

The Emissions Inventory Maintenance Application

by

Jo Ellen Brandmeyer, Aaron Parks, Sunil Rao, and
Robert Zerbonia

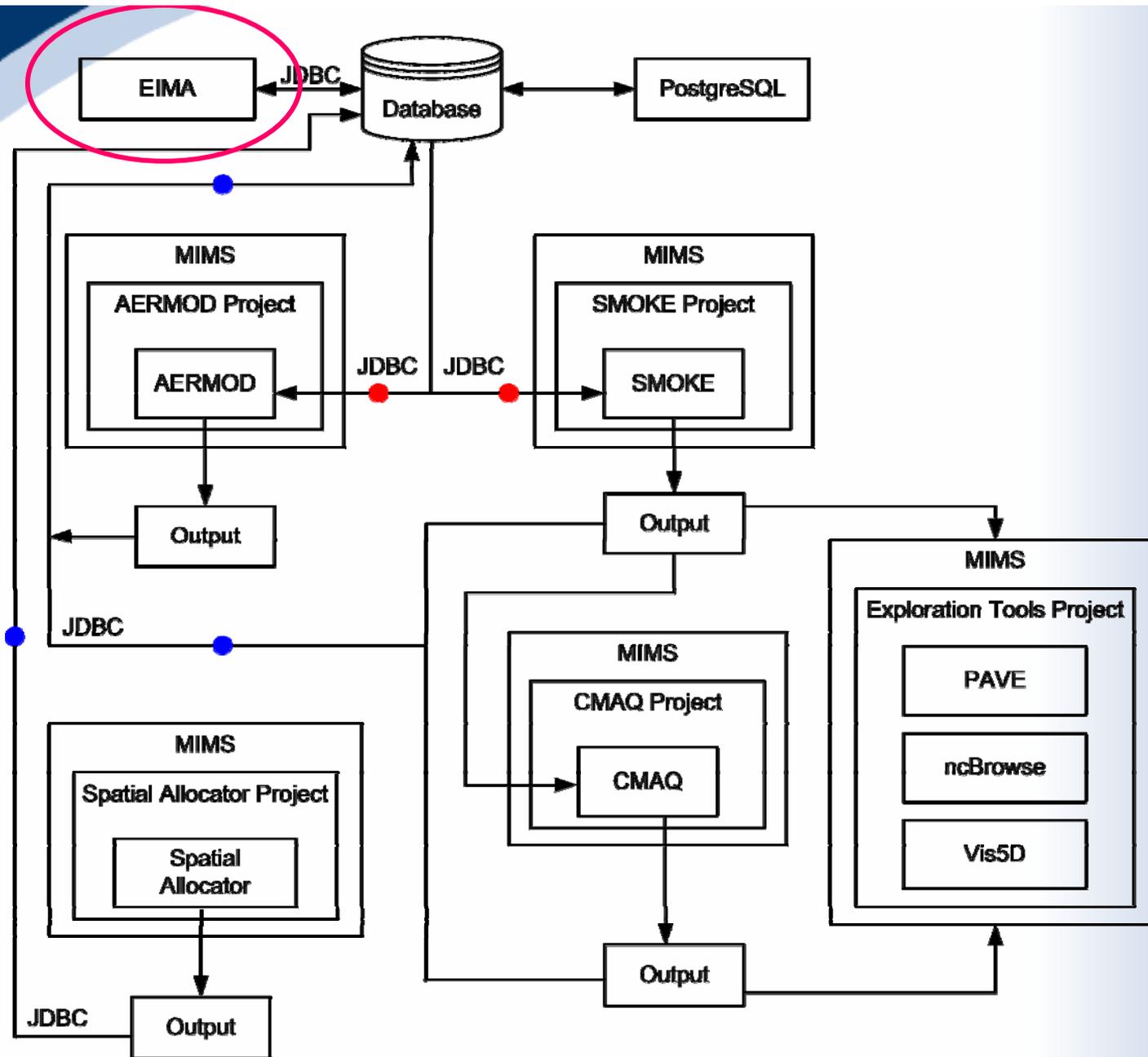
16th International Emission Inventory Conference
“Emission Inventories: Integration, Analysis, Communication”
May 14-17, 2007

Overview

- Project Background – AQMDSS and EIMA
- Key Requirements for EIMA
- Implementation
- Installation Experiences in Beijing, PRC

Project Background

- The Air Quality Management Decision Support System (AQMDSS)
 - ◆ Integrated system to support air quality modeling on multiple spatial and temporal scales.
 - AERMOD modeling system for industrial-scale simulations
 - CMAQ for regional-scale simulations
 - SMOKE for emissions inventory processing
 - PAVE and other utilities for visualization
 - ◆ Built around database implemented in PostgreSQL
 - Enhanced version of NIF v3.0
 - Transaction support for all-or-nothing data changes
 - ◆ User interface and model executions through MIMS
 - ◆ All open source and most public domain



