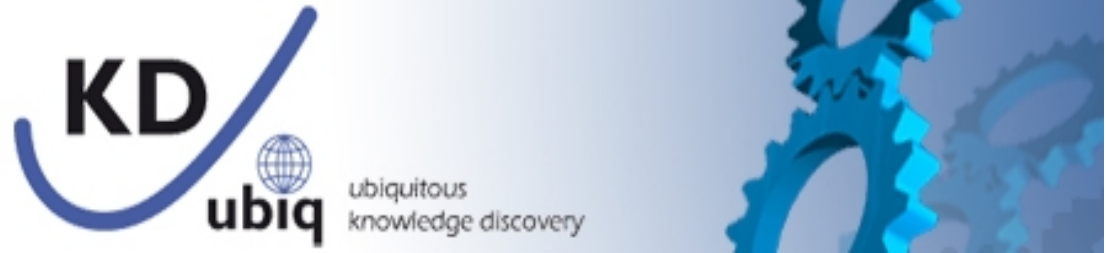
- By using a regression algorithm, the following mining model predicts loan risk level based on age, income, homeowner, and marital status:

```
Create Mining Model MyRegressionModel (  
  Customer_id long key,  
  Age long continuous,  
  Homeowner boolean discrete,  
  Marital_status Boolean discrete,  
  Loan_risk_LEVELcontinuous predict  
)  
Using [My Regression Algorithm]
```

- The following statement returns all the coefficients of the regression:

```
Select * from MyRegressionModel.content
```

Visual Basic example using the OLE DB for DM Clustering component



```
(1) Dim ClusterConnection As New ADODB.Connection
(2) ClusterConnection.Provider = "MSDMine"
(3) DMMName = "[CollPlanDMM]"
(4) DataFileName = ".\CollegePlan.mdb"
(5) ClusterConnection.ConnectionString = "location=localhost;"
    & _ "initial catalog=[FoodMart 2000];"
(6) ClusterConnection.Open
(7) ClusterConnection.Execute "CREATE MINING MODEL
    [ClusterModel]"
    & _ "([Student Id] LONG KEY, [College Plans] TEXT DISCRETE
    PREDICT,"
    & "[Gender] TEXT DISCRETE PREDICT, [Iq] LONG CONTINUOUS
    PREDICT,"
    & _ "[Parent Encouragement] TEXT DISCRETE PREDICT,
    [Parent Income]
    LONG CONTINUOUS PREDICT)"
    & _ "USING Microsoft_Clustering"
(8) ...
```



SQL/MM Part 6: Data Mining

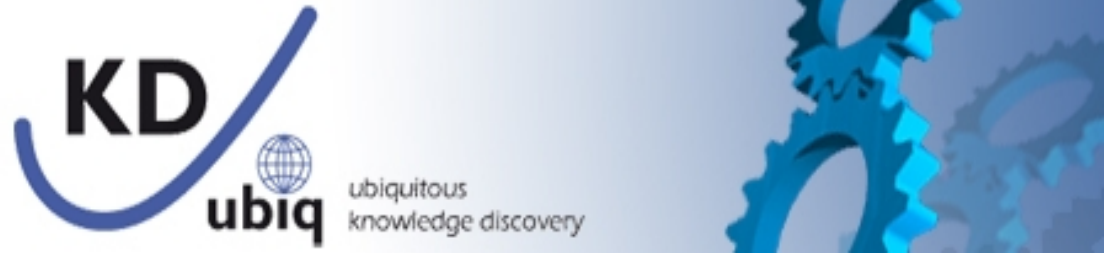


SQL/MM: Data Mining



- Part of ISO/IEC 13249 (SQL/MM) SQL Multimedia and Applications
- Database standard for data mining
- Allows Data Mining Queries as part of SQL
- Somewhat similar in scope to OLE DB
- Supports a data warehouse scenario
- Supports all three phases of mining:
 - Training
 - Test
 - Application

SQL/MM: Data Mining



- Describes meta data for mining, input data, results
- Supports several data mining techniques:
 - Association Rules
 - Clustering
 - Classification
 - Regression
- Introduces a set of user-defined types

Example: Part I



Example from: SQL/MM DataMining ISO/IEC WD 13249-6 Working Draft

- 1) Create a DM_MiningData value using the static method DM_defMiningData
- 2) Create a DM_MiningSchema value using the method DM_genMiningSchema of the DM_MiningData value
- 3) Create a DM_ClasSettings value using the default constructor and assign the DM_MiningSchema value as the schema to use
- 4) Declare the column named 'r' as the predicted field using the DM_clasSetTarget method
- 5) Create a DM_ClasTask value using the DM_defClasTask method
- 6) Store the newly created DM_ClasTask value in table MT.

All the steps described above can be expressed as a single SQL statement

Example: Part II



Example from: SQL/MM DataMining ISO/IEC WD 13249-6 Working Draft

```
WITH MyData AS (  
  DM_MiningData::DM_defMiningData('CT')  
)  
INSERT INTO MT (ID, TASK)  
VALUES (  
  1,  
  DM_ClasTask::DM_defClasTask(  
    MyData, NULL,  
    (  
      (new DM_ClasSettings())  
      .DM_clasUseSchema(MyData.DM_genMiningSchema())  
    ).DM_clasSetTarget('r')  
  )  
)
```

Example: Part III



Example from: SQL/MM DataMining ISO/IEC WD 13249-6 Working Draft

Now that the DM_ClasTask value is generated and stored in the MT table, the classification training can be initiated and the classification model is computed. Since the model shall be used in later application and test runs, it is stored in a table MM having two columns ID of type integer and MODEL of type

DM_ClasModel:

```
INSERT INTO MM (ID, MODEL)
VALUES (
1,
MyTask.DM_buildClasModel()
)
```

XMLA - XML for Analysis

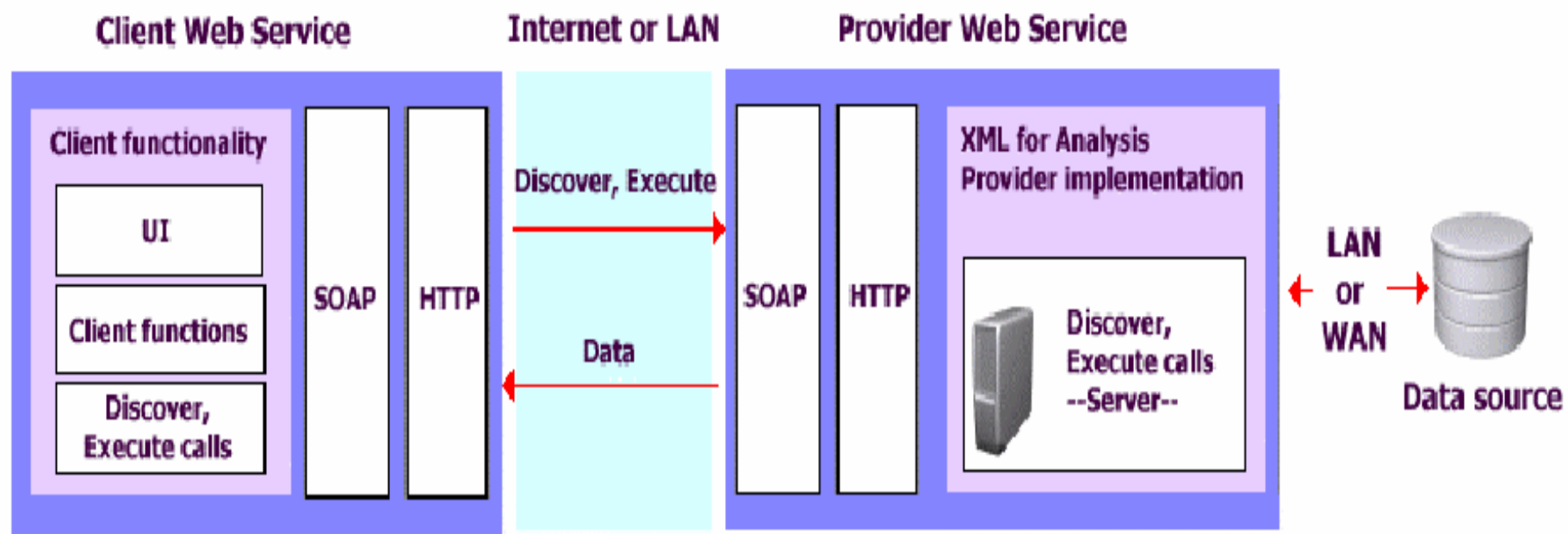
<http://www.xmla.org/>



What is XML for Analysis?



- **XML for Analysis** is a set of XML Message Interfaces that use the industry standard SOAP to define the data access interaction between a client application and an analytical data provider (OLAP and Data Mining) working over the Internet.

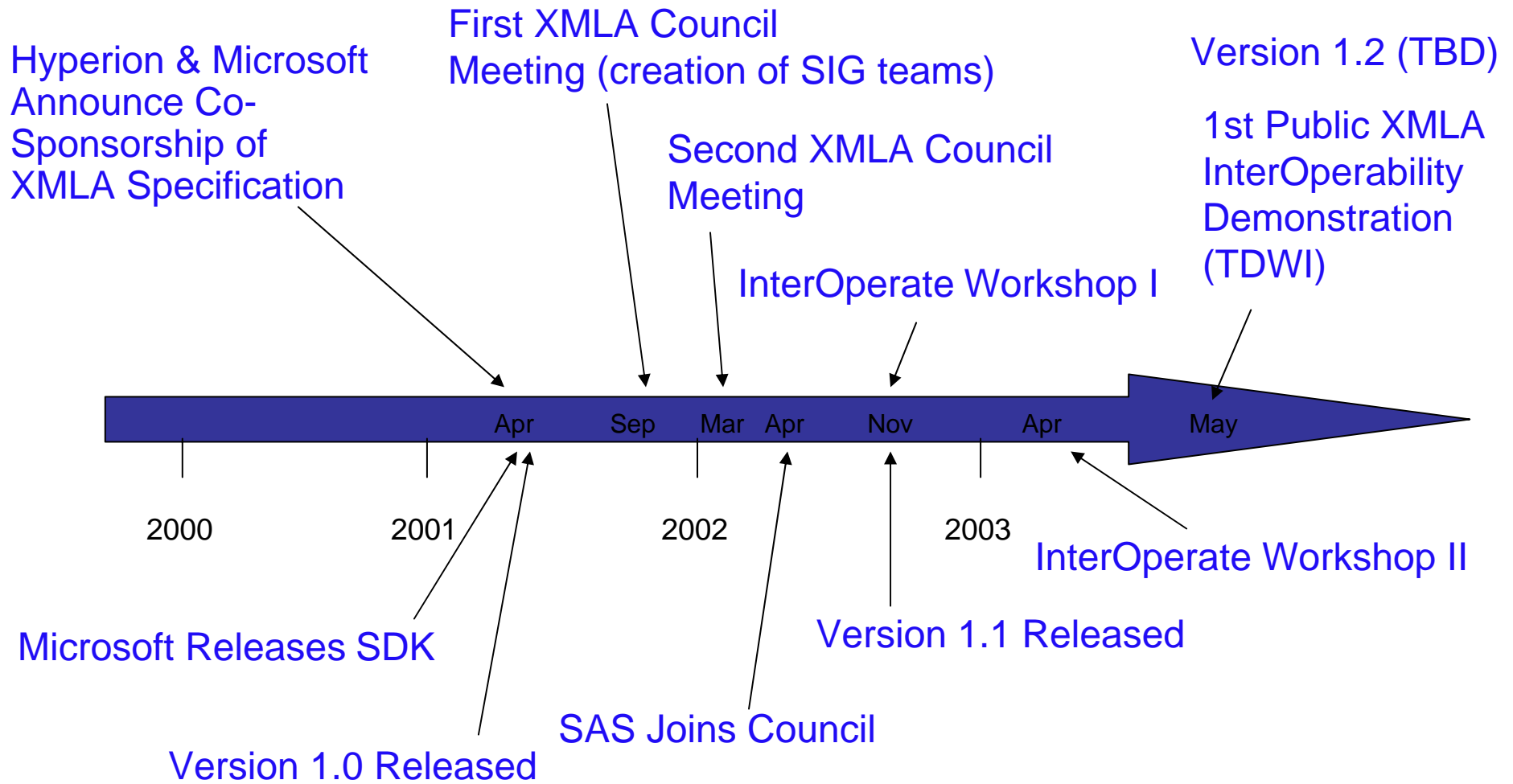


What are the benefits of XMLA?

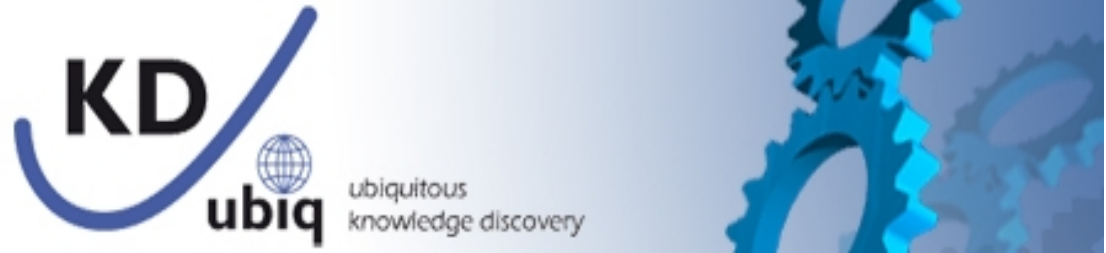


- **Customers** will gain the ability to protect server and tools investments and ensure that new analytical deployments will interoperate and work cooperatively.
- **Developers** will gain the ability to leverage existing developer skills and to use open access XML-based Web services, eliminating the need to program to multiple APIs and query languages.
- **Independent software vendors** will be able to reduce complexity and costs for development and maintenance by writing to a single access interface.

