

Implementing MOVES as Quality Software

A presentation by
Mitch Cumberworth
at the

EPA Mobile Source Present
and Future Models Workshop
- November 6, 2002 -



MOBILE6 Software Limitations

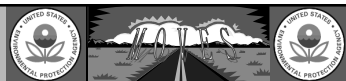
- **Not Easy to Further Maintain and Adapt**
 - Already modified hundreds of times
 - Insufficient attention given to software design
 - Data stored in Fortran Block Data statements
 - Programming changes typically needed to interface to other software
- **Lacks Graphical User Interface (GUI)**



Software Goals for MOVES (1 of 2)

- **Use State-of-the-Art Methods and Tools**
 - How would one produce MOVES today?
 - Use industry-standard tools
 - Adopt object orientation
- **Document Software Design**
- **Preserve Important Characteristics of MOBILE/NONROAD**
 - Still be able to execute on single desktop computer
 - Not require users to purchase software licenses

3



Software Goals for MOVES (2 of 2)

- **Portable to Other Platforms, e.g. :**
 - Unix (several varieties) and Linux™
 - Macintosh OS™
 - **Future Platforms**
- **Execution Time Performance**
- **Ease of Use**
 - Facilitate Development of GUIs
 - Accept simple forms of input data
- **Facilitate Development with MIMS**
 - EPA Modeling Framework

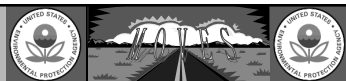
4



MOVES Development Methods: “Moderately Extreme Programming”

- Treat process of software design, development, and testing as **iterative**.
- Produce **automated tests** as software is written
- Have **user representatives** on team
- **But not abandoning traditional approach**
 - Produce design documentation
 - Maintain version control:
 - Why was software changed? When? By whom?

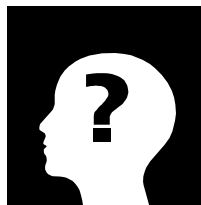
5



Choosing Programming Languages for MOVES

• JAVA™

• C++



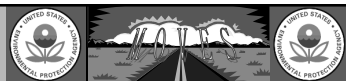
6



Principal Choice: JAVA™

- Object-oriented, state of the art
- Has associated programming standards and HTML program documentation tool
- Widely used, popular among programmers
- “Write once, run anywhere.”
- Excellent for GUIs
- Is native language of MIMS
- Potential problem: Performance?

7



Backup Choice: C++

- Object-oriented, state of the art
- Widely used, popular among programmers
- Will be used in MOVES if and where necessary to improve performance

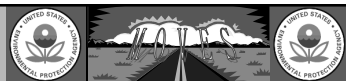
8



Choosing a Database for MOVES

- **Other Principal Tool Needed is Database Management Software**
- **A Significant Departure from MOBILE6**
- **Should be:**
 - Relational, implement Structured Query Language
 - Work with Java™ and C++
 - Platform independence
 - Not require user license
 - High performance

9



Tentative Choice is MySQL™

- **Product of MySQL AB Corporation**
- **“The most popular Open Source SQL Database...”**
- **Implements only basic level of SQL, but is high performance**
- **Written in C++**
- **Binary releases available for many operating systems**

10



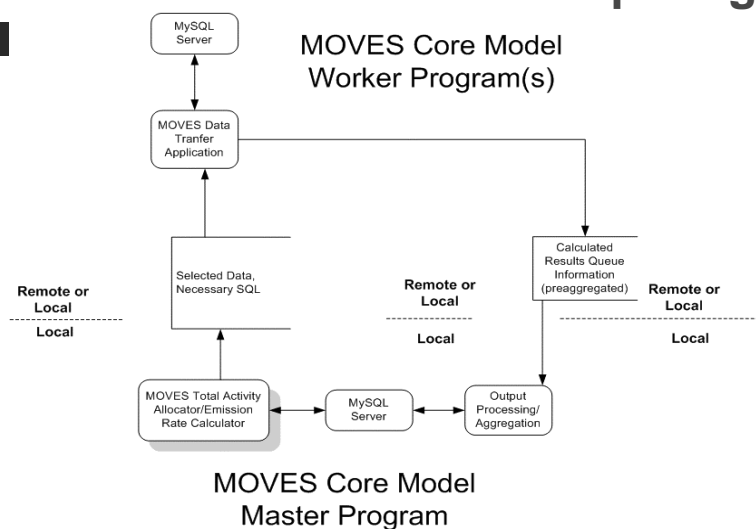
Other Software Tools

- **Relational Database CASE Tool**
 - ERWin™
- **Software Version Control**
 - Concurrent Versioning System (CVS™)
- **Build Utility**
 - ANT™
- **Testing Utilities**
 - JUnit™, JFCUnit™

11



MOVES Application Architecture for Distributed or Grid Computing



12



For More Information

- **About Extreme Programming**
 - *Extreme Programming Explained - Embrace Change* by Kent Beck, Copyright © 2000 by Addison-Wesley
- **About Java**
 - URL <http://java.sun.com>
- **About MySQL**
 - URL <http://www.mysql.com>