Key Requirements for EIMA

- Flexibility
 - ◆ Quickly accommodate updates to AQMDSS components
- User Interface
 - ◆ Simplified Chinese for client
 - ◆ English for contractors
- Limited Data Access
 - ◆ Users update only those data tables pertinent to their work
 - ◆ Users may look up supporting data
- Data Validation
 - ◆ Maximize data validation during data entry
 - ◆ Maintain connection among data tables

Implementation

- Flexibility
 - Minimize Programming Changes
 - ◆ Data-driven implementation
 - ◆ Change data in database tables to change data maintenance screens
 - Java Internationalization
 - ◆ Display based on user
 - ◆ No code changes required to support additional languages
 - User Types
 - ◆ EIMA controls screen access to data tables by user type
 - ◆ PostgreSQL additional security
- User Interface
- Limited Data Access

Implementation (cont.)

- Data Validation
 - Code Fields
 - ◆ SMOKE codes added to database as NIF v3 enhancement
 - ◆ Drop-down boxes for data entry
 - Avoid Duplicate Entries
 - ◆ Unique keys in database
 - No Orphaned Records
 - ◆ Support parent/child relationships during data entry and update
 - ◆ Transaction support

Data Maintenance Screen for User Type *expert*

Data Maintenance
Select Table to Modify From a Table Group

Table Groups	Table Description	Add	Update
Scenario Data	Meteorological Scenarios	Add	Update
Area Source Data	Area Source Processes	Add	Update
On-Road Mobile Source Data	Reference Counties for On-Road Mobile Sources	Add	Update
Biogenic Source Data	BEIS3 Emission Factors	Add	Update
Point Source Data	Sites for Point Sources	Add	Update
Growth and Control Data	ALLOWABLE Control Packet	Add	Update
Ambient Data	Ambient Stations	Add	Update
AERMOD Data	AERMOD Site Data	Add	Update
Gridded Results	Gridded Results	Add	Update
Supporting Tables	Ambient Data Parameters	Add	Update
Master Tables	Types of Point Releases	Add	Update

Done

EIMA Meets Requirements

- EIMA version 1.0 meets all of these requirements.
- Successful installation in Beijing.
- Fulfills need to enter a new emissions inventory manually.
- Open source – released under GNU Public License.
- Implementation in PostgreSQL brings standard SQL to emissions inventory creation and processing.
 - ◆ Utilities to support ad hoc queries, to work with data structure, and to maintain data tables.
 - ◆ Supports stored procedures.
 - ◆ Interfaces provided to numerous programming languages – not just Java.

Installation Experiences in Beijing, PRC

- Installed at Beijing Municipal Environmental Protection Bureau
- System: Pentium Dual-Processor 64-bit Zenon Servers with HyperThreading
- No Internet Connection to Computational Computer
- Operating System: Red Hat Linux AS 4 with Simplified Chinese User Interface
- Java: Sun J2SE 64-bit
- Database Connectivity: PostgreSQL and its JDBC Driver
- Software Framework: MIMS and Libraries

Example Bulk Data Load

- Data Tables for Point Sources
 - ◆ Start with most basic tables
 - a.k.a. “supporting table” or “base table”
 - Not dependent on any other table
 - ◆ Load in order up the chain of dependencies

Base Tables

country_code	
country_code	
country_name	
date_created	
userid_created	
date_modified	
userid_modified	
5 rows	8 >

time_zones	
tz_code	
tz_desc	
22 rows	2 >

state_county_fips_codes		
state_county_fips_code	bpchar[5]	
state_county_fips_code_desc	varchar[60]	
state_fips_code	bpchar[2]	
county_fips_code	bpchar[3]	
date_created	timestamp[8]	
userid_created	varchar[20]	
date_modified	timestamp[8]	
userid_modified	varchar[20]	
country_code	bpchar[2]	
county_std_tz	bpchar[3]	
dst_code	bool[1]	
county_pop	int4[4]	
county_pop_yr	int4[4]	
< 2	8,738 rows	68 >

Site for a Point Source (Facility)

point_source_site		
country_code	bpchar[2]	
state_county_fips	bpchar[5]	
pt_facility_id	varchar[15]	
facility_id_2	varchar[12]	
facility_category_code	bpchar[2]	
cems_facility_code	varchar[6]	
sic_primary	bpchar[4]	
facility_name	varchar[80]	
site_description	varchar[40]	
location_address	varchar[50]	
city	varchar[60]	
county	varchar[40]	
province	varchar[40]	
postal_code	varchar[14]	
facility_latitude	float8[8]	
facility_longitude	float8[8]	
facility_elevation	float4[4]	
facility_location	point[16]	
tri_id	varchar[20]	
date_created	timestamp[8]	
userid_created	varchar[20]	
date_modified	timestamp[8]	
userid_modified	varchar[20]	
< 4	101 rows	7 >