History of XMLA- Development



Example of XMLA SOAP Request



The following is an example of an **Execute** method call with `<Statement>` set to an OLAP MDX SELECT statement:

```
<?xml version="1.0" ?>
- <Execute>
  - <Command>
    <Statement>select [Measures].members on Columns from Sales</Statement>
  </Command>
  - <Properties>
    - <PropertyList>
      <DataSourceInfo>Provider=Essbase;Data Source=local;</DataSourceInfo>
      <Catalog>Foodmart 2000</Catalog>
      <Format>Multidimensional</Format>
      <AxisFormat>ClusterFormat</AxisFormat>
    </PropertyList>
  </Properties>
</Execute>
```

Example of XMLA SOAP Response



This is the abbreviated response for the preceding method call:

```
<?xml version="1.0" ?>
- <SOAP-ENV:Envelope xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/" SOAP-
  ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
- <SOAP-ENV:Body>
  - <m:ExecuteResponse xmlns:m="urn:schemas-microsoft-com:xml-analysis">
    - <m:return SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/">
      - <root xmlns="urn:schemas-microsoft-com:xml-analysis:mddataset">
        - <xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
          xmlns:xars="urn:schemas-microsoft-com:xars">
          <!-- The schema for the data goes here. -->
          </xsd:schema>
          <!-- The data in MDDataset format goes here. -->
        </root>
      </m:return>
    </m:ExecuteResponse>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

What Provider Vendors Support XMLA?



What Consumer & Consulting Vendors Are/ Will Support XMLA?





JDM: The Java API for Data Mining



Objectives



To develop a Java API that supports

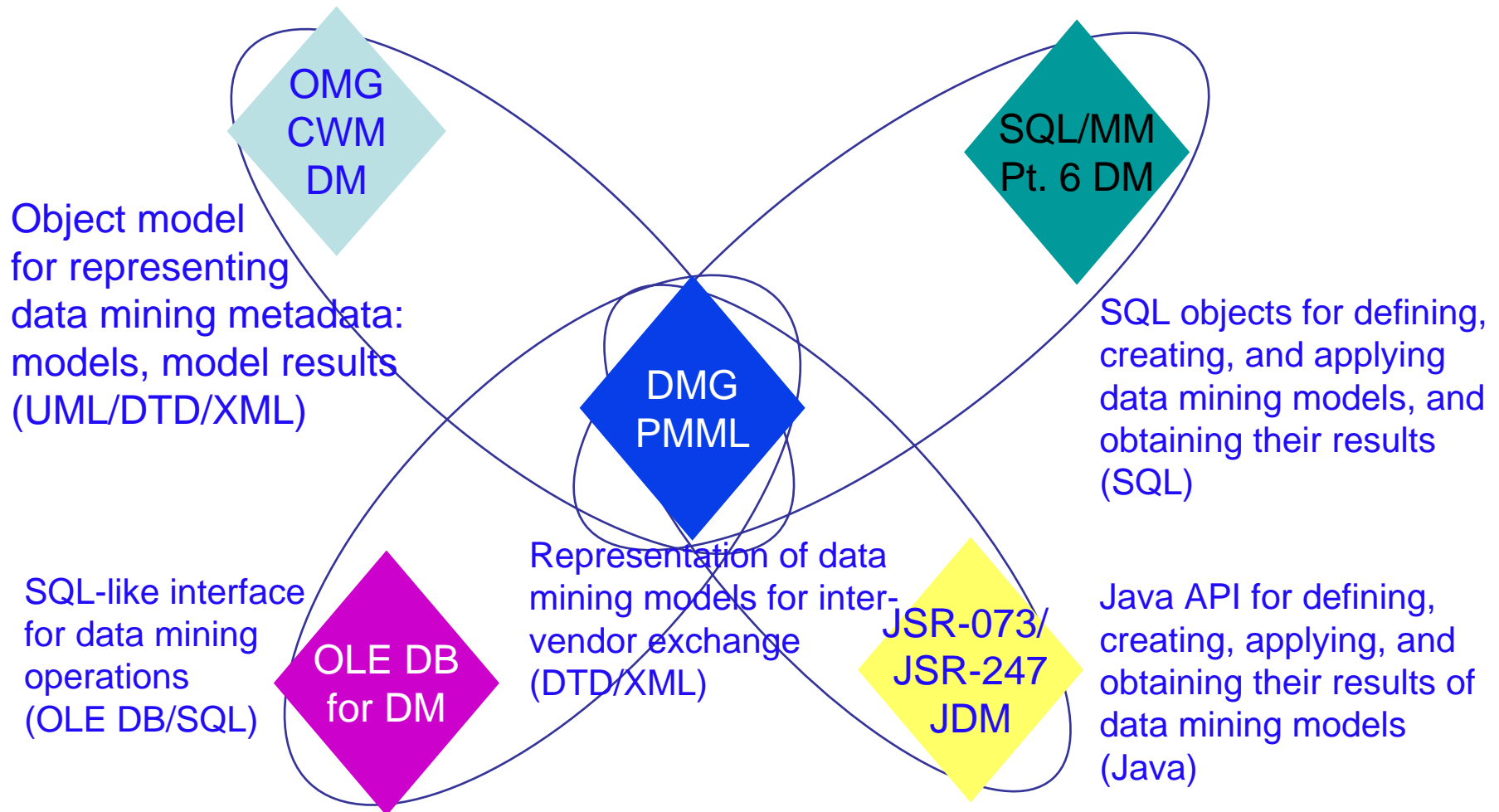
- Building of models
- Scoring of data using models
- Creation, storage, access and maintenance of data and metadata supporting data mining results
- To provide for data mining systems what JDBC™ did for relational databases
- Implementers of data mining applications can expose a single, standard API understood by a wide variety of client applications and components
- Data Mining clients can be coded against a single API that is independent of the underlying data mining system / vendor

Approach and Development



- Leverages other related standards
 - PMML (DMG)
 - CWM (OMG)
 - SQL/MM (ISO)
 - JCX (JSR-16)
 - JMI (JSR-40)
 - JOLAP (JSR-69)
 - CRISP-DM
 - OLEDB DM
- Final version of JDM 1.0, 2005:
<http://www.jcp.org/en/jsr/detail?id=73>
- JDM 2.0 (final release June 2007):
<http://www.jcp.org/en/jsr/detail?id=247>

Related Standards



Expert Group



- Mark Hornick, Oracle (Lead)
- BEA Systems CA, Inc.
- Corporate Intellect Ltd.
- E.piphany, Inc.
- Fair Isaac Corporation
- Hyperion Solutions Corporation
- IBM
- KXEN
- Oracle
- SAP AG
- SAS Institute Inc.
- SPSS
- Strategic Analytics
- Sun Microsystems, Inc.
- Weka (Trigg, Len)

Use Case I



A programmer is tasked with development of a target marketing tools that allows the user to

- Choose a target campaign
- E-mail a random sample of the customers
- Build a model based on the responses
- Apply the model to improve the targeting of the campaign

Use Case II



Using JDM (for the 3rd and 4th tasks) the programmer

- Defines the target data for the modelling using the Physical and Logical Data Classes
- Uses the Classification Function Settings class to set default parameters for the learning task
- Creates a build task that generates and persists the model
- Creates an apply task that applies the model to select the campaign targets
- Minimises risk associated with a change in the data mining vendor by using the standard JDM interface

How will it work?

Part I



JDM defines a set of interfaces for

- **Defining the data** to be used in the mining
 - Physical/Logical Data
- **Defining the data mining parameters**
 - Function settings
 - Support for Novice Users
 - Algorithm settings
 - Expert User
 - Algorithm specific settings

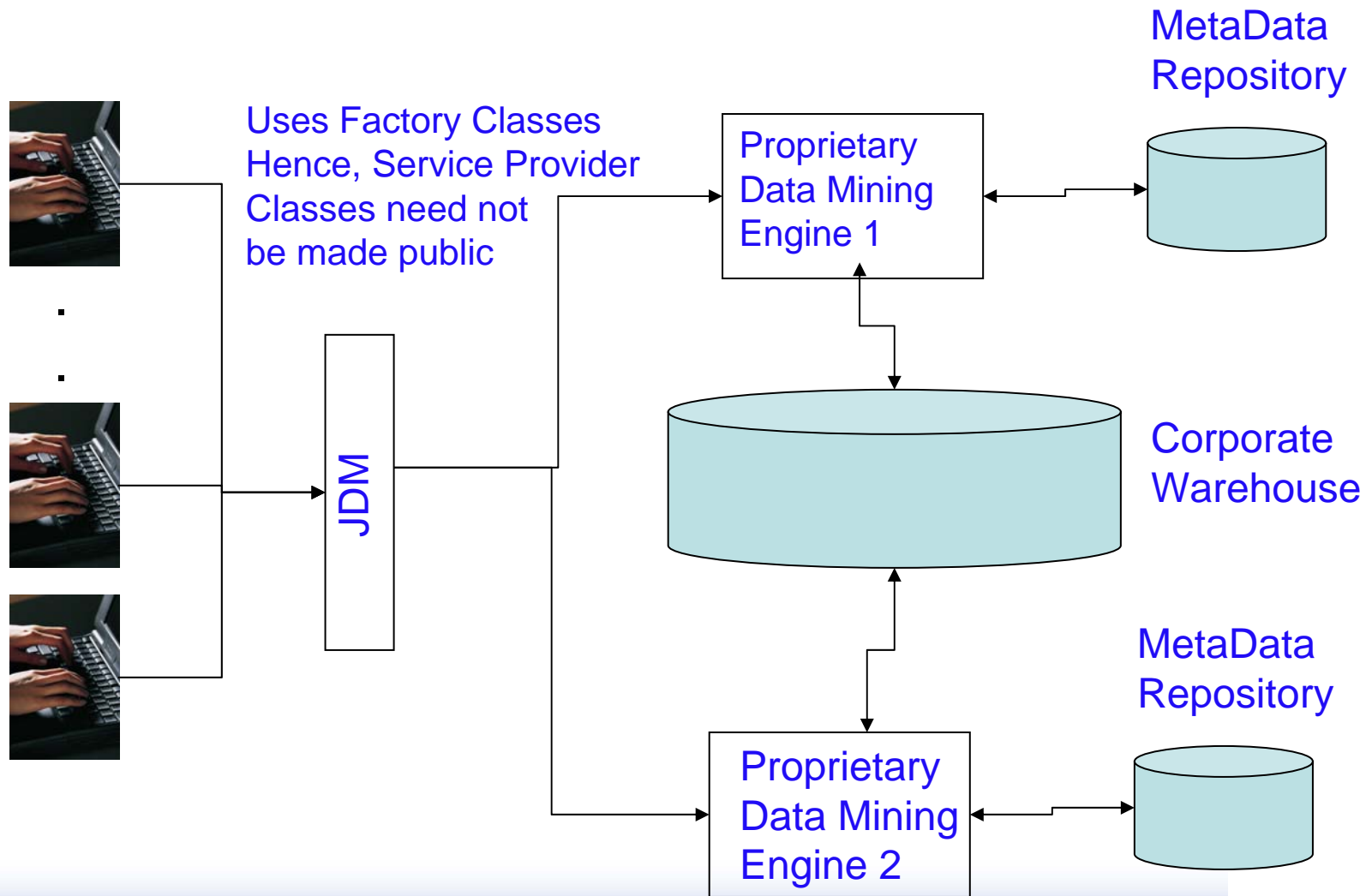
How will it work?

Part II

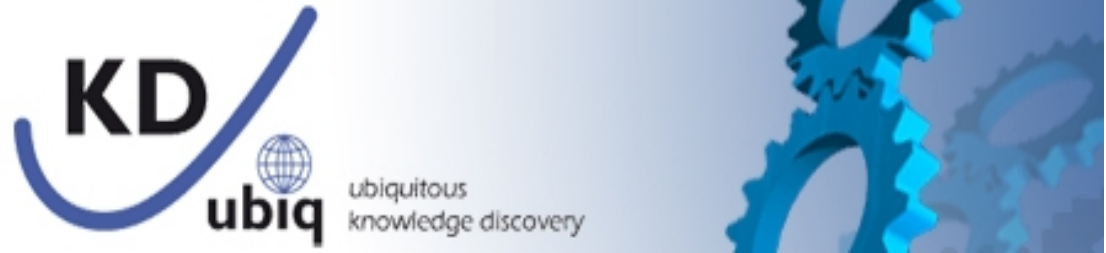


- Performing Tasks
 - Executing a data mining algorithm
 - Importing/Exporting to PMML
 - Testing the knowledge
 - Applying the knowledge on new data
 - Batch and Real-time Scoring
 - Compute Statistics
- Interrogating the resulting knowledge
- Persistence of all Meta Data/Data

Typical Architecture



Conformance Rules for Service Providers



- a la carte approach to functions and algorithms supported
 - vendors implement functions and algorithms that their products support
 - At least one function must be supported
- All core packages must be supported
- All methods within a implemented class must be implemented
 - semantics specified for each method must be implemented to ensure common interpretation of a given result
- Must support J2EE and/or J2SE
- Extension may be done through subclassing

Data Mining Functions Supported



- Classification
- Regression
- Attribute Importance
- Clustering
- Association Rules
- Feature Extraction (since 2.0)
- Time Series (2.0)
- Anomaly Detection (2.0)

Algorithms Supported



- Naïve Bayes
- Decision Trees
- Feed Forward Neural Networks
- Support Vector Machines
- K-Means
- Apriori (JDM 2.0)
- Non-negative Matrix Factorization (JDM 2.0)
- ARIMA (JDM 2.0)

Code Example: Part I



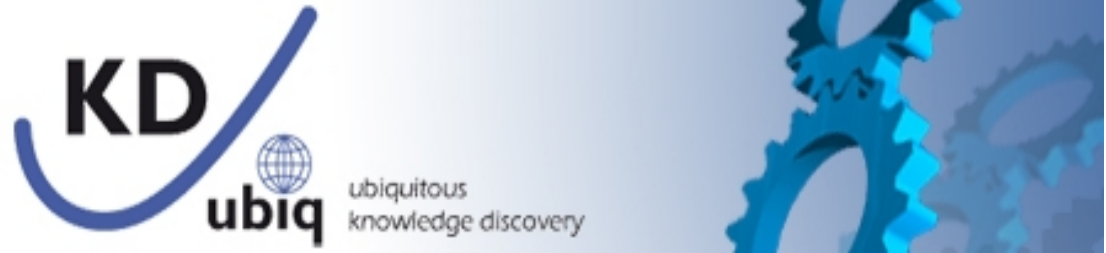
```
// Get a connection
(1) ConnectionSpec connSpec =
(javax.datamining.resource.ConnectionSpec)
jdmCFactory.getConnectionSpec();
(2)             connSpec.setName( "user1" );
(3)             connSpec.setPassword( "pswd" );
(4)             connSpec.setURI( "myDME" );
(5) javax.datamining.resource.Connection dmeConn =
jdmCFactory.getConnection(connSpec );
```

Code Example: Part II



