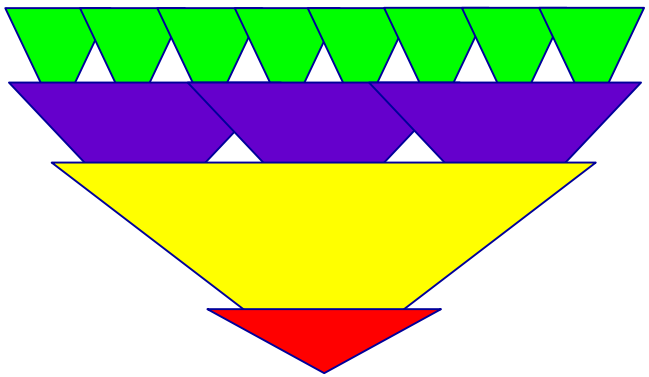
# Frameworks, Environments and Components for Visualization



James “Jeeembo” Kohl
Oak Ridge National Laboratory

DIVA Visualization Framework Workshop
Tuesday, July 8, 2003



Research supported by the Mathematics, Information and Computational Sciences Office, Office of Advanced Scientific Computing Research, U. S. Department of Energy, under contract No. DE-AC05-00OR22725 with UT-Battelle, LLC.

# Fundamental Terminology Conundrum

- "Components" for Visualization
  - Basic Building Blocks…
  - What Granularity?
    - Object? Module? Library? Application?
- What is a "Framework"? ("F" vs. "f" ?!)
  - Simple Container/Substrate for Composition?
  - Include Core Functionality…?
  - Application-Specific "Frameworks"…

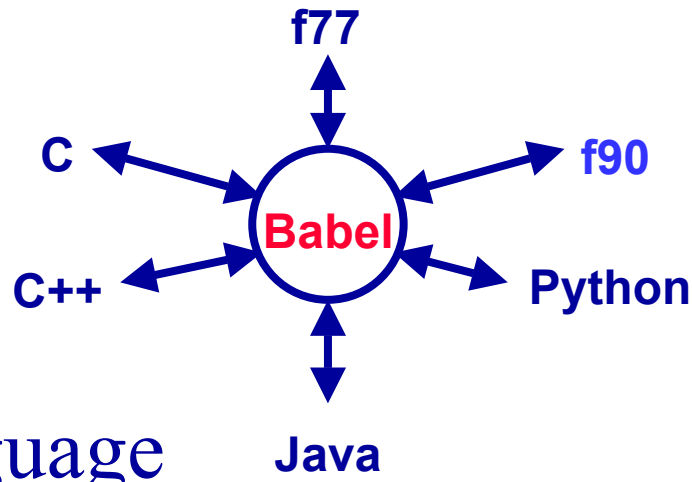
# Terminology Proposal

- **Framework** ≡ Simple Component Container
  - Substrate for Plugging In/Composing Components
  - Basic Run-Time Execution Model (Parallel/Dist)
- **Environment** ≡ Application-Specific Core (Viz)
  - Add *Intrinsic* Functional Modules to "Framework"
  - Main Loop, Data Wrangling, Networking/Compositing
- **Application** ≡ Complete Component Collection
  - Actual Viz & Data Analysis Modules, Composed
  - Isosurface, Volume Render, Feature Tracking…

# Next Step ~ Interface Specification!

- No Implementation (Yet)! ☺
  - Just *Abstract* Interface Spec…
- "Standardize" and "Generalize" Where Possible
  - Decide What Basic Functions are Needed
  - Decide How Methods Should Be Formulated
    - High-Level Functional Blocks, Interaction Patterns, Data…
- Incorporate Existing Tools and Technology
  - Plan Ahead, Make Sure All Existing Models Covered
  - Don't Necessarily Mimic Any *One* Tool ~ "*General*"

# "*Abstract*" Interfaces Using SIDL/Babel




- Not bound to a specific language
  - Use SIDL spec for language interoperability
  - Babel compiles interface for target language
- Still a good idea, even if everything is C++:
  - Scientific simulation probably written in F90
  - Some people actually like Java and Python ☺
- Compatibility with CCA Technology…

# Discussion Topics

- What are the "Intrinsic" or Core Components?
  - What are the Internal Subsystems?
  - Where Do The Key Pieces Go?!
- What are the Main Level Viz Components?
  - At what Granularity should things be defined?
- Actually Start Specifying the Interfaces!
  - Sketch out the Basic Methods…
  - Even if it involves major "hand waving"… ☺