Indexes:

Column(s)	Type
country_code + state_county_fips + pt_facility_id	Primary key
pt_facility_id	Must be unique

Site for a Point Source

Adding a Record (Part I: Primary Fields)

Enter or Select Primary Values to Create a New Record
Table: Sites for Point Sources

Variable	Values Range	Permitted Values	Value
Country ID	No Range	Show Values	0
FIPS	No Range	Show Values	0
Facility ID	No Range		

OK Done

Add Note for This Record

Site: Rest of Fields

Adding a Record (Part II: Additional Fields)

Enter or Select Additional Values to Create a New Record
Table: Sites for Point Sources

Variable	Values Range	Permitted Values	Value
Additional Facility ID	No Range	na	
Facility Category ID	No Range	na	01
CEMS Facility ID	No Range	na	
SIC ID	No Range	na	0
Name of Facility	No Range	na	
Site Description	No Range	na	
Address	No Range	na	
City	No Range	na	
County	No Range	na	
Province	No Range	na	
Postal Code	No Range	na	

Next Level

state_county_fips_codes		
state_county_fips_code		
state_county_fips_code_desc		
state_fips_code		
county_fips_code		
date_created		
userid_created		
date_modified		
userid_modified		
country_code		
county_std_tz		
dst_code		
county_pop		
county_pop_yr		
< 2	8,738 rows	68 >

emission_release_point_type		
emis_rel_point_type_code		
emis_rel_point_type_desc		
	6 rows	1 >

sic		
sic_code		
sic_description		
	1,090 rows	6 >

unit_codes		
unit_of_measure_code		
unit_of_measure_desc		
conversion_factor_amt		
net_standard_code		
	137 rows	20 >

xy_coordinate_type		
xy_coord_type_code		
xy_coord_type_desc		
	2 rows	1 >

horizontal_collection_method		
horizontal_collection_method_code		
horizontal_collection_method_code_description		
	38 rows	1 >

horizontal_reference_datum_code		
horizontal_reference_datum_code		
horizontal_reference_datum_code_desc		
	3 rows	1 >

reference_point_code		
reference_point_code		
reference_point_code_description		
	8 rows	1 >

coordinate_data_source_code		
coordinate_data_source_code		
coordinate_data_source_code_description		
	79 rows	1 >

point_source_er		
country_code	bpchar[2]	
state_county_fips	bpchar[5]	
pt_facility_id	varchar[15]	
pt_eu_id	varchar[6]	
er_point_type	bpchar[2]	
cems_boiler_id	bpchar[5]	
sic_eu	bpchar[4]	
design_capacity	float4[4]	
design_capacity_unit_numerator	varchar[10]	
design_capacity_unit_denominator	varchar[10]	
max_nameplate_capacity	float4[4]	
eu_description	varchar[80]	
stack_height	float4[4]	
stack_diameter	float4[4]	
stack_fenceline_distance	float4[4]	
exit_gas_temp	float4[4]	
exit_gas_velocity	float4[4]	
exit_gas_flow_rate	float8[8]	
x_coordinate	float8[8]	
y_coordinate	float8[8]	
xy_coordinate_type	varchar[8]	
stack_elevation	float4[4]	
er_location	point[16]	
horizontal_area_fugitive	int8[8]	
release_ht_fugitive	int8[8]	
fugitive_unit	varchar[10]	
release_pt_description	varchar[80]	
horizontal_method_code	bpchar[3]	
horizontal_accuracy	varchar[6]	
horizontal_ref_datum_code	bpchar[3]	
reference_point_code	bpchar[3]	
source_map_scale_number	varchar[10]	
coordinate_data_source_code	bpchar[3]	
date_created	timestamp[8]	
userid_created	varchar[20]	
date_modified	timestamp[8]	
userid_modified	varchar[20]	
< 12	200 rows	0 >

Conclusions

- EIMA Meets System Requirements
- EIMA Version 1 Successfully Installed in Beijing, PRC
 - ◆ User interface in both Simplified Chinese and English
 - ◆ Available under GNU Public License
- AQMDSS Links Database to SMOKE
 - ◆ Data extraction utilities create SMOKE input files from the emission inventory database
 - ◆ Code checks during data entry reduces QA errors during SMOKE processing
- Emissions Inventory Readily Accessible
 - ◆ PostgreSQL provides numerous programming and ad hoc query interfaces to emissions inventory



Questions?