```
// Create and populate the Physical Data object - Define the  
Data to be used  
(6) PhysicalDataSetFactory pdsFactory  
    = (PhysicalDataSetFactory) dmeConn.getFactory( "  
javax.datamining.data.PhysicalDataSet" );  
(7) PhysicalDataSet pd = pdsFactory.create( "minivan.data" );  
(8)          pd.importMetaData();  
(9) dmeConn.saveObject( "myPD", pd );
```

Code Example: Part III



```
// Create LogicalData object
(10) LogicalDataFactory ldFactory = (LogicalDataFactory)
    dmeConn.getFactory("javax.datamining.data.LogicalData" );
(11) LogicalData ld = ldFactory.create( pd );

// Specify how attributes should be used
(12) LogicalAttribute income = ld.getAttribute( "income" );
(13) income.setAttributeType( AttributeType.numerical );
```

Code Example: Part IV



```
// Create the FunctionSettings for Classification
(14) ClassificationSettingsFactory cfsFactory =
(ClassificationSettingsFactory) dmeConn.getFactory(

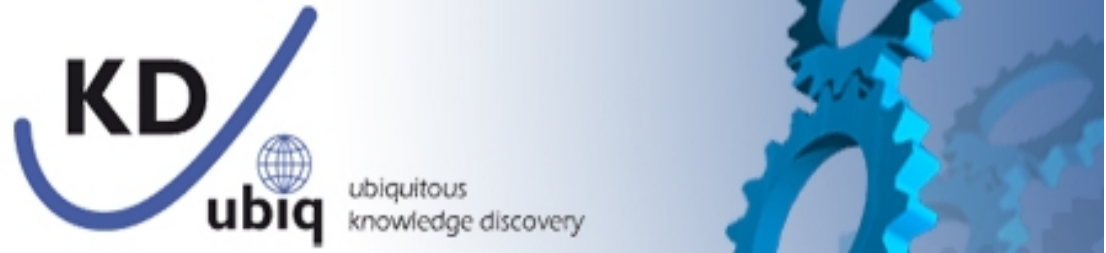
"javax.datamining.supervised.classification.ClassificationSettin
" );

(15) ClassificationSettings settings = cfsFactory.create();

(16) settings.setTargetAttributeName( "buyMinivan" );

(17) settings.setCostMatrix( costs ); // predefined cost matrix
```

Code Example: Part V



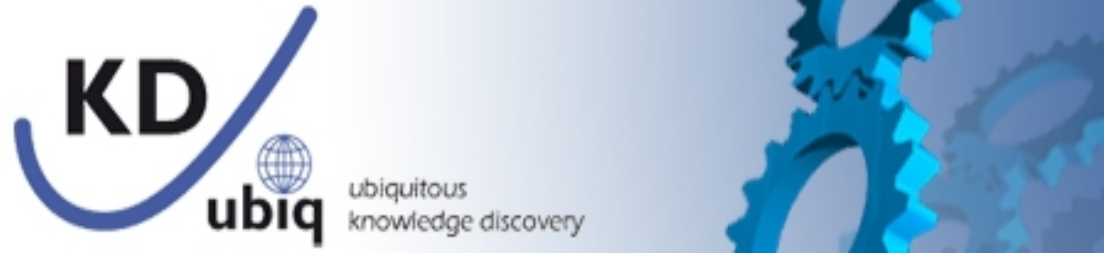
```
// Create the AlgorithmSettings and add it to the FunctionSettings
(18) NaiveBayesSettingsFactory nbFactory = (NaiveBayesSettingsFactory) dme-
Conn.getFactory(

“javax.datamining.algorithm.naivebayes.NaiveBayes-Settings” );

(19) NaiveBayesSettings nbSettings = nbFactory.create();
(20)             nbSettings.setSingletonThreshold( .01L );
(21)             nbSettings.setPairwiseThreshold( .01L );

// Associate LD and AS with the FunctionSettings
(22) settings.setAlgorithmSettings( nbSettings );
(23) settings.setLogicalData( ld );
(24) dmeConn.saveObject( “myFS”, settings );
```

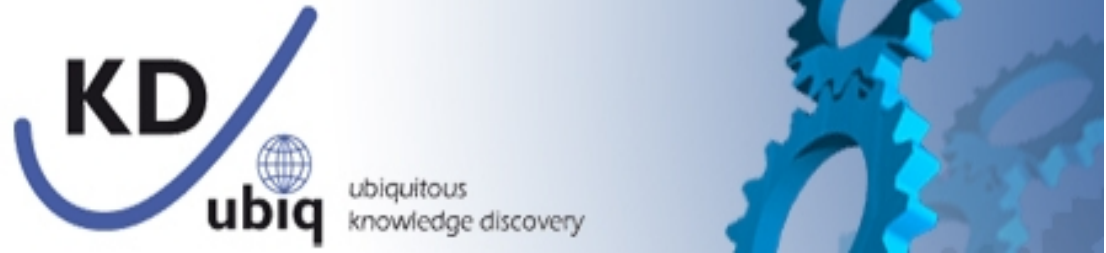
Code Example: Part VI



```
// Create the build task
(26) BuildTaskFactory btFactory
      = (BuildTaskFactory)
dmeConn.getFactory("javax.datamining.task.BuildTask" );
(27) BuildTask buildTask = btFactory.create( "myPD", "myFS", "myModel" );

(28) VerificationReport report = buildTask.verify();
(29) if ( report != null ) { // either error or warning
(30)     ReportType reportType = report.getReportType (); // check if it's
just a warning or an error
(32) } else {
(33)     dmeConn.saveObject( "myBuildTask", buildTask );
```

Code Example: Part VII



```
// Execute the task and block until finished
```

```
(34)     ExecutionHandle handle = dmeConn.execute( "myBuildTask" );
```

```
(35) handle.waitForCompletion( null );
```

```
    // wait without timeout until done
```

```
    // Access the model
```

```
(36)     ClassificationModel model
```

```
            = (ClassificationModel) dmeConn.getObject( "myModel",
```

```
            NamedObject.model );
```

```
(37) }
```

```
    // Close the connection
```

```
(38) dmeConn.close();
```

Extensions of JDM-2.0



- Work on JDM 2.0 is just finalized. Numerous extensions can be expected (see <http://jcp.org/aboutJava/communityprocess/pr/jsr247/index.html>)
- Active areas include:
 - Transformations
 - Time Series
 - Feature Extraction
 - Multi-record real-time scoring
 - Multi-target models
 - Multivariate statistics
 - Scheduling tasks and creating task dependencies
 - Text mining
 - Anomaly detection
 - Generic Settings
- JDM aims for stronger compatibility with PMML, SQL/MM, CWM-DM



PMML: The Predictive Model Markup Language

<http://www.dmg.org>

Predictive Model Mark-up Language (PMML)



- Industry led standard for representing the output of data mining
- Supported by
 - Full Members: IBM, Oracle, Magnify, SPSS, SAS, StatSoft, Microsoft, CorporateIntellect, KXEN, Salford Systems
 - Numerous Associated Members
- Objective
 - define and share predictive models using an open standard

Rationale



- Complex mosaic of **software applications**
 - **Knowledge generators**
 - Data Mining Vendors
 - Different data mining algorithms have different languages for expressing the knowledge discovered
 - Vendor dependent representations for knowledge e.g. C/C++ routines
 - **Knowledge consumers**
 - Real-time Scoring / Personalisation engines
 - Marketing Tools
 - Visualisation Tools
- Need for a vendor independent representation of data mining output

PMML and Ubiq. KD



Since model consumers in ubiquitous KD are much more varied (IT processes, visualization, embedded devices), a common standard and exchange format for models is even more important than for standalone applications.

PMML



- **Benefits**
 - proprietary issues and incompatibilities no longer a barrier to the exchange of models between applications
 - based on XML
 - develop models using any generator vendor, deploy the models using any consumer vendor application
- **Development**
 - Current Release 3.1 (since Dec. 2005):
<http://www.dmg.org/v3-1/GeneralStructure.html>
 - Supported by most current releases of member vendors applications

PMML Document



```
?xml version="1.0"?> <PMML
version="3.1"
xmlns="http://www.dmg.org/PMML-
3_1"
xmlns:xsi="http://www.w3.org/2001/
XMLSchema-instance" >
  <Header .../>
  <MiningBuildTask .../>
    <DataDictionary .../>
    <TransformationDictionary .../>
    <SequenceMiningModel .../>
    <Extension .../>
  </PMML>
```

- Basic XML structure
- DOCTYPE declaration not required
- A PMML document must
 - be a valid XML document
 - obey PMML conformance rules
- Root element <PMML>
- 6 child elements
 - 2 required
 - Header
 - Data Dictionary
 - 4 optional

Header: Part I



- **Attributes**
 - copyright
 - Description
- **Elements**
 - Application (that generated the PMML)
 - Name: Capri
 - Version: 2.0
 - Annotation
 - Free text
 - TimeStamp
 - Date/Time of model creation

Header: Part II



```
<?xml version="1.0" ?>  
<PMML version="3.1" >  
  <Header copyright="CorporateIntellect"  
description="Results of CAPRI" >
```

```
    <Application name="CORAL" version="3.0" >  
    <Annotation>This is a PMML document with results from the  
      CAPRI run on commodity market data.</Annotation>  
    <Timestamp>2003-03-02 18:30:00 GMT  
+00:00</Timestamp>
```

```
</Header>
```

```
...
```

```
...
```

```
</PMML>
```

Mining Build Task



- May contain any XML value describing the configuration of the training run that produced the model
- Information provided in this element is essentially meta-data
 - not used specifically in the deployment of the model by the PMML consumer
- Specific content structure not defined in PMML

Data Dictionary complete

```
<?xml version="1.0" ?>  
<PMML version="3.1" >  
  <Header ... />
```

```
    <DataDictionary numOfFields= "3" >  
      <DataField name= "Type" optype="categorical">  
        <Value value = "BU" />  
        <Value value = "HO" />  
        <Value value = "CO" />  
      </DataField>  
      <DataField name= "Age" optype= "continuous">  
        <Interval closure= "closedClosed" leftMargin= "0"  
rightMargin= "150" />  
      </DataField>  
      <DataField name= "PostCode" optype="categorical" taxonomy =  
"Location" />  
      <Taxonomy name="Location"> .....  
    </Taxonomy>  
  </DataDictionary >
```

```
</PMML>
```

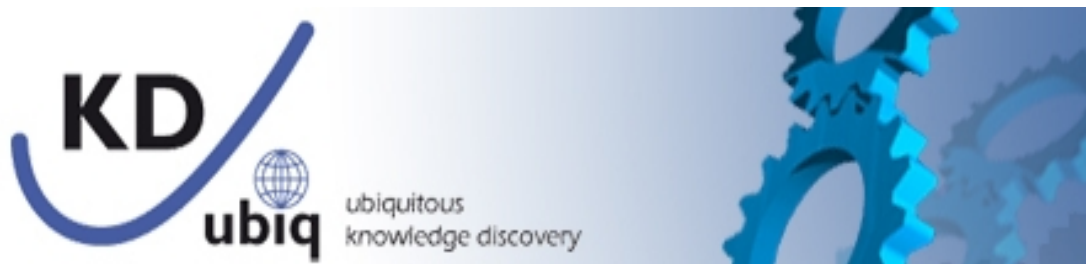
Data Dictionary: Part I



Attributes: - Number of Fields
» aids consistency checks

DataField		
Attributes I	Name	
	displayName	
	dataType	<ul style="list-style-type: none">• e.g. string, integer, date...
	Optype	<ul style="list-style-type: none">• categorical/ordinal/continuous• defines legal operations on the field
	Taxonomy	<ul style="list-style-type: none">• Name of taxonomy that defines a hierarchy on the values
	isCyclic	

Data Dictionary: Part II



Attributes: - Number of Fields

» aids consistency checks

DataField		
Attributes II	Value	<ul style="list-style-type: none">• defines domain for ordinal and categorical attributes value
		<ul style="list-style-type: none">• displayValue
		<ul style="list-style-type: none">• property: valid/ invalid/ missing
	Interval	<ul style="list-style-type: none">• Defines the range of valid values for continuous fields
<ul style="list-style-type: none">• closure: openClosed, closedOpen, openOpen, closedClosed• leftMargin• rightMargin		

Data Dictionary: Part III



Taxonomy	Define hierarchies on specific fields within the data dictionary
Attributes	name: associates the taxonomy with the appropriate field within the data dictionary (see DataField attribute taxonomy)
Element	ChildParent
Attributes	childField : name of field within the table (see Elements below) that represents the child value
	parentField : name of field within the table (see Elements below) that represents the parent value
	parentLevelField : name of field within the table (see Elements below) that represents the level in the hierarchy
	isRecursive : Yes/No: if the whole hierarchy is defined in the same table or an individual table per level

Data Dictionary: Part IV



Element	ChildParent
Elements	Inline Table/Table Locator

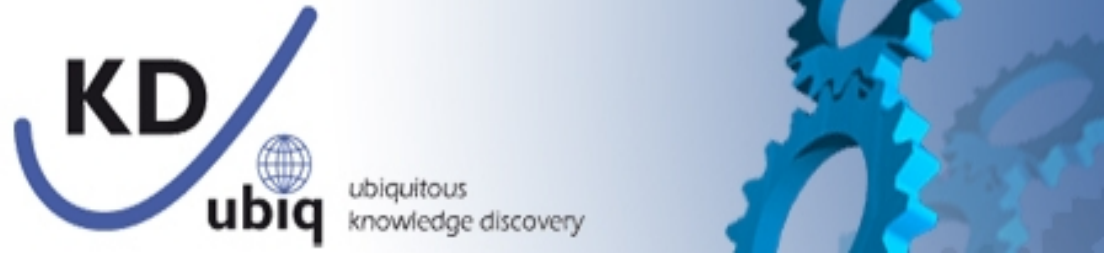
Data Dictionary complete

```
<?xml version="1.0" ?>  
<PMML version="3.1" >  
  <Header ... />
```

```
    <DataDictionary numOfFields= "3" >  
      <DataField name= "Type" optype="categorical">  
        <Value value = "BU" />  
        <Value value = "HO" />  
        <Value value = "CO" />  
      </DataField>  
      <DataField name= "Age" optype= "continuous">  
        <Interval closure= "closedClosed" leftMargin= "0"  
rightMargin= "150" />  
      </DataField>  
      <DataField name= "PostCode" optype="categorical" taxonomy =  
"Location" />  
      <Taxonomy name="Location"> .....  
    </Taxonomy>  
  </DataDictionary >
```

```
</PMML>
```

Taxonomy Example



```
<Taxonomy name="Location">
  <ChildParent childColumn="Post Code" parentColumn="District">
    <TableLocator x-dbname="myDB" x-tableName="PostCode_District" />
  </ChildParent>
  <ChildParent childColumn="member" parentColumn="group"
  isRecursive="yes"> <InlineTable>
  <Extension extender="MySystem">
    <row member="W9" group="CentralLondon"/>
    <row member="NW9" group="NorthLondon"/>
    <row member="NW2" group="NorthLondon"/>
    <row member="W1" group="CentralLondon"/>
    <row member="CentralLondon " group="London"/>
    <row member="NorthLondon " group="London"/>
    <row member="EastLondon " group="London"/>
    <row member="London" group="England"/>
  </Extension>
  </InlineTable>
</ChildParent> </Taxonomy>
```

Transformation Dictionary : Part I



- Defines mapping of source data values to values more suited for use by the mining algorithm
- PMML supports
 - **Normalization**: map values to numbers, the input can be continuous or discrete.
 - **Discretization**: map continuous values to discrete values.
 - **Value mapping**: map discrete values to discrete values.
 - **Aggregation**: summarize or collect groups of values, e.g. compute average

Transformation Dictionary: Part II



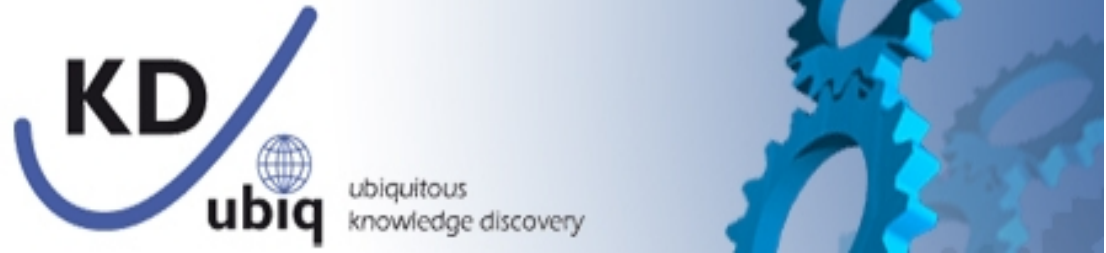
- **TransformationDictionary**
 - DerivedField Elements
 - **Attributes**
 - name
 - displayName
 - **Elements**
 - Expression (one of the following)
 - » NormContinuous
 - » NormDiscrete
 - » Discretize
 - » MapValues
 - » Aggregates

Transformation Dictionary: Part III



```
<DerivedField name="normalAge">  
  <NormContinuous field="age">  
    <LinearNorm orig="45" norm="0"/>  
    <LinearNorm orig="82" norm="0.5"/>  
    <LinearNorm orig="105" norm="1"/>  
  </NormContinuous>  
</DerivedField>  
<DerivedField name="male">  
  <NormDiscrete field="marital status" value="m"/>  
</DerivedField>  
<DerivedField name="female">  
  <NormDiscrete field="marital status" value="f"/>  
</DerivedField>
```

Transformation Dictionary: Part IV



```
<DerivedField name="binnedProfit">
  <Discretize field="Profit">
    <DiscretizeBin binValue="negative">
      <Interval closure="openOpen" rightMargin="0" />
    </DiscretizeBin>
    <DiscretizeBin binValue="positive">
      <Interval closure="closedOpen" leftMargin="0" />
    </DiscretizeBin>
  </Discretize>
</DerivedField>
<DerivedField name="houseType">
  <MapValues outputColumn="longForm">
    <FieldColumnPair field="Type" column="shortForm"/>
    <InlineTable><Extension>
```

Transformation Dictionary: Part V



```
<row><shortForm>BU</shortForm><longForm>bungalow</longForm> </row>  
<row><shortForm>HO</shortForm><longForm>house</longForm> </row>  
<row><shortForm>CO</shortForm><longForm>cottage</longForm> </row>
```

```
</Extension></InlineTable>  
  </MapValues>
```

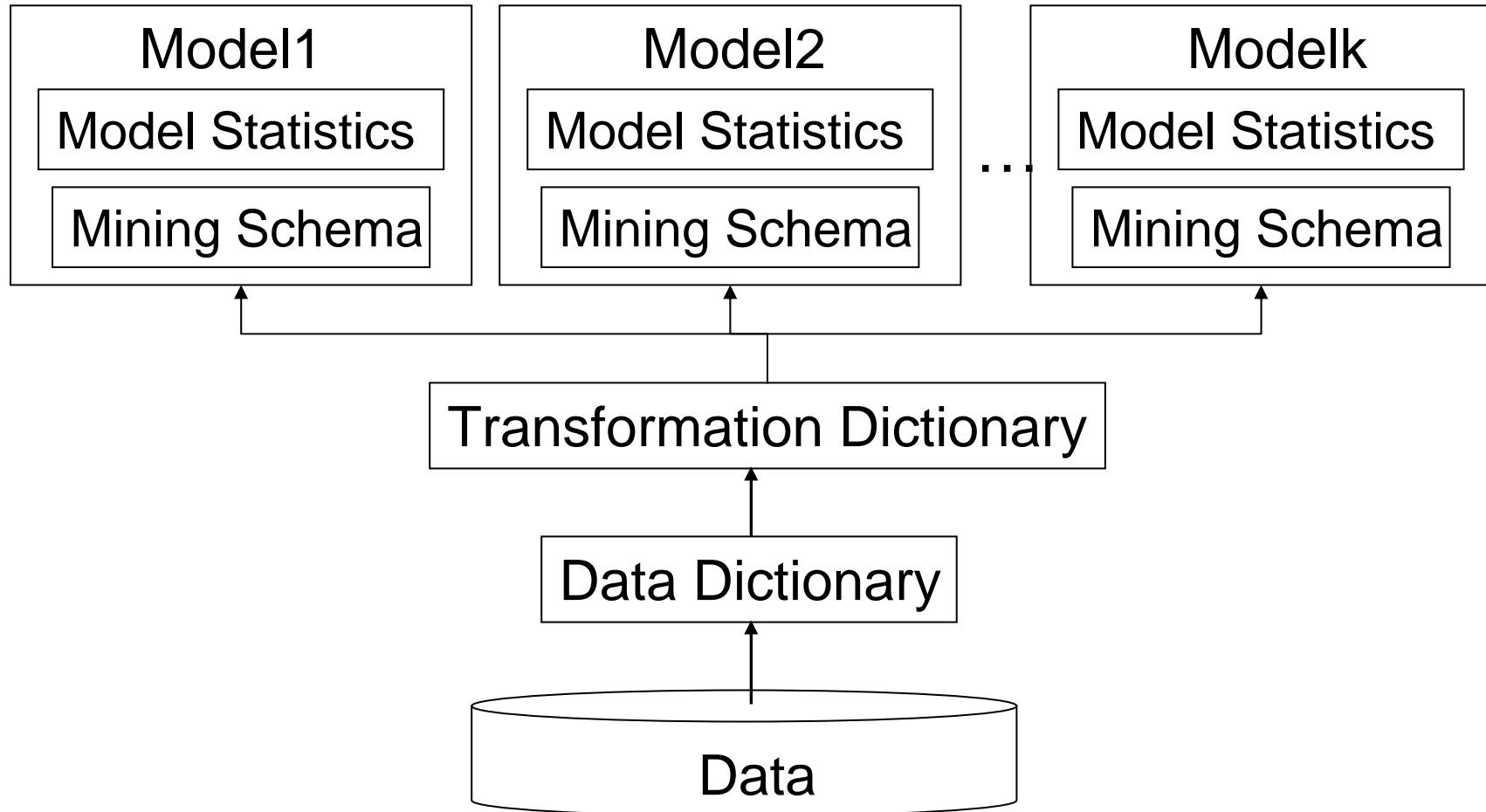
```
</DerivedField>
```

```
  <DerivedField name="itemsBought">
```

```
    <Aggregate field="item" function="multiset" groupField="transaction"/>
```

```
</DerivedField>
```

The PMML Document



Mining Schema



Element	MiningField
Attributes	Name
	usageType: active/ predicted/ supplementary
	Outliers: asIs/ asMissingValue/ asExtremeValues
	lowValue
	highValue
	missingValueReplacement
	missingValueTreatment: asIs/ asMean/ asMode/ asMedian/ asValue
	invalideValueTreatment

Mining Schema

```
<?xml version="1.0" ?>
<PMML version="3.1" >
  <Header ... />
  <DataDictionary ... />
  <SequenceModel functionName="sequences" algorithmName="Capri2"
    minimumSupport="24.17" minimumConfidence="0.00" numberOfItems="5"
    numberOfSets="5" numberOfSequences="11" numberOfRules="3">
    <Extension name="orderby" value="none"/>
    <MiningSchema >
      <MiningField name="Price" usageType="predicted" />
      <MiningField name="location" usageType="active" />
      <MiningField name="bedrooms" usageType="active" />
      <MiningField name="houseType" usageType="active" />
      <MiningField name="Area" usageType="supplementary" />
    </MiningSchema >
  </SequenceModel >
</PMML>
```

Model Statistics



- Elements
 - **UnivariateStatistics**
 - Attributes
 - Field
 - Elements
 - Discrete Statistics
 - Continuous Statistics
 - Counts: Valid, Invalid and Missing counts
 - NumericInfo: min/ max/ mean/ standard deviation/ median/ interQuartileDistance

Supported Data Mining Models



- Tree Model
- Neural Networks
- Clustering Model
- Regression Model
- General Regression Model
- Naïve Bayes Model
- Association Rules
- Sequence Rule Model
- Text Model (v. 3.1)
- Rule Set (v. 3.1)
- Support Vector Machine (v. 3.1)

Association Rules



Example for Association Rule model

Header Information

```
?xml version="1.0" ?>
```

```
<PMML version="3.1" xmlns="http://www.dmg.org/PMML-3_1"  
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
```

```
<Header copyright="www.dmg.org" description="example model for  
association rules"/>
```

see <http://dmg.org/v3-1/AssociationRules.html>

Association Rules



Data Dictionary

```
<DataDictionary numberOfFields="2" >  
  <DataField name="transaction" optype="categorical"  
    dataType="string" />  
  <DataField name="item" optype="categorical" dataType="string" />  
</DataDictionary>
```

Association Rules



Parameters of Mining

```
<AssociationModel  
  functionName="associationRules"  
  numberOfTransactions="4"  
  numberOfItems="3"  
  minimumSupport="0.6"  
  minimumConfidence="0.5"  
  numberOfItemsets="3"  
  numberOfRules="2">
```

...



Association Rules



Mining Schema

```
<MiningSchema>  
  <MiningField name="transaction" usageType="group" />  
  <MiningField name="item" usageType="predicted"/>  
</MiningSchema>
```

Association Rules



Items

```
<!-- We have three items in our input data -->  
<Item id="1" value="Cracker" />  
<Item id="2" value="Coke" />  
<Item id="3" value="Water" />
```

Association Rules



Item Sets

```
<!-- and two frequent itemsets with a single item -->  
<Itemset id="1" support="1.0" numberOfItems="1">  
  <ItemRef itemRef="1" />  
</Itemset>
```

```
<Itemset id="2" support="1.0" numberOfItems="1">  
  <ItemRef itemRef="3" />  
</Itemset>
```

```
<!-- and one frequent itemset with two items. -->  
<Itemset id="3" support="1.0" numberOfItems="2">  
  <ItemRef itemRef="1" /> <ItemRef itemRef="3" />  
</Itemset>
```

Association Rules



Association Rules

<!-- Two rules satisfy the requirements -->

```
<AssociationRule support="1.0" confidence="1.0" antecedent="1"
consequent="2" />
```

```
<AssociationRule support="1.0" confidence="1.0" antecedent="2"
consequent="1" />
```

```
</AssociationModel>
```

```
</PMML>
```



PMML Consumers



- Post-Processing
- Visualization
- Verification and Evaluation
- Deployment
- Hybrids and Meta-Learning

PEAR: Post-Processing Association Rules



- Sets of Association rules are browsed like web pages
- PMML-formated association rules can be uploaded
- Jorge et al., 2002
- www.liacc.up.pt/~amjorge/Projects/Class/software.html

The screenshot shows a web browser window titled "http://muralhas/pear/ - Microsoft Internet Explorer". The browser's address bar and menu bar are visible. The main content area displays a web page with a navigation bar at the top containing links for "home", "input", "visualize", and "export". The "visualize" link is currently selected. Below the navigation bar, the page title is "Visualize". There are several input fields and a button: "Support >= 0 %", "Confidence >= 0 %", "F(sup, conf):", "Navigation Operators: (Select here the ItemSet Operator)", and a "get Rules!" button. Below these fields, the page displays the following information:

PMML document: file://muralhas/pear/imp/en_infoline2.xml
with **Support** >=0 and **Confidence** >=0

items found: General_Statistics; Environment_and_Territory; Export_and_Import; Agriculture_and_Fishing; Population_and_SocialConditions; Economics; Industry_and_Energy; Commerce_Services_and_Tourism; Diverse;

Starting Page: <<Top 30 Support>>

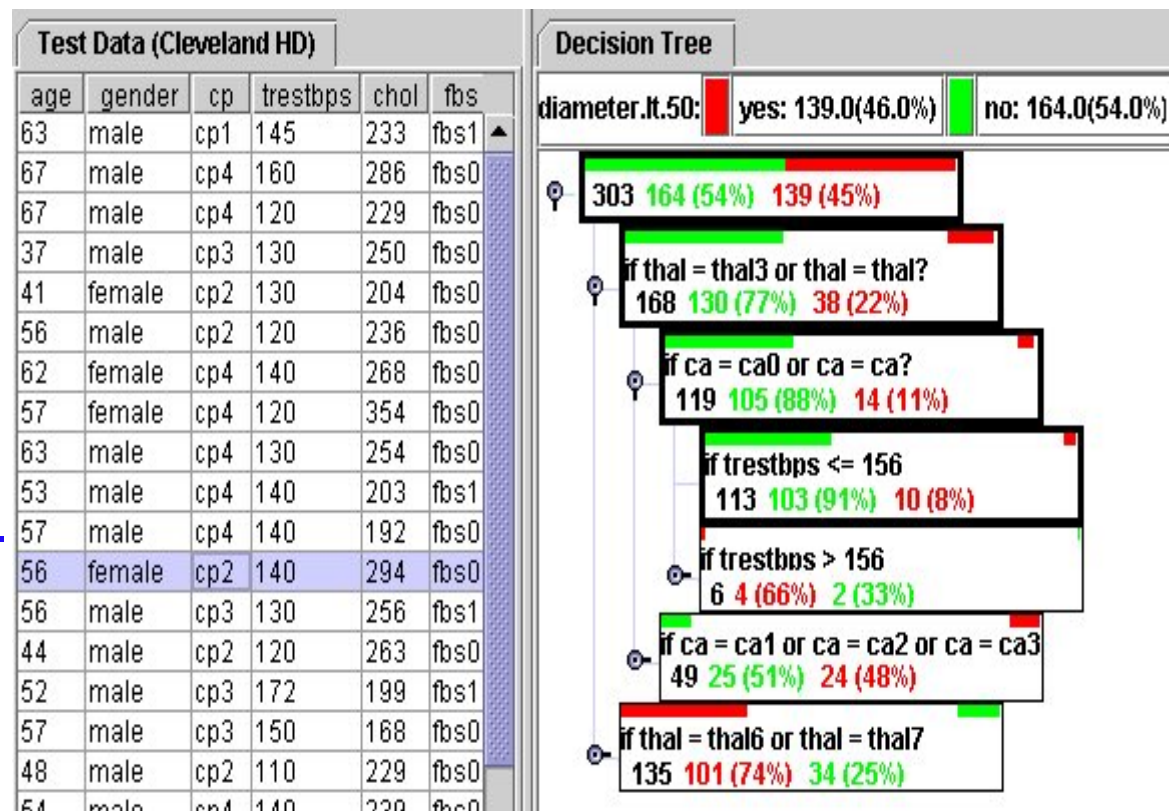
[[Rules chart](#) | [Support and Confidence chart](#)]

Id	Rules	Support	Confidence
3	Environment_and_Territory, Population_and_SocialConditions -> General_Statistics	0,161	0,405
160	Population_and_SocialConditions, Commerce_Services_and_Tourism -> Industry_and_Energy	0,158	0,481
33	General_Statistics -> Environment_and_Territory	0,141	0,539
45	Population_and_SocialConditions, Industry_and_Energy -> Environment_and_Territory	0,127	0,413
76	Population_and_SocialConditions, Industry_and_Energy -> Economics	0,127	0,635
170	Economics, Diverse -> Industry_and_Energy	0,127	0,548
193	Population_and_SocialConditions, Industry_and_Energy -> Commerce_Services_and_Tourism	0,127	0,596
203	Economics, Industry_and_Energy -> Commerce_Services_and_Tourism	0,11	0,633
34	General_Statistics, Population_and_SocialConditions -> Environment_and_Territory	0,103	0,631
41	Agriculture_and_Fishing, Population_and_SocialConditions -> Environment_and_Territory	0,099	0,494

VizWiz - PMML Visualization

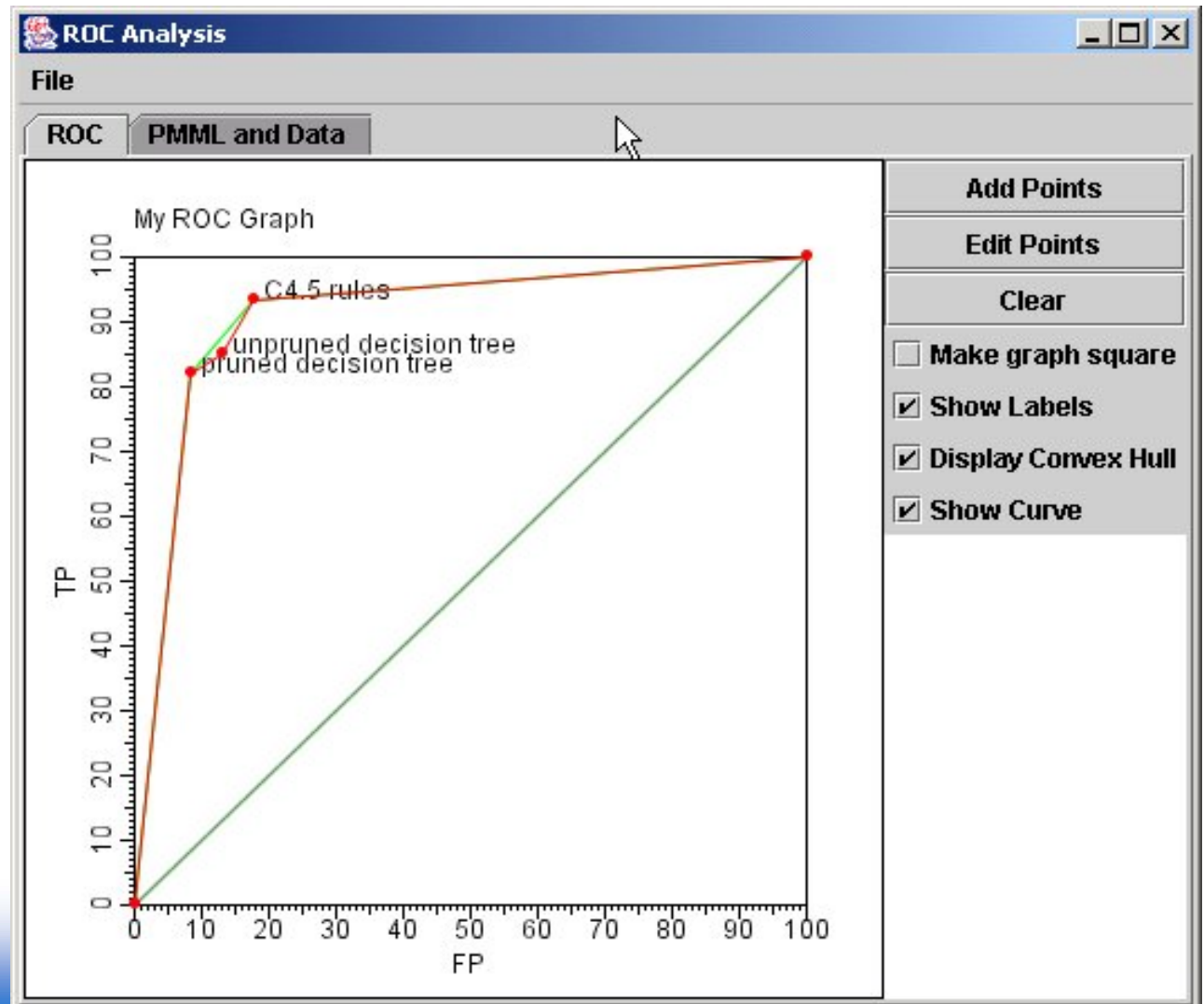


- Reads, visualizes and writes PMML files
- Coupling with WEKA in progress
- Java Applet
- Some non-standard extensions required for best visualization
- Wettschereck, 2003
- <http://soleunet.ijs.si/wbsite/html/senic-141-tool.html>



ROCOOn – Visualizing ROC graphs

- Use Receiver Operator Characteristics (ROC) to
 - compare and
 - evaluate models
- Java Applet
- Understands PMML as an extension to VizWiz
- Farrand and Flach (<http://www.cs.bris.ac.uk/%7Efarrand/rocon/index.html>)



Summary



- Standards help to streamline efforts
- Sign of maturity in field of KD
- From “Art” to “Engineering”
- Standards are still incomplete, but:
Use what is available!
- More tools utilizing standards are needed



Related Standards



- There are other relevant standards for Ubiquitous Knowledge Discovery of which we have to be aware, although they do not explicitly cover data mining:
 - Web Services
 - Grid Services
 - Semantic Web
- These standards are necessary for embedding data mining into larger processes and systems

Overview



- What is the grid?
- Example grid system: DataMiningGrid
- Details on Grid Middleware standards
- Example system: ACGT
- Details on Web Service standards



Related Standards: Grid



The Grid is offering an important environment for data mining tasks in distributed environments

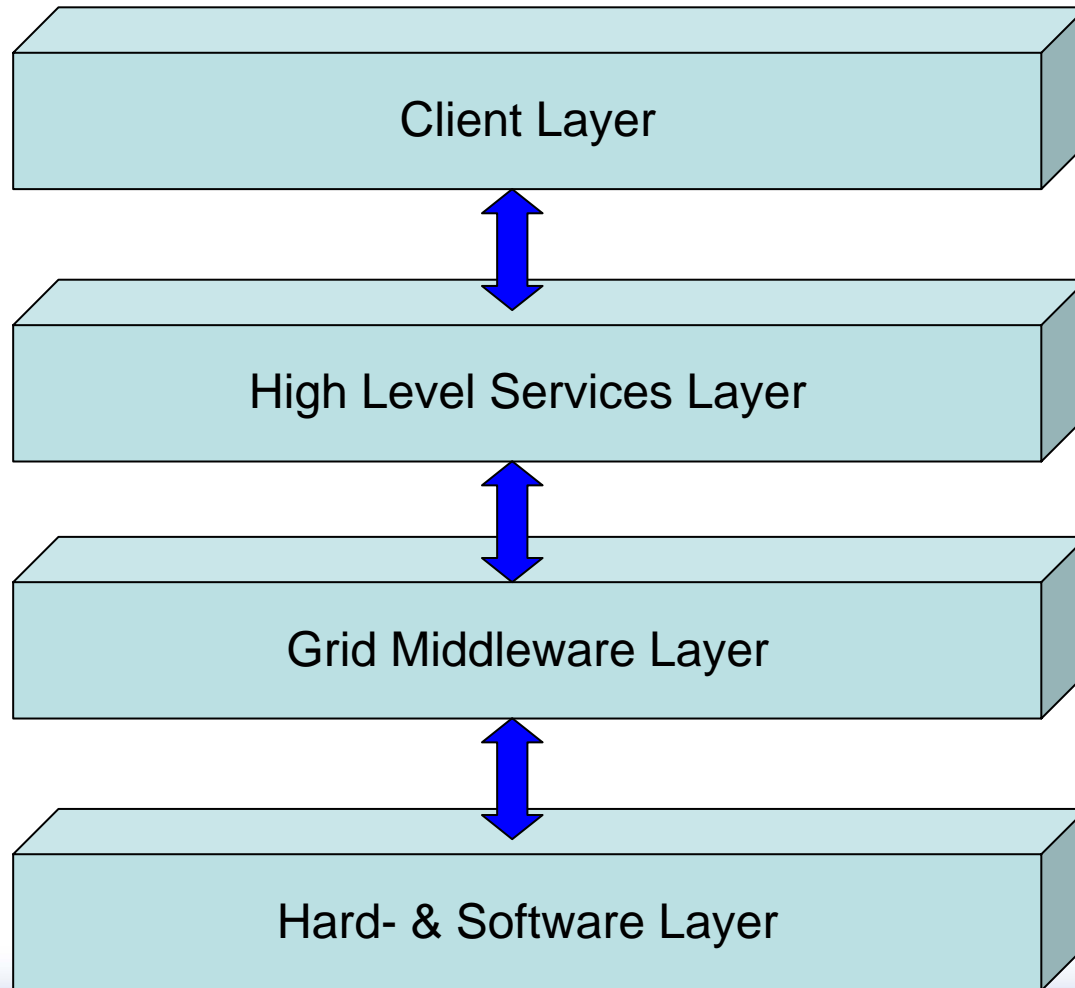
- **Definition:** „Coordinated resource sharing and problem solving in dynamic, multi-institutional virtual organizations.“ (Foster, Kesselman, Tuecke, *The Anatomy of the Grid*, 2001)

Related Standards: Grid



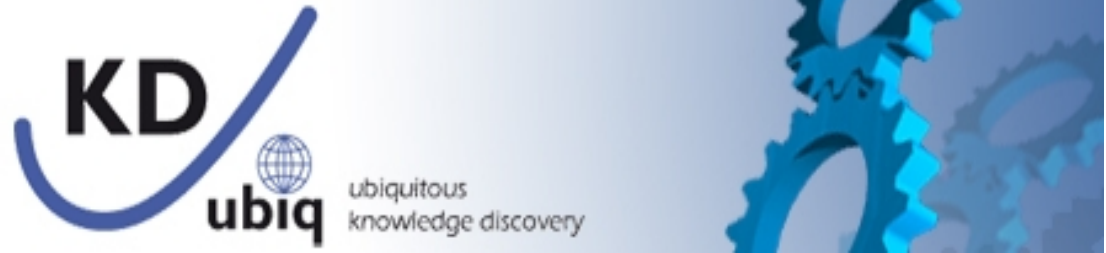
- **Type of grid:**
 - **Computational Grid** – aggregates computational power from a distributed collection of systems
 - **Data Grid** – secure access to distributed, heterogeneous pools of data
- **Types of applications**
 - Distributed supercomputing applications (lots of CPU and memory)
 - High-throughput computing using otherwise idle resources
 - On-demand computing applications integrating remote resources with local computation
 - Data intensive computing applications etc.

Common System Architecture



- Converging architecture for grid-enabled data mining systems
- Layered architecture for decomposition into groups with a particular level of abstraction
- Different layers address different standards

Example System: DataMiningGrid



- Objectives:
 - To develop framework for developing and deploying data mining applications on the grid
 - This means to develop new generic interfaces, grid services and components that will allow for seamless use of existing data mining algorithms in grid computing environments
- EU-funded project IST-2004-004475, Duration: Sept. 2004 – Nov. 2006
Result Availability: Software will be available as open source under Apache License V2, gone to SourceForge early 2007
- Project website <http://www.datamininggrid.org>
- Partners: University of Ulster (coord.), Fraunhofer IAIS, DaimlerChrysler, Univ. Ljubljana, Technion

DataMiningGrid Technologies

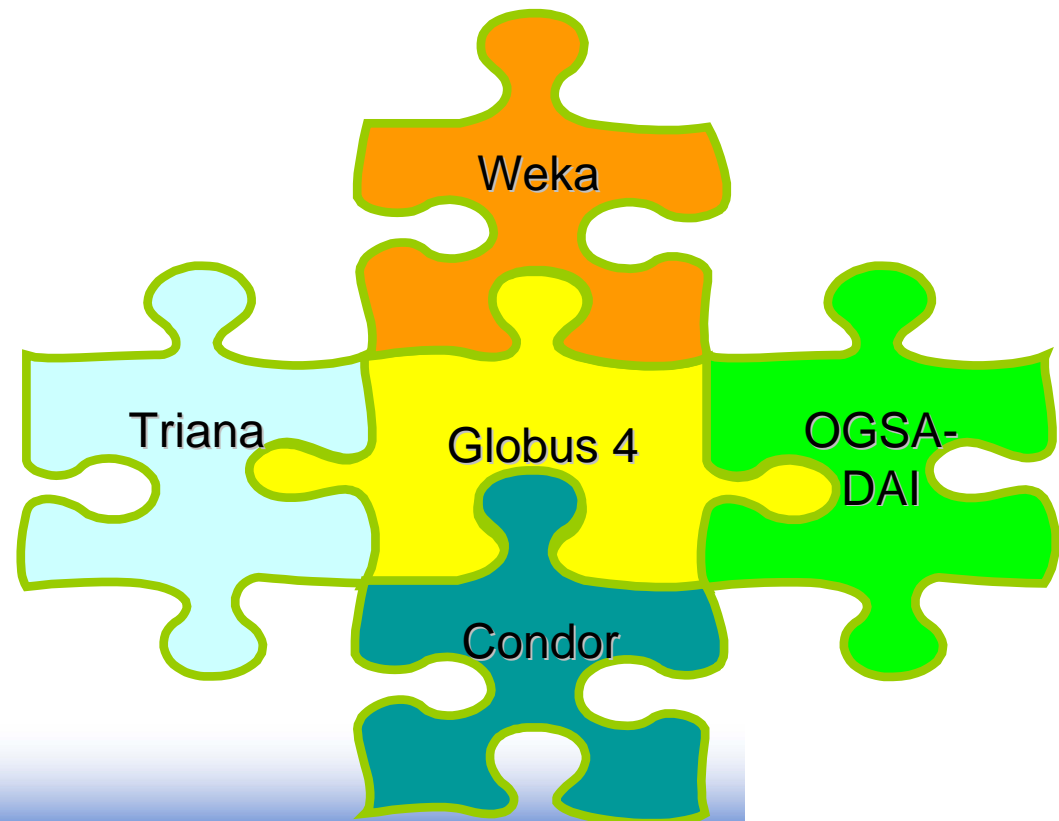


To build such a complex system, we have to use existing open source components as much as possible!

Only possible with standards!

- **Distributed Batch Processing:** Condor
- **Middleware:** Globus 4 Toolkit
- **Workflow:** Triana Workflow Editor
- **Data Access:** OGSA-DAI
- **Data Mining:** Weka Data Mining Toolkit

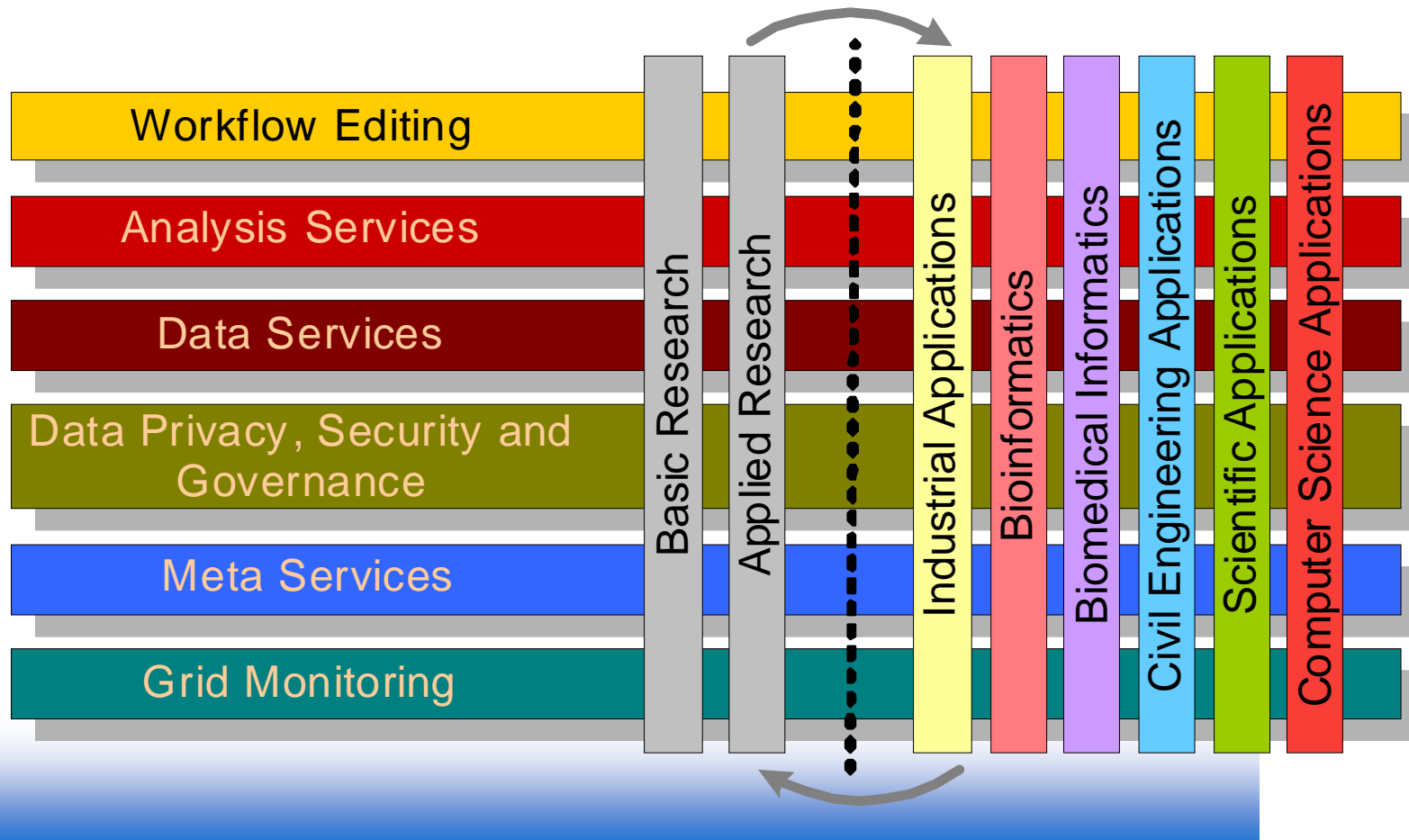
- **User Interfaces**
 - **Graphical Workflow Editor** for Expert Users
 - **Customizable web interfaces** for End Users



DataMiningGrid Building Blocks



To develop generic framework for developing and deploying data mining applications on the grid

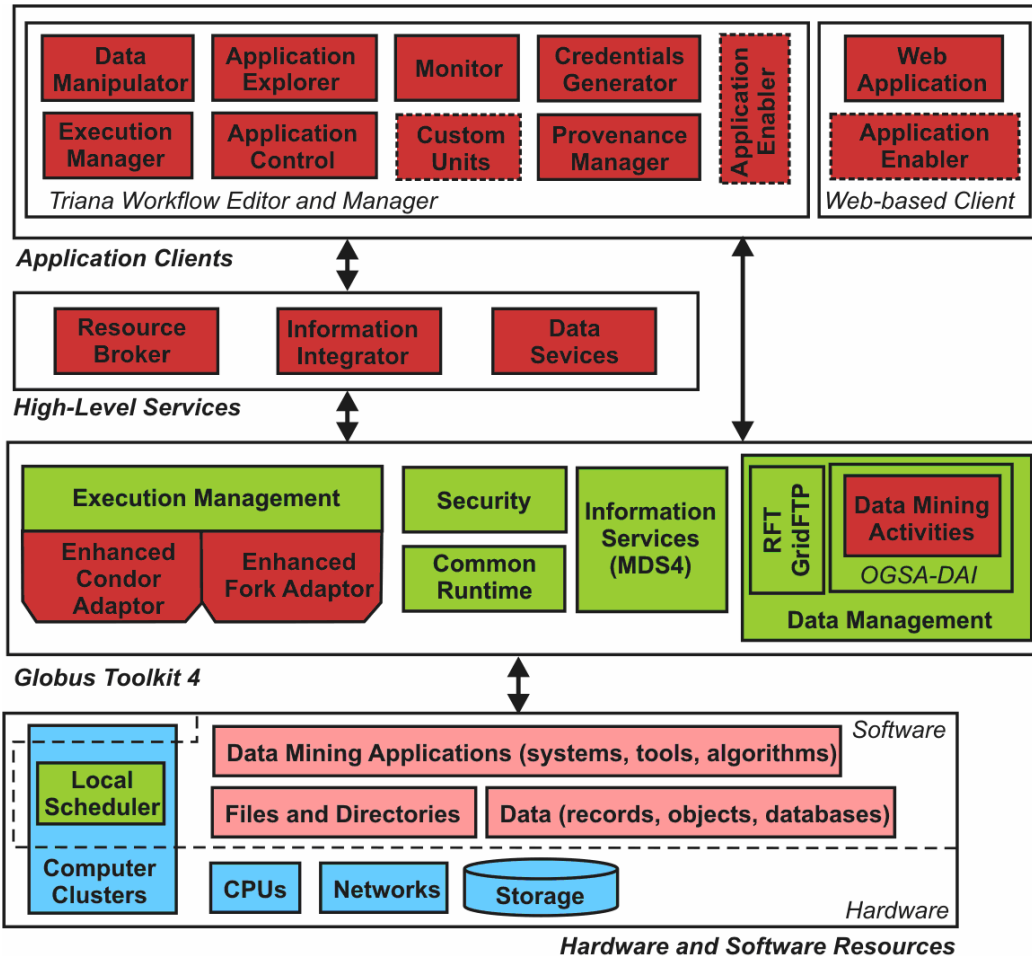


Detailed technical objectives



- Establish *requirements* for data mining in grid computing environments
- Develop grid tools and services
 - *Grid-enable* data mining applications in a generic way
 - *User-oriented workflow* management of data mining processes
 - *Data access* to data and *association* of data with data mining operations
 - *Identification, characterization, dynamical selection/allocation of data mining resources* in grid computing environments
 - *Evaluation* of developed technology on basis of a test bed consisting of selected demonstrator applications
 - Promote and encourage adoption of technology

DataMiningGrid Architecture

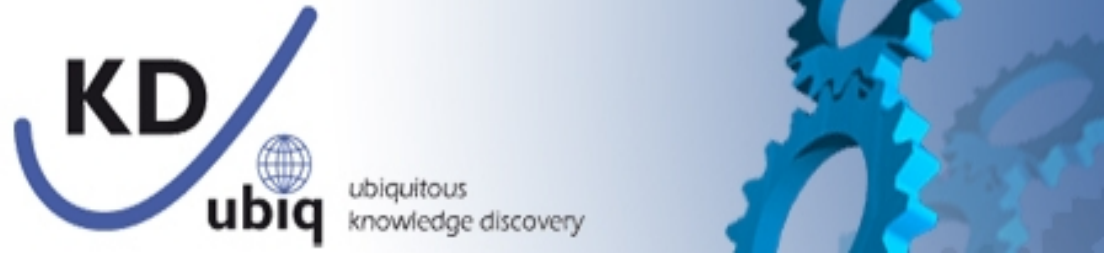


- Based on mainstream grid technology
 - Triana workflow editor and manager
 - GT4 as core grid middleware, Condor as local scheduler
 - GridBus resource broker
 - OGSA-DAI data access and integration

Legend:

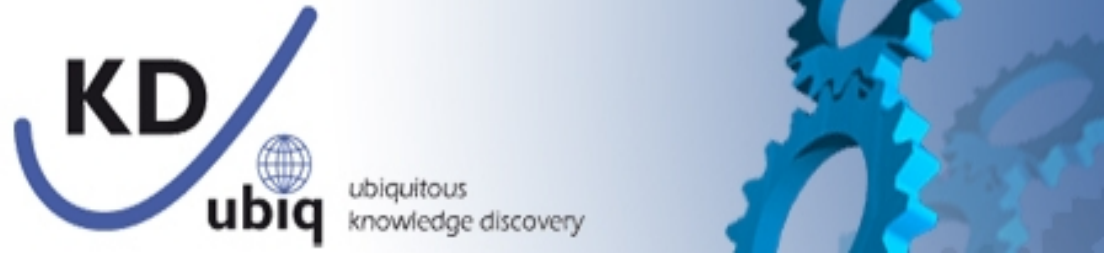
- | | |
|---|--|
| Components and services developed by DataMiningGrid (DMG) | Software resource: Data and DMG applications |
| Middleware: GT4, local scheduler, etc. | Hardware resource: Primary/secondary storage |
| | Hardware resource: Processing elements, networks, etc. |
| | Interfacing between components, services and layers |

Hard- & Software layer



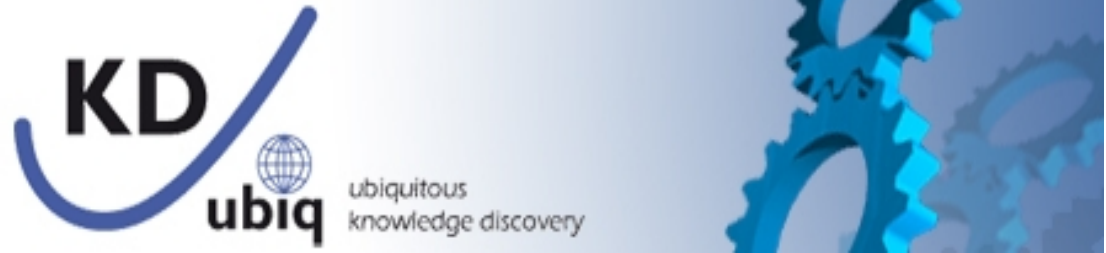
- Layer refers to
 - Hardware Resources (CPU, Storage, Network etc.)
 - Software Resources (DM-Apps, Data, ...)
 - Local Schedulers (Clusters)
- Standards involved
 - Condor (as scheduler)
 - Java (as programming language for applications)
 - Weka (data mining algorithms)

Grid Middleware Layer



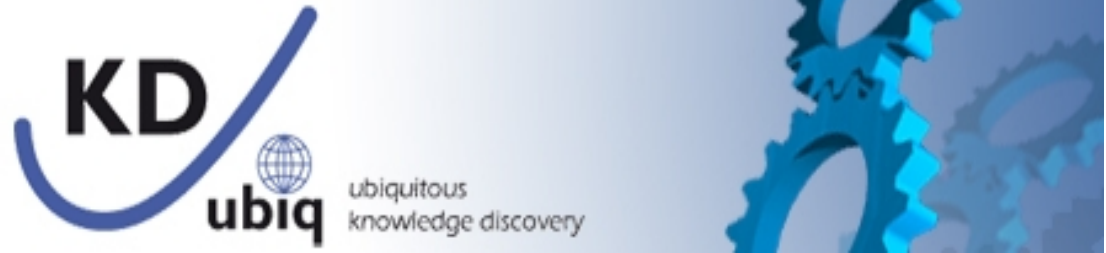
- Layer refers to
 - Grid middleware & services
 - Execution Management
 - Data Access
 - Information
 - Security
- Standards involved:
 - GT4 as middleware (Job Description Format)
 - OGSA-DAI
 - GridFTP
 - X.509 certificates

High Level Services Layer



- Layer refers to
 - High level components (services)
 - Resource brokering
 - Extended information
 - Extended data
- Standards involved
 - WSRF

Client Layer



- Layer refers to
 - Client side components
 - Web apps
 - Workflow construction
- Standards involved
 - Triana WF Editor and Manager

Grid Middleware Standards



- There are many attempts to standardize Grid technology. Especially important for Ubiq. KD are
 - OGSA (Open Grid Service Architecture)
<http://www.globus.org/ogsa>
 - OGSA-DAI (Data Access Integration)
<http://www.ogsadai.org.uk/>
- Globus Toolkit as *de facto* middleware standard
<http://www.globus.org>
- There is convergence among Grid and Web Services standards

Globus Toolkit 4



- The *Globus Toolkit* is a software toolkit that can be used to develop grid-based applications
 - OGSA (Open Grid Service Architecture) implementation from version 3.x
 - ver. 4 released on April 29, 2005 is *de facto* implementation of the WSRF (Web Service Resource Framework) specification
 - Current release: Globus Toolkit 4.0.5 (June 25, 2007)
 - includes many *high-level services* (GRAM (Grid Resource Allocation Manager), MDS (Monitoring and Discovery System), ...) that we can use to build grid applications
 - includes OGSA-DAI WSRF compliant version

Globus Toolkit



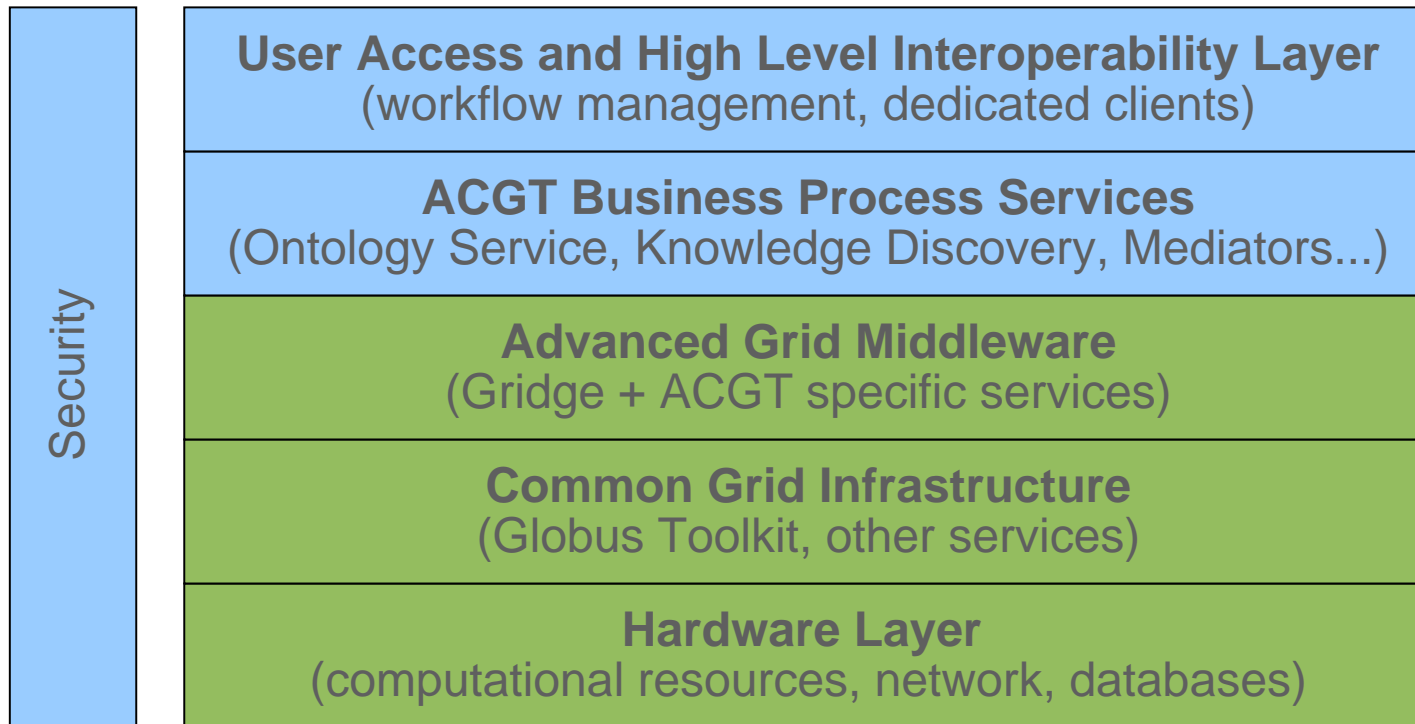
Taken From: Lee Liming: A Globus Primer,
<http://www.globus.org/toolkit/presentations/>

ACGT



- **Project full title:**
Advancing Clinico-Genomic Clinical Trials on Cancer
- **Goal:**
 - deliver a new weapon in the fight against Cancer
 - faster diagnosis and sharper identification of the optimal treatment for every patient
 - develop an advanced GRID architecture allowing the analysis and comparison of clinical and genetic results within large scale databases
- EU-funded project FP6-IST-026996, Duration: Feb. 2006 – Jan. 2010
- Project website <http://eu-acgt.org>
- 25 partners

ACGT high level Architecture

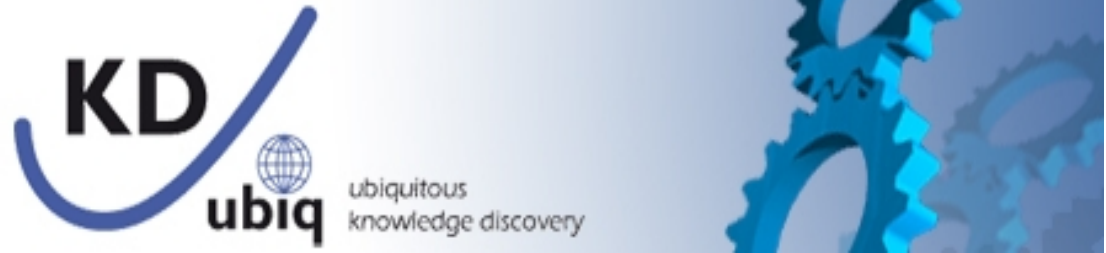


Layers description (1)



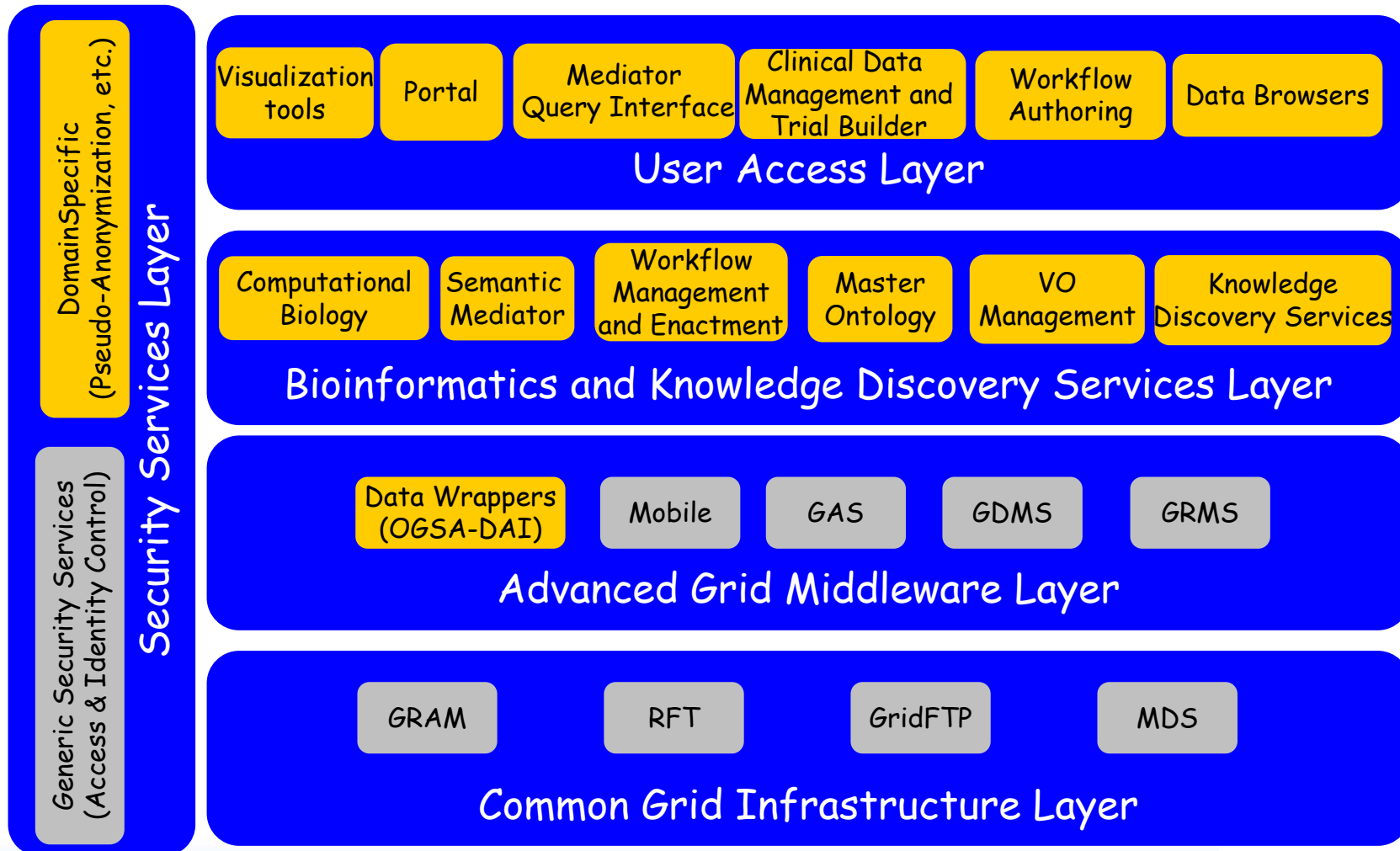
- Hardware Layer
 - Basic hardware infrastructure, Computational resources, Network, Databases
- Common Grid Infrastructure
 - Provides remote access to individual resources from Hardware Layer
 - Globus Toolkit: GRAM, GridFtp, MDS..
- Advanced Grid Middleware
 - Monitoring sensors, Collective services (operate on a set of lower level services to provide more advanced functionality), Gridge Toolkit (GRMS, GAS, Monitoring Tools, DMS, Mobile support services), OGSA-DAI, Other ACGT specific services

Layers description (2)

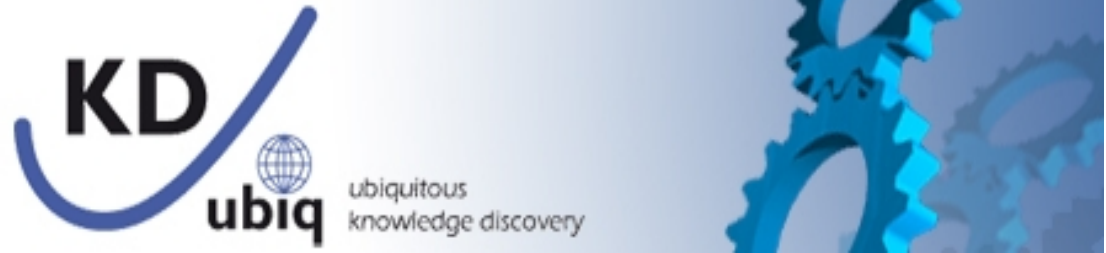


- ACGT Business Process Services
 - High level services providing interoperability in ACGT environment and integration of different data and resources
 - Ontology Services, Knowledge Discovery Services, VO Management Services, Mediator Services, Analytical Services, Vocabularies
 - Implemented as **Web services**
- User Access and High Level Interoperability Layer
 - Applications providing access to ACGT Environment for end user (standalone applications or portals)
 - Workflow management applications (Taverna + scuf! BPEL?)
 - Applications dedicated for specific ACGT scenarios
 - Visualization Tools

Components in the architecture

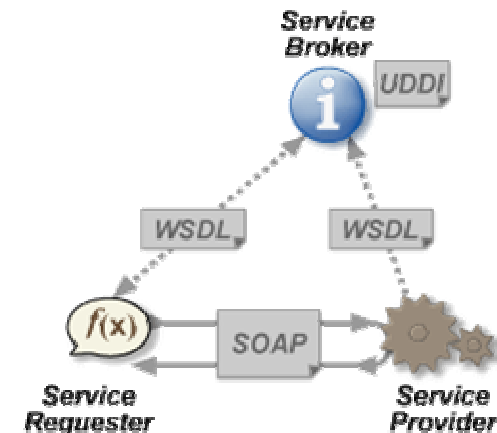


Web services



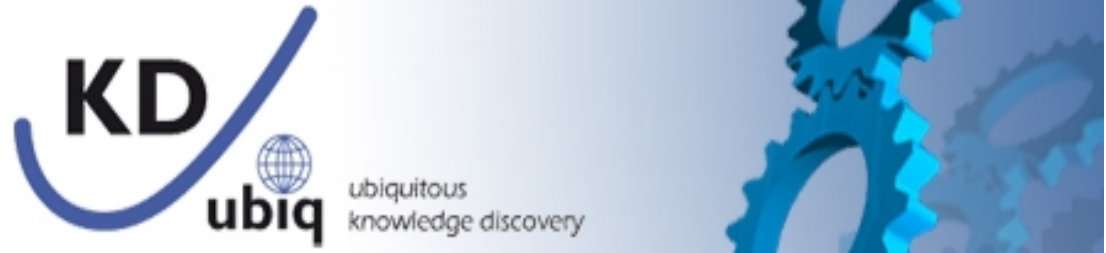
- Web services increasingly become the glue for combining all kinds of IT-services
- Highly relevant environment for data mining
- Already used by some of the data mining standards mentioned before
- Too many standards for web services exist for describing or even listing them all, examples are
 - WSDL describes remote application
 - SOAP/XML allows access
 - Etc

Most specifications are standardized at [OASIS](#) and the [W3C](#).



Source: Wikipedia

Web services (2)



- Few core specifications, the most common are:
 - **UDDI** (Universal Description, Discovery, and Integration): XML based metadata registry for web services
 - **WSDL** (Web Service Description Language): XML based description of remote application (methods, parameters etc.)
 - **SOAP** (Simple Object Access Protocol): XML based protocol for communication with bindings to underlying protocols like HTTP, SMTP etc.
 - **WS-Security**: Definition of how to use XML Encryption and XML Signature in SOAP to secure message exchanges.
 - **WS-ReliableExchange**: Protocol for reliable messaging between Web services.

ACGT Tools



- Grid Jobs
Submitted directly to Grid infrastructure using Grid Resource Management System
- ACGT Services
Tools implemented Web Services, described in WSDL, accessed through network using SOAP protocol
- Important for ACGT:
 - **GSI (Grid Security Infrastructure)**
GSI enabled Web services for seamless integration in the grid environment

References: Part I



- Grossman, R.L., Hornick, M.F., Meyer, G. (2002). *Data Mining Standards Initiatives*, Communications of the ACM, Vol. 45:8 see also <http://www.dmg.org>
- Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000). *CRISP-DM 1.0: Step-by-step data mining guide*, CRISP-DM consortium, <http://www.crisp-dm.org>
- Clifton, C., Thuraisingham, B. (2001). *Emerging standards for data mining*. Computer Standards & Interfaces Vol 23 pp 187 – 193.
- *Compare and Contrast JOLAP and XML for Analysis* http://www.essbase.com/resource_library/articles/jolap_xmla.cfm
- JCX <http://www.jcp.org/en/jsr/detail?id=016>
- JOLAP <http://www.jcp.org/en/jsr/detail?id=69>

References: Part II



- Jorge, A., Poças, J. and Azevedo, P. (2002). *Post-processing operators for browsing large sets of association rules*. Proc. Discovery Science 02. (eds. Lange, S., Satoh, K. and Smith, C. H.), Lübeck, Germany, LNCS, 2534, Springer-Verlag.
- Farrand, J. and Flach P. (2003). *ROCOOn: a tool for visualising ROC graphs*. See: <http://www.cs.bris.ac.uk/%7Efarrand/rocon/index.html>
- Melton, J. and Eisenberg, A. *SQL Multimedia and Application Packages (SQL/MM)*, <http://www.acm.org/sigmod/record/issues/0112/standards.pdf>
- OMG Common Warehouse MetaModel <http://www.omg.org/cwm/>
- SOAP <http://www.w3.org/TR/SOAP/>
- Tang, Z., Kim, P. *Building Data Mining Solutions with SQL Server 2000*, <http://www.dmreview.com/whitepaper/wid292.pdf>

References: Part III



- Wettschereck, D., Jorge, A., Moyle, S. *Data Mining and Decision Support Integration through the Predictive Model Markup Language Standard and Visualization* in Mladenic D, Lavrac N, Bohanec M, Moyle S (editors): *Data Mining and Decision Support: Integration and Collaboration*, Kluwer Publishers.
- XMLA <http://www.xmla.org/>
- Schwenkreis, F. ,SQL/MM Part 6: Data Mining;
www.jtc1sc32.org/doc/N0601-0650/32N0606T.pdf

Further Resources



There have been workshops on data mining standards at various past KDD conferences.

Some of the proceedings are available online:

- KDD'03: www.ncdm.uic.edu/workshops/dm-ssp03.htm
- KDD'04: www.ncdm.uic.edu/workshops/dm-ssp04.htm
- KDD'05 www.ncdm.uic.edu/documents/DMSSP-proceedings-2005.pdf
- <http://www.acm.org/sigs/sigkdd/explorations/issues/6-2-2004-12/dm-ssp-05-v1.pdf>
- KDD'06: <http://www.ncdm.uic.edu/dm-ssp-06.htm>

Acknowledgements



An earlier version of this tutorial has been presented at ECML/PKDD-2003 by Sarab Anand, Marko Grobelnik, Dietrich Wettschereck in September 2003

We gratefully acknowledge support by the FET-Open Coordination Action IST-6FP-02132 KDUbiq